

Introduction to Computer Architecture

Lecture 1: Introduction and Basics

Pooyan Jamshidi

Week 1: January 9 & 11, 2024



**Engineering
and Computing**

UNIVERSITY OF SOUTH CAROLINA

CSC212: Introduction to Computer Architecture | Spring 2024 | <https://pooyanjamshidi.github.io/csce212/>

[Slides are primarily based on those of Onur Mutlu for the Computer Architecture Course at CMU]

Brief Self Introduction



■ Pooyan Jamshidi

- ❑ **Assistant Professor @ USC (CSE)**, since August 2018
- ❑ Postdoc 2 @ Carnegie Mellon University (US), 2016 - 2018
- ❑ Postdoc 1 @ Imperial College London (UK), 2014 - 2016
- ❑ Ph.D. from Dublin City University (Ireland), 2010 - 2014
- ❑ M.Sc. from Amirkabir University of Technology (Iran), 2006
- ❑ B.Sc. from Amirkabir University of Technology (Iran), 2003
- ❑ Worked with **Google** and **NASA**
- ❑ Homepage: <https://pooyanjamshidi.github.io/>
- ❑ Email: pjamshid@cse.sc.edu

■ Research and Teaching in:

- ❑ Machine Learning Systems = AI/ML + Computer Systems
- ❑ Autonomous Robots = AI/ML + Robotics
- ❑ Causal AI = Causal Inference, Causal Representation Learning, Transfer Learning
- ❑ Neural Architectures + Hardware Accelerators
- ❑ AI/ML Systems (See CSCE 585)
- ❑ Autonomous and Adaptive Systems (NASA Autonomous Space Lander)

Artificial Intelligence and Systems Laboratory (AISys)

<https://pooyanjamshidi.github.io/AISys/>

Research Areas:

- Causal AI
- ML for Systems
- Systems for ML
- Adversarial ML
- Robot Learning

Sponsors:



Sonam Kharde
(Postdoc)



M.A. Javidian
(Postdoc)



Fatemeh Ghofrani
(PhD student)



Nasrin Imanpour
(PhD student)



Kimia
Noorbakhsh (RA)



Saeid Ghafouri
(RA)



Hamed Damirchi
(PhD student)



Shahriar Iqbal
(PhD student)



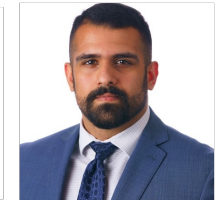
Jianhai Su
(PhD student)



Abir Hossen
(PhD student)



Mahdi Sharifi
(PhD student)



Morteza Maleki
(RA)

Collaborators:



Key Research Directions in Computer Architecture at AISys

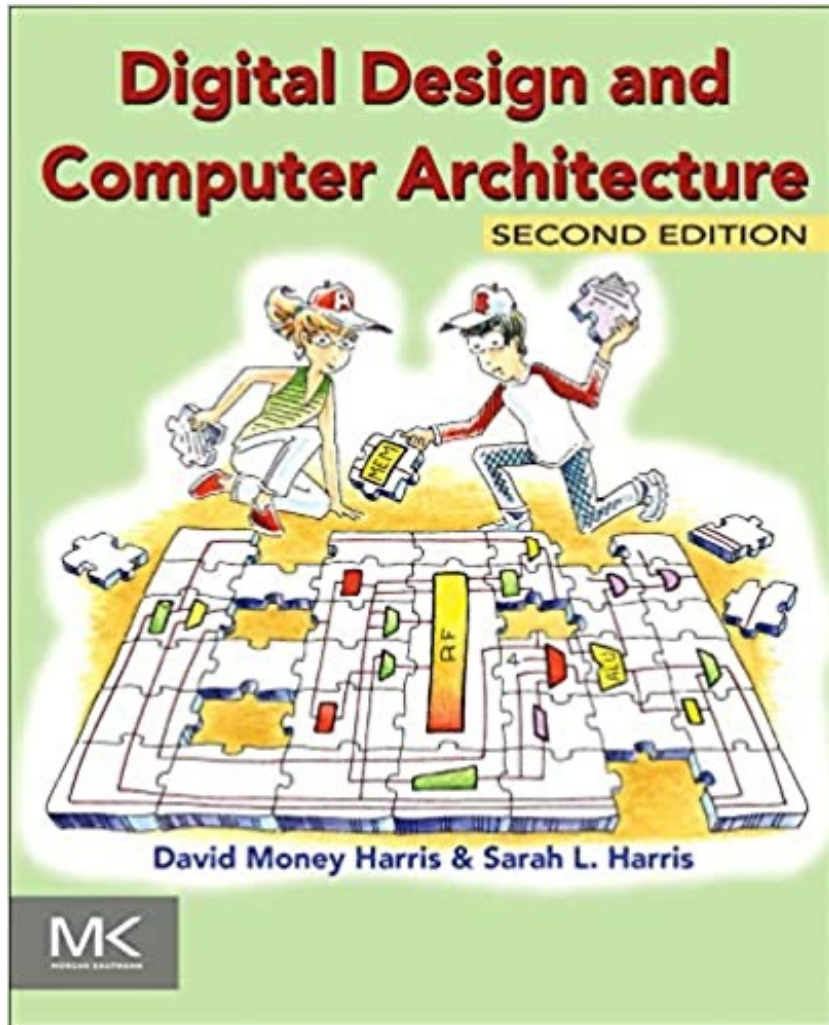
- Low-latency and Energy-efficient Neural Architectures
 - Software-Hardware Co-Design
 - Neural Architecture Search
 - Hardware Accelerators
- Domain-Specific Architectures
 - Architectures for AI/ML
 - Hardware Accelerator for LLMs

Course Information

- **Course Website:** <https://pooyanjamshidi.github.io/csce212/>
- **Piazza** (Communications):
 - Discussion boards for each assignment and the course overall
 - Please post all questions on Piazza so that others can benefit from your questions
 - Answer others' questions - if you know the answer ;-)
 - Learn from others' questions and answers
- **GradeScope** (Assignments)
- **Teaching Assistant**
 - Rasool Sharifi
 - Homepage: <https://rasool-sharifi.github.io/>
 - Email: ASHARIFI@email.sc.edu



Textbook (Harris & Harris)



- Chapter 6 (Architecture)
- Chapter 7 (Microarchitecture)
- Chapter 8 (Memory?)

Evaluation (subject to minor changes)

Assignments	50%
Midterm	25%
Final	25%

Please volunteer to present a related topic to architecture if you are excited about it!



Basic Goals & Structure the Computer Architecture Course

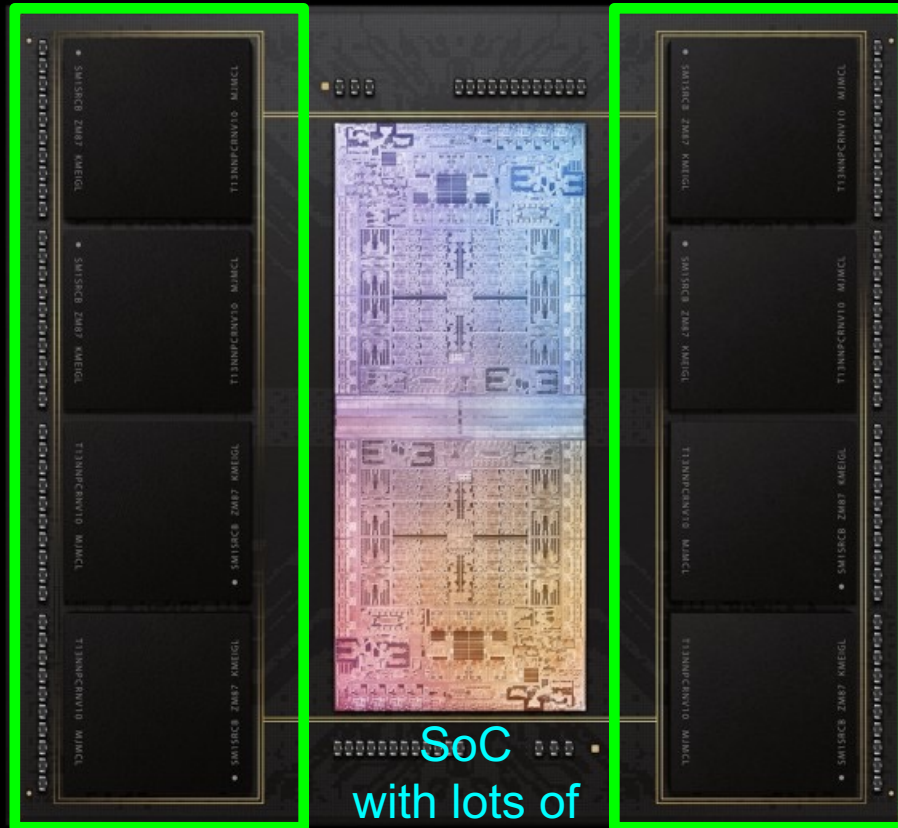
What Will We Learn in This Course?

How Computers Work

(from the ground up)

We Will Study How Something Like This Works

Sensors



Storage

Main Memory

SoC
with lots of
compute
& caches

Main Memory

Storage

Apple M1 Ultra System (2022)

Major High-Level Goals of This Course

- In Introduction to 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 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
 - ...

Course Components

- **Lectures** (understanding concepts)
- **Readings** (reinforcing & going deeper)
- **Homework** (problem-solving & preparation)
- **Project** (hands-on experience in some concepts)
- **Exam** (test of understanding)

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

- **My advice:** Focus on learning & scholarship & understanding

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

Summary

- Learning is for life (never ends)

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

**Your mindset
will determine
what you
get out of the course**

How to Approach This Course

Find and choose
the learning style
that works best for you

What Will We Learn in This Course?

How Computers Work

(from the ground up)

And Why We Care

Why Do We Have Computers?

Why Do We Do Computing?

To Solve Problems

To Gain Insight

**To Enable
a Better Life & Future**

How Does a Computer Solve Problems?

Orchestrating Electrons

In today's dominant technologies

How Do Problems Get Solved by Electrons?

So, I Hope You Are Here for This

CSCE 145/206

- How does an assembly program end up executing as digital logic?
- **What happens in-between?**
- How is a computer designed using logic gates and wires to satisfy specific goals?

CSCE 211

“C” as a model of computation

Programmer’s view of how a computer system works

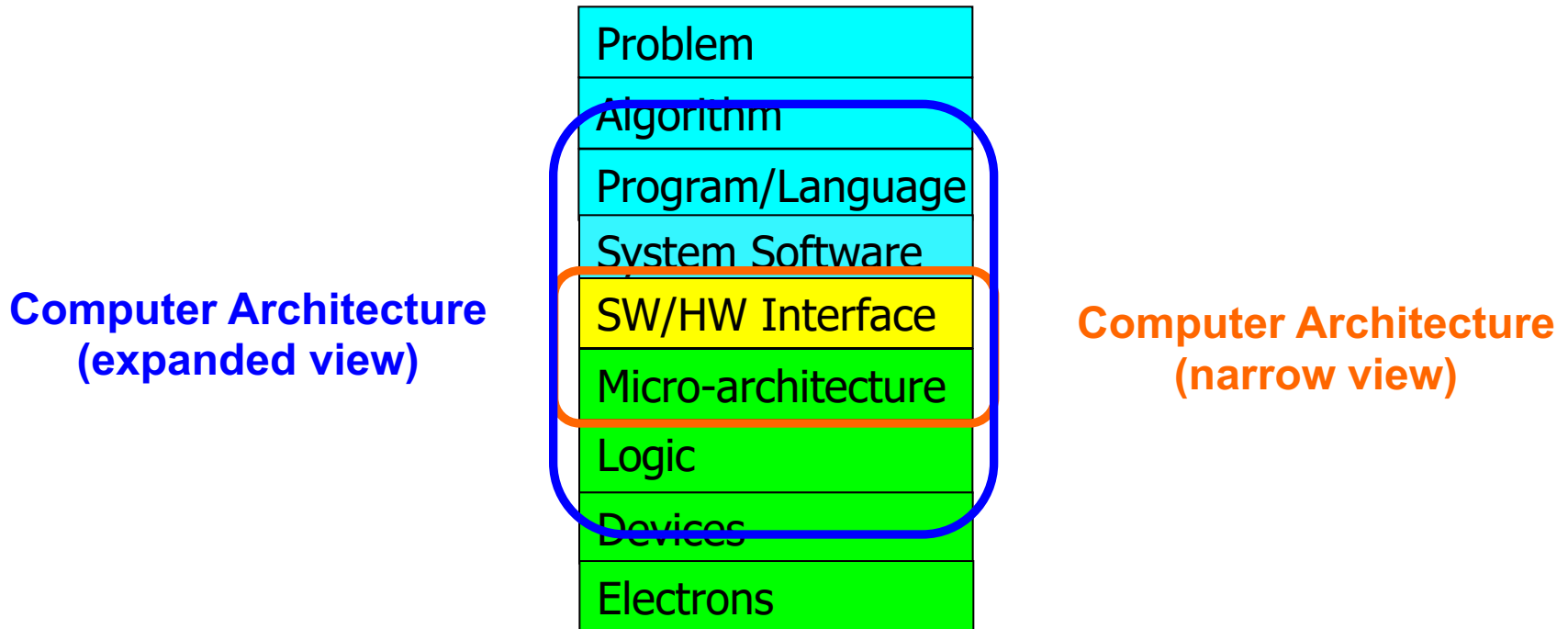
*Architect/microarchitect’s view:
How to design a computer that meets system design goals.*

Choices critically affect both the SW programmer and the HW designer

HW designer’s view of how a computer system works

Digital logic as a model of computation

The Transformation Hierarchy



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**

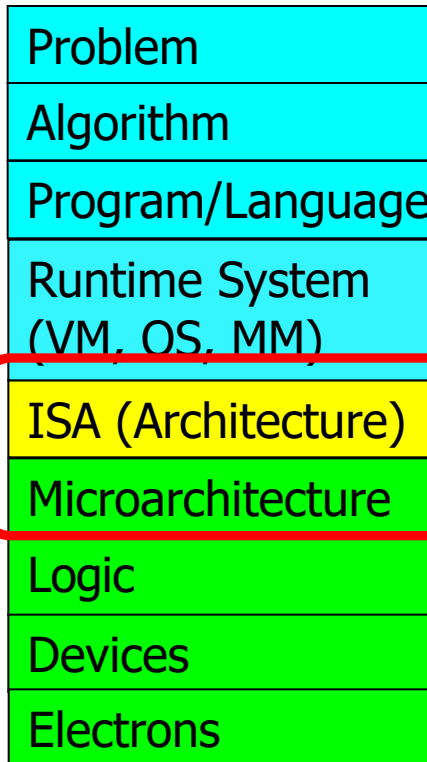
Many algorithms for the same problem

Microarchitecture

An implementation of the ISA

Digital logic circuits

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



ISA

(Instruction Set Architecture)

Interface/contract between SW and HW.

What the programmer assumes hardware will satisfy.



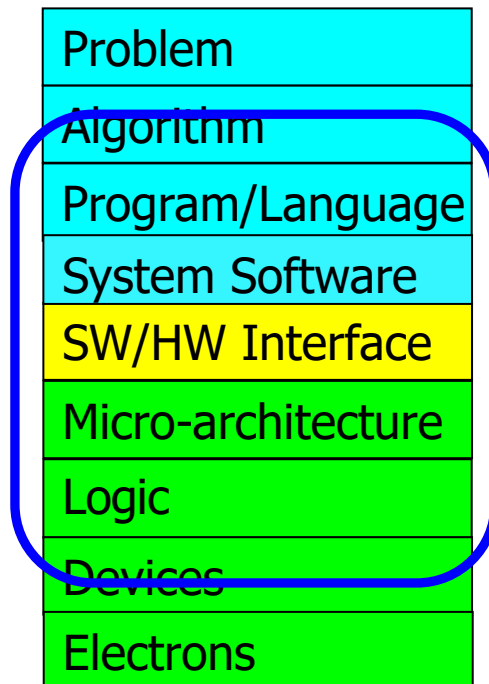
Computer Architecture

- 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

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

Different Platforms, Different Goals



Different Platforms, Different Goals



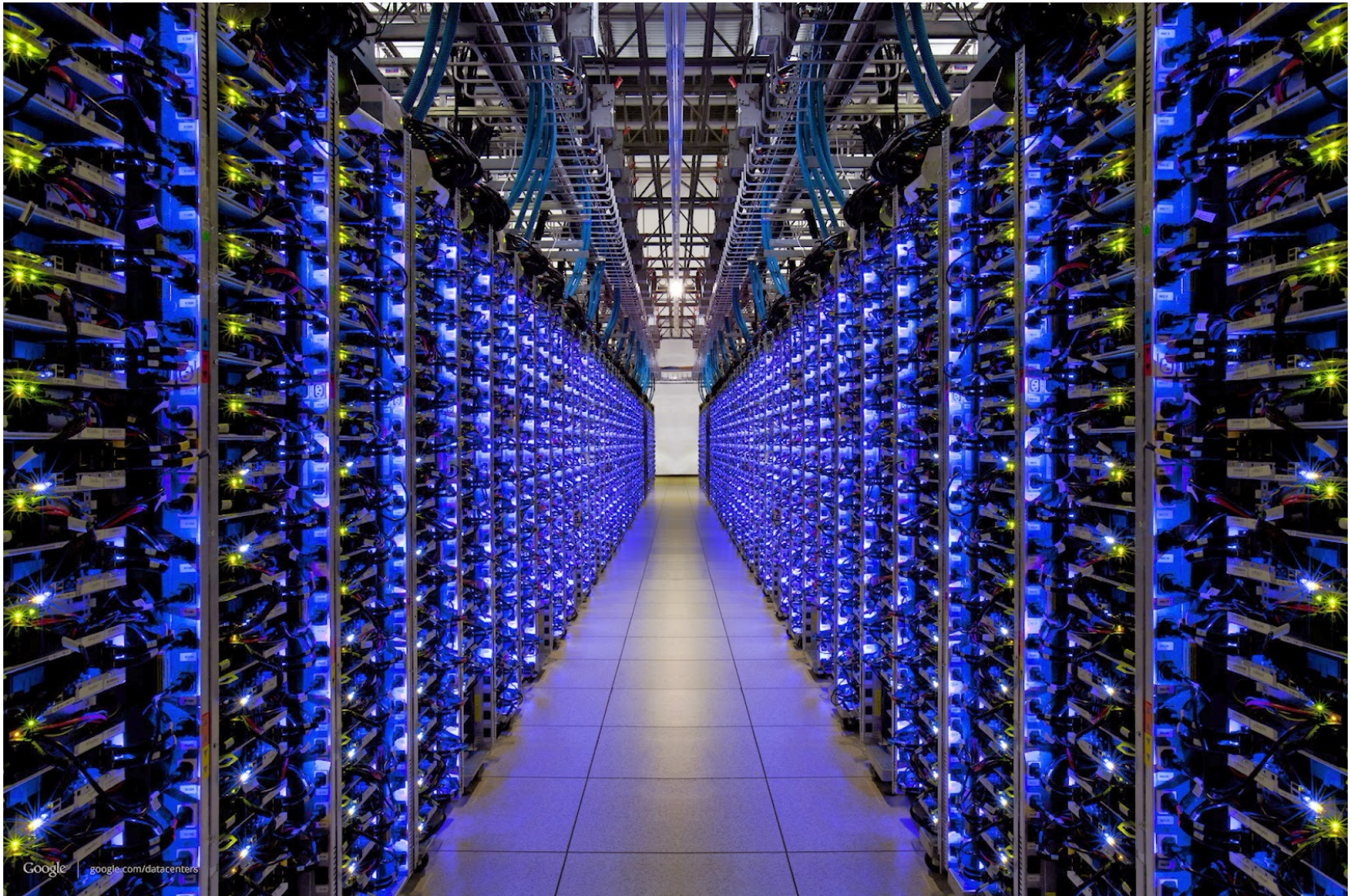
Different Platforms, Different Goals



Different Platforms, Different Goals



Different Platforms, Different Goals



Different Platforms, Different Goals



Jack Dongara

Different Platforms, Different Goals



Different Platforms, Different Goals

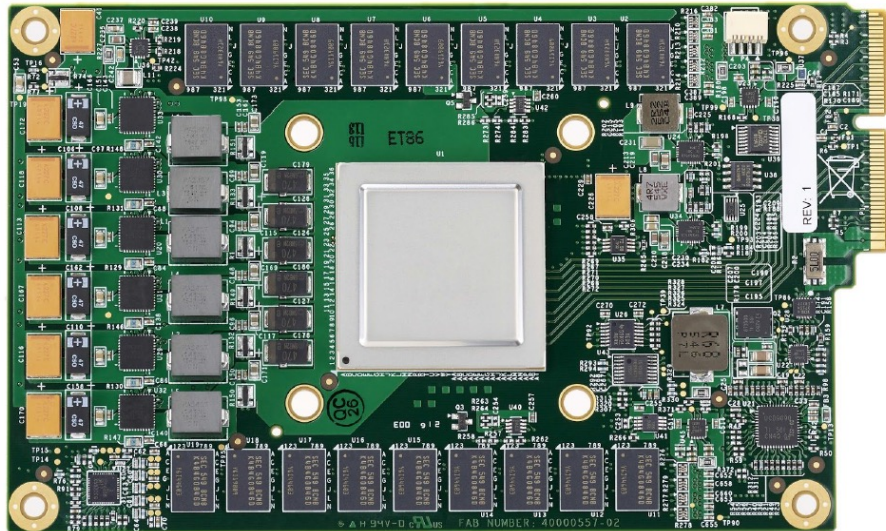


Figure 3. TPU Printed Circuit Board. It can be inserted in the slot for an SATA disk in a server, but the card uses PCIe Gen3 x16.

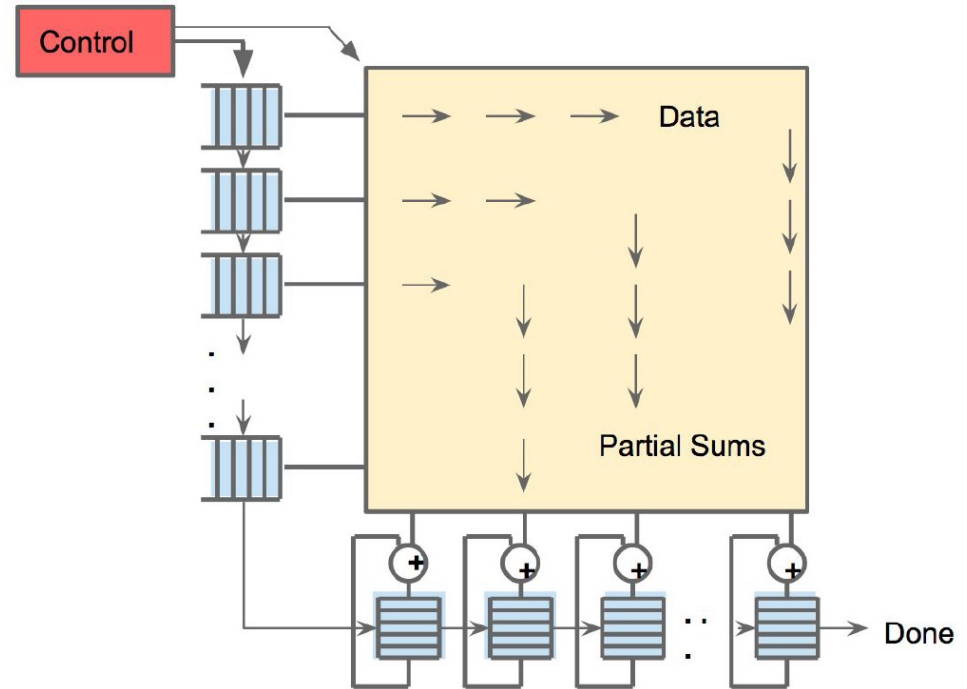
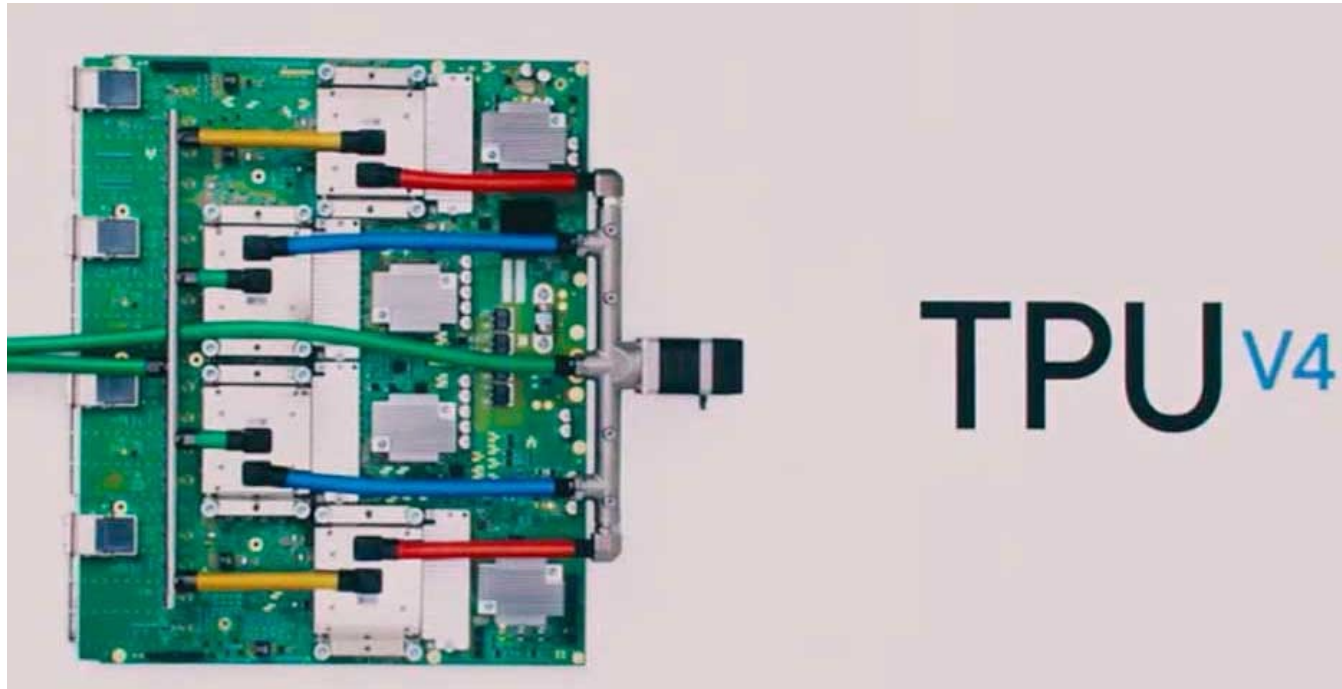


Figure 4. Systolic data flow of the Matrix Multiply Unit. Software has the illusion that each 256B input is read at once, and they instantly update one location of each of 256 accumulator RAMs.

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

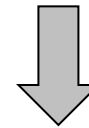
Different Platforms, Different Goals



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

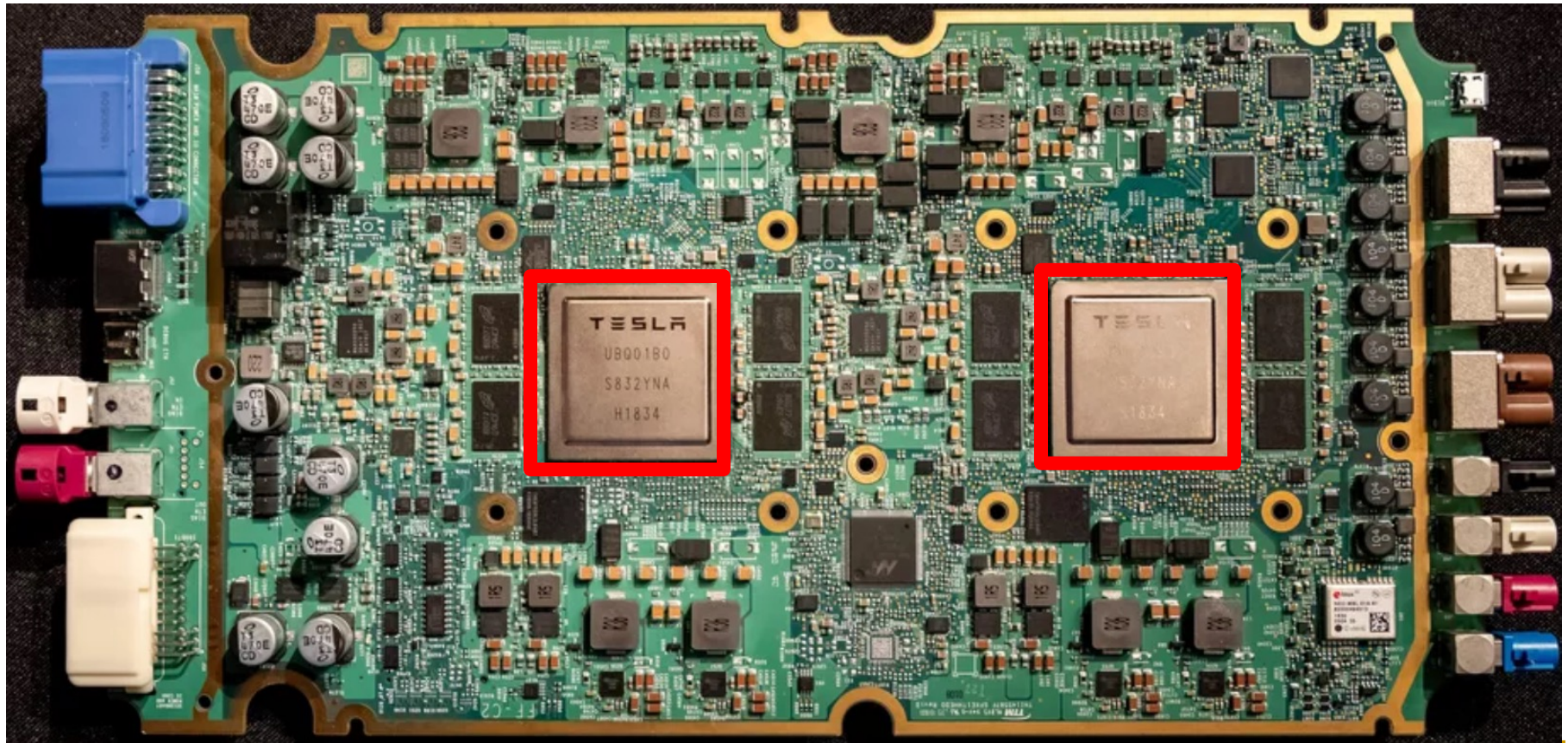


1 ExaFLOPS per board

<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 mm², 6 billion transistors, 600 GFLOPS GPU, 12 ARM 2.2 GHz CPUs.
- Two redundant chips for better safety.



Different Platforms, Different Goals



■ Tesla Dojo Chip & System

D1 Chip

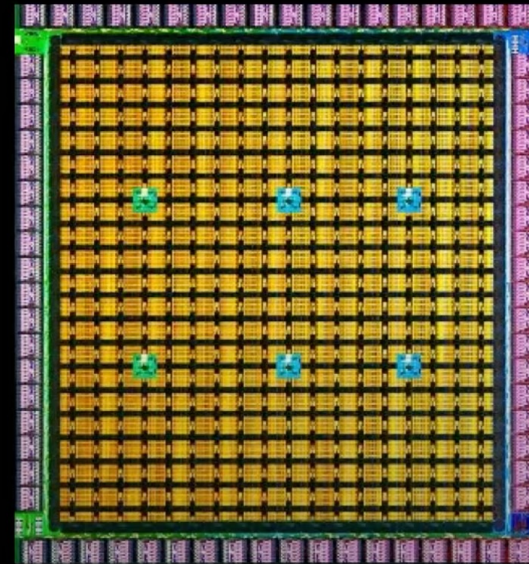
362 TFLOPs BF16/CFP8

22.6 TFLOPs FP32

10TBps/dir. On-Chip Bandwidth

4TBps/edge. Off-Chip Bandwidth

400W TDP



645mm²
7nm Technology

50 Billion
Transistors

11+ Miles
Of Wires

Different Platforms, Different Goals



■ Tesla Dojo Chip & System

Neural Network Training - Compute



2021: 3x Clusters

1752 GPUs
5PB NVME
Infiniband EDR

Auto-labelling

4032 GPUs
8PB NVME
Infiniband EDR

Training

5760 GPUs
12PB NVME
Infiniband HDR

Training



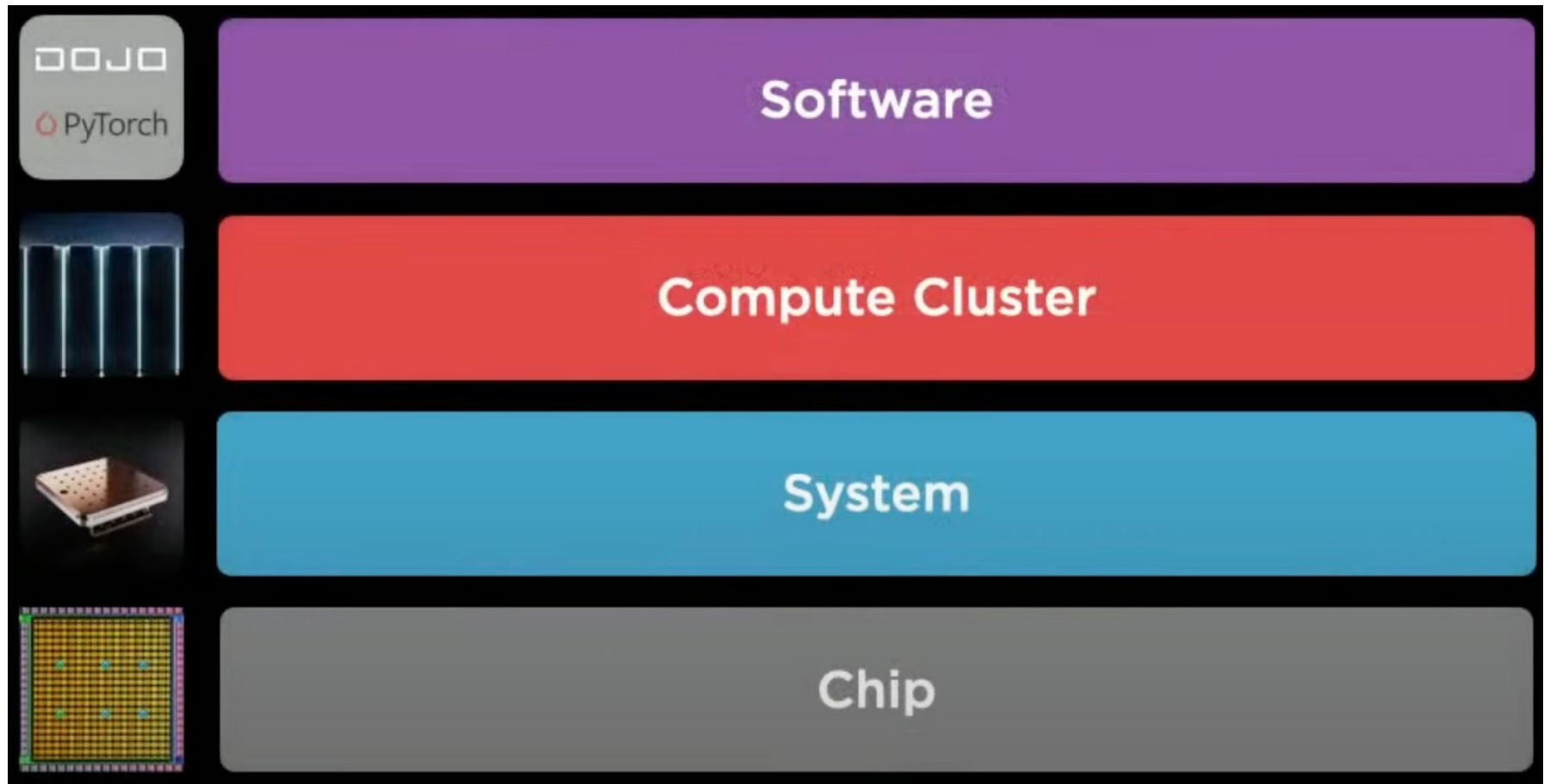
1:45:20 / 3:03:20 • Hardware Integration >



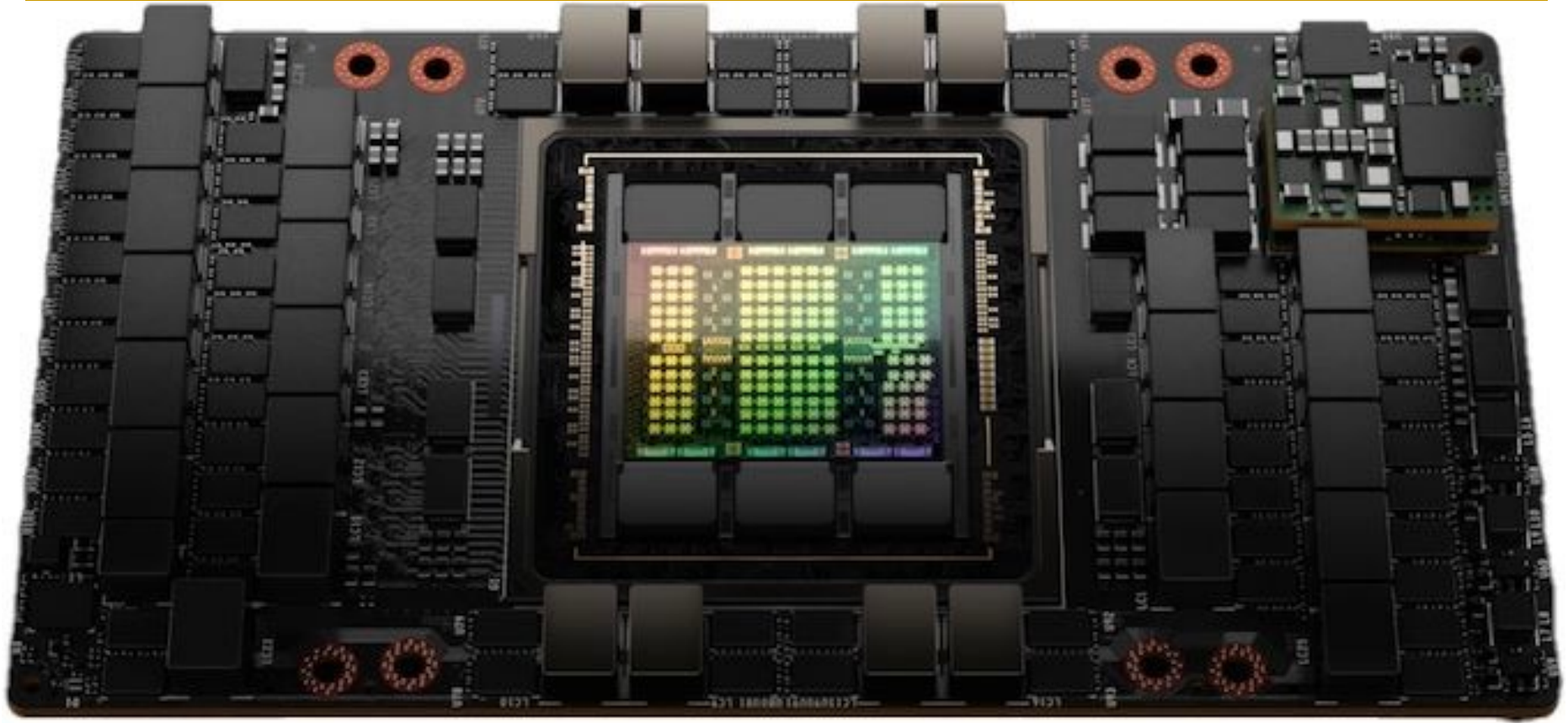
Different Platforms, Different Goals



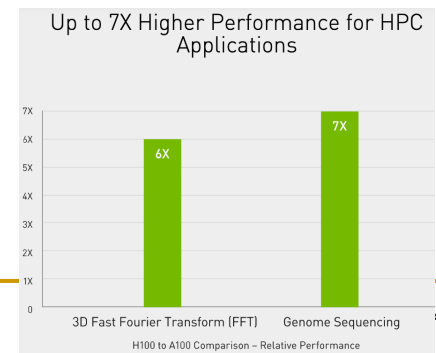
- Tesla Dojo Chip & System



Different Platforms, Different Goals

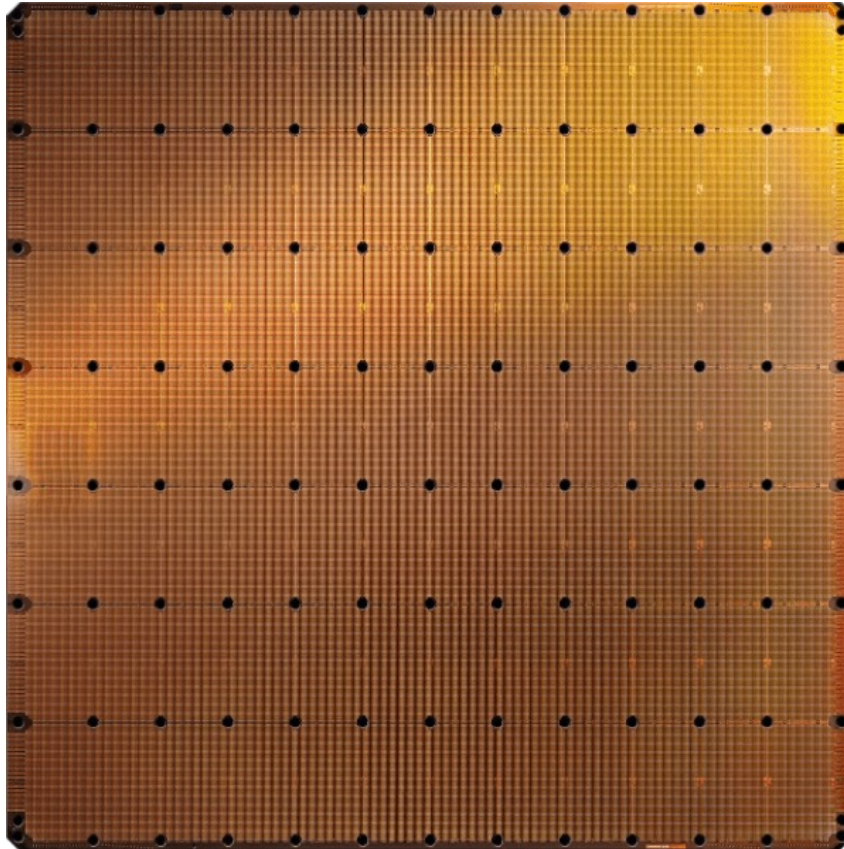


NVIDIA is claiming a **7x improvement** in dynamic programming algorithm (**DPX instructions**) performance on a single H100 versus naïve execution on an A100.



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

Different Platforms, Different Goals



Cerebras WSE-2
2.6 Trillion transistors
46,225 mm²

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



Largest GPU
54.2 Billion transistors
826 mm²
NVIDIA Ampere GA100

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](https://doi.org/10.1109/MM.2020.3013728)

FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications

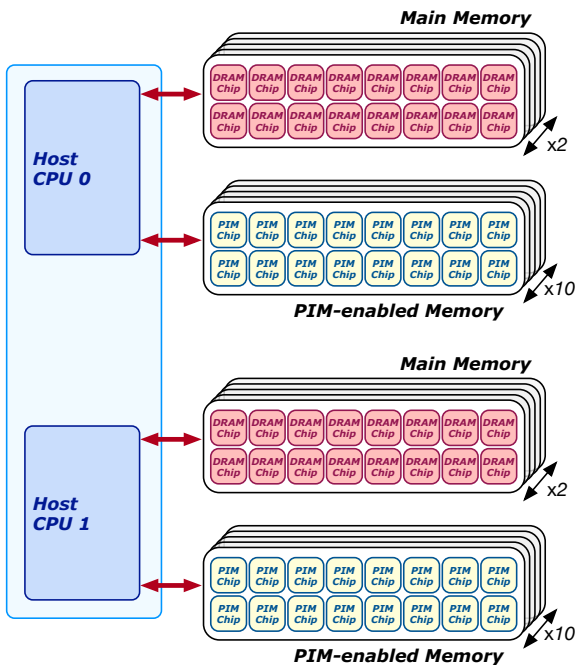
July-Aug. 2021, pp. 39-48, vol. 41

DOI Bookmark: [10.1109/MM.2021.3088396](https://doi.org/10.1109/MM.2021.3088396)



SmidgION from ONT

Different Platforms, Different Goals



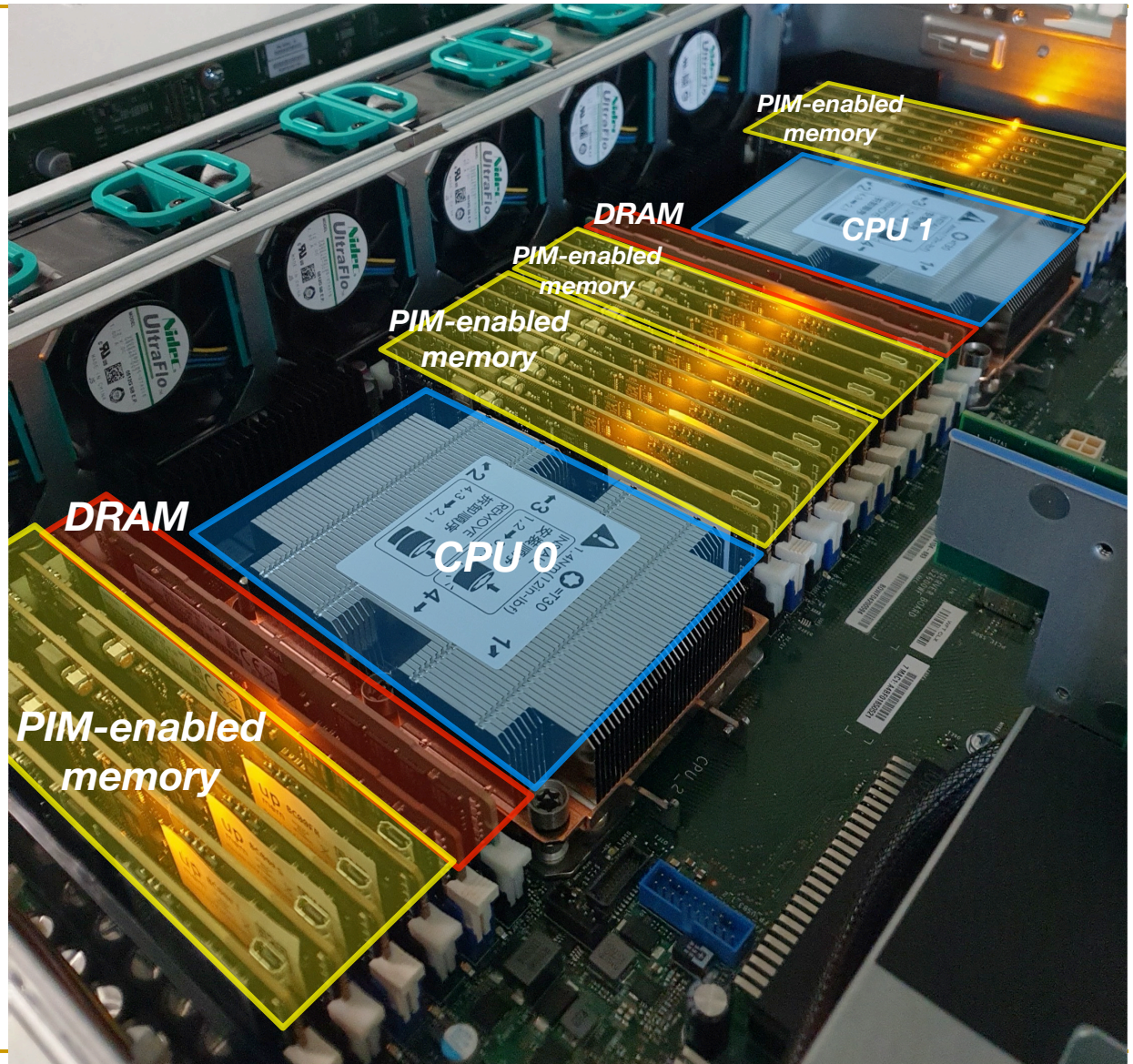
Benchmarking a New Paradigm: An Experimental Analysis of a Real Processing-in-Memory Architecture

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 440 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.

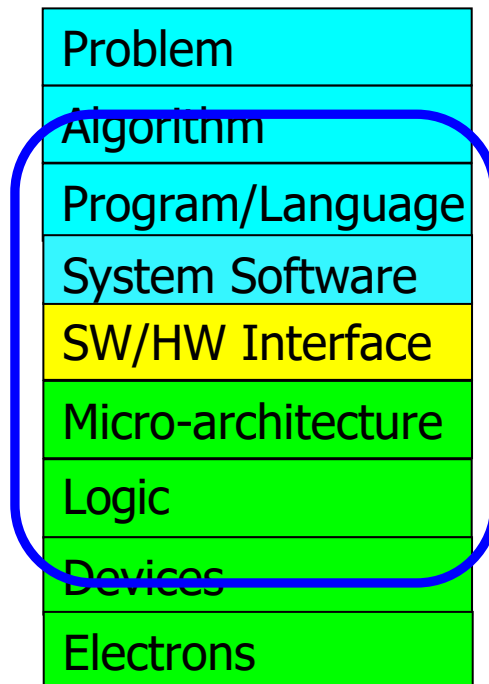


<https://arxiv.org/pdf/2105.03814.pdf>

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

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?

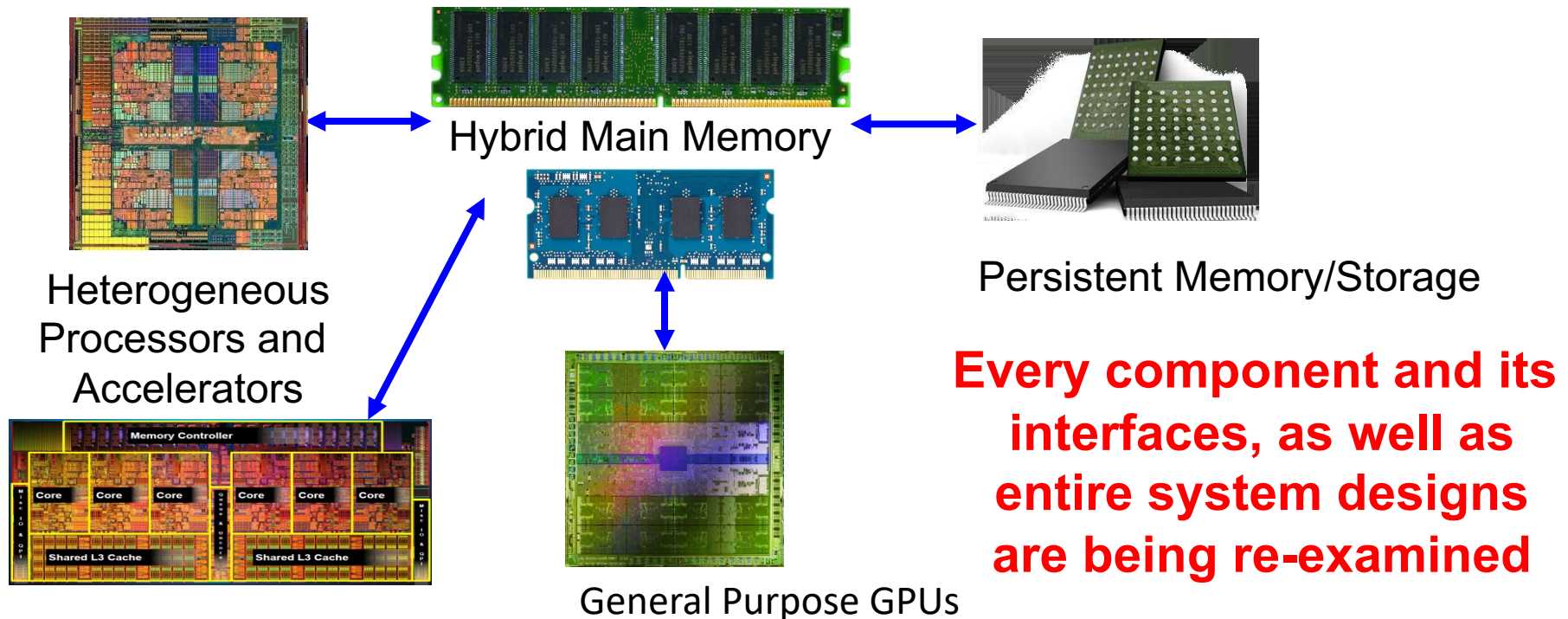
- **Enable better systems:** make computers faster, cheaper, smaller, more reliable, ...
 - By exploiting advances and changes in underlying technology/circuits
- **Enable new applications**
 - Life-like 3D visualization 20 years ago? Virtual reality?
 - Self-driving cars?
 - Personalized genomics? Personalized 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



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.

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. Leiserson¹,  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.

Axiom, Revisited

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

but **much more so**

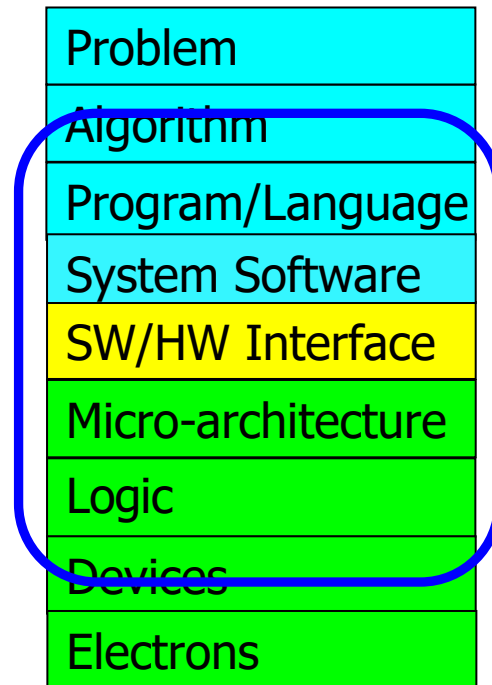
when you

communicate well between and optimize across

the top and the bottom

Hence the Expanded View

**Computer Architecture
(expanded view)**



Computer Architecture

Why Is It So Exciting Today?

Many Interesting Things
Are Happening Today
in Computer Architecture

Many Interesting Things
Are Happening Today
in Computer Architecture

Performance
Energy Efficiency
Sustainability

Many Interesting Things
Are Happening Today
in Computer Architecture

Reliability
Safety
Security
Privacy

Many Interesting Things
Are Happening Today
in Computer Architecture

More Demanding Workloads

Many Interesting Things
Are Happening Today
in Computer Architecture

New (Device) Technologies

Many Interesting Things
Are Happening Today
in Computer Architecture

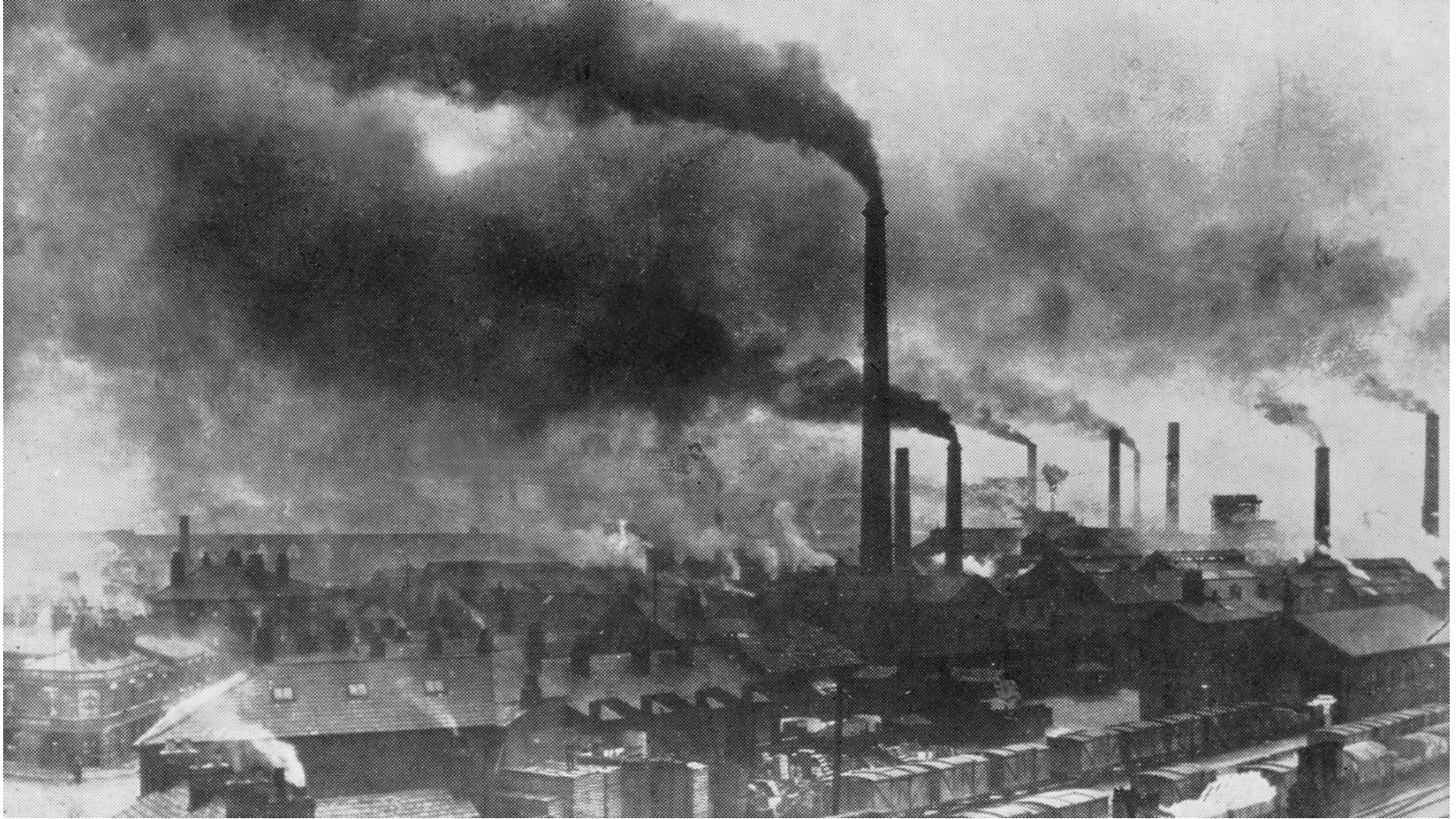
Many Interesting Things
Are Happening Today
in Computer Architecture

Performance
Energy Efficiency
Sustainability

Do We Want This?



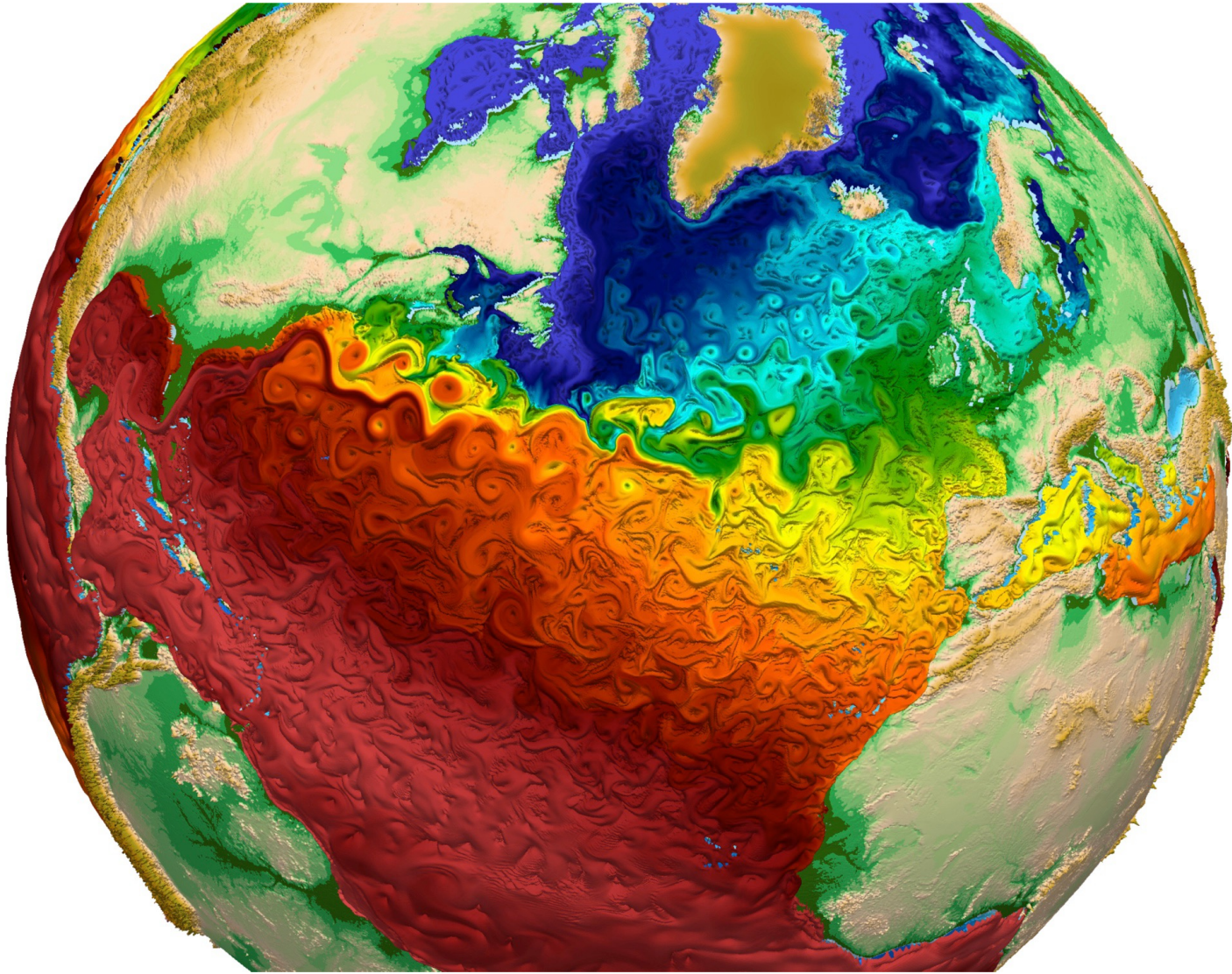
Or This?



Challenge and Opportunity for Future

High Performance,
Energy Efficient,
Sustainable

Many Difficult Problems: Climate



Many Difficult Problems: Intelligence



Many Difficult Problems: Intelligence

Forbes

Jun 17, 2020, 11:54am EDT | 20,934 views

Deep Learning's Carbon Emissions Problem



Rob Toews Contributor ⓘ

AI

I write about the big picture of artificial intelligence.



MIT
Technology
Review

Featured Topics Newsletters Events Podcasts

Sign in

Subscribe

ARTIFICIAL INTELLIGENCE

Training a single AI model can emit as much carbon as five cars in their lifetimes

Deep learning has a terrible carbon footprint.

By Karen Hao

June 6, 2019

Source: <http://spectrum.ieee.org/image/MjYzMzAyMg.jpeg>

Source: <https://www.forbes.com/sites/robtoews/2020/06/17/deep-learnings-climate-change-problem/>

Source: <https://www.technologyreview.com/2019/06/06/239031/training-a-single-ai-model-can-emit-as-much-carbon-as-five-cars-in-their-lifetimes/>

Many Difficult Problems: Congestion

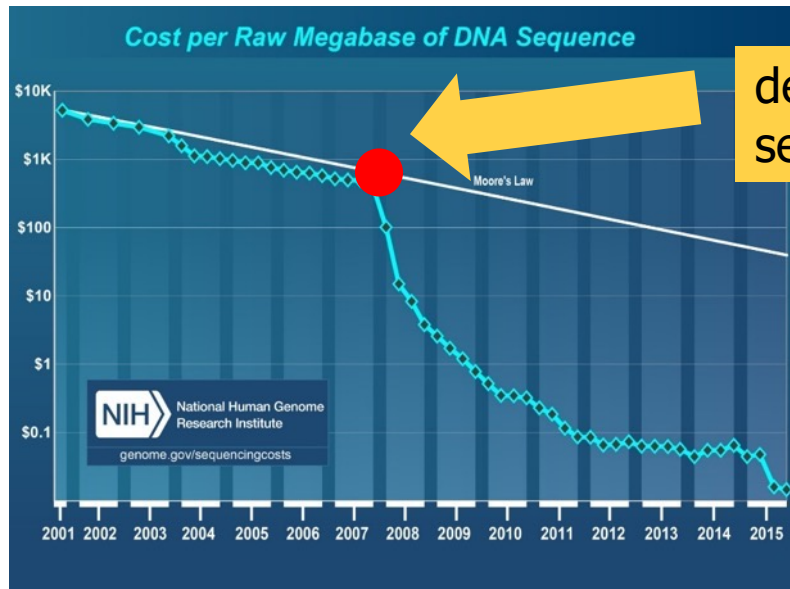


Many Difficult Problems: Public Health



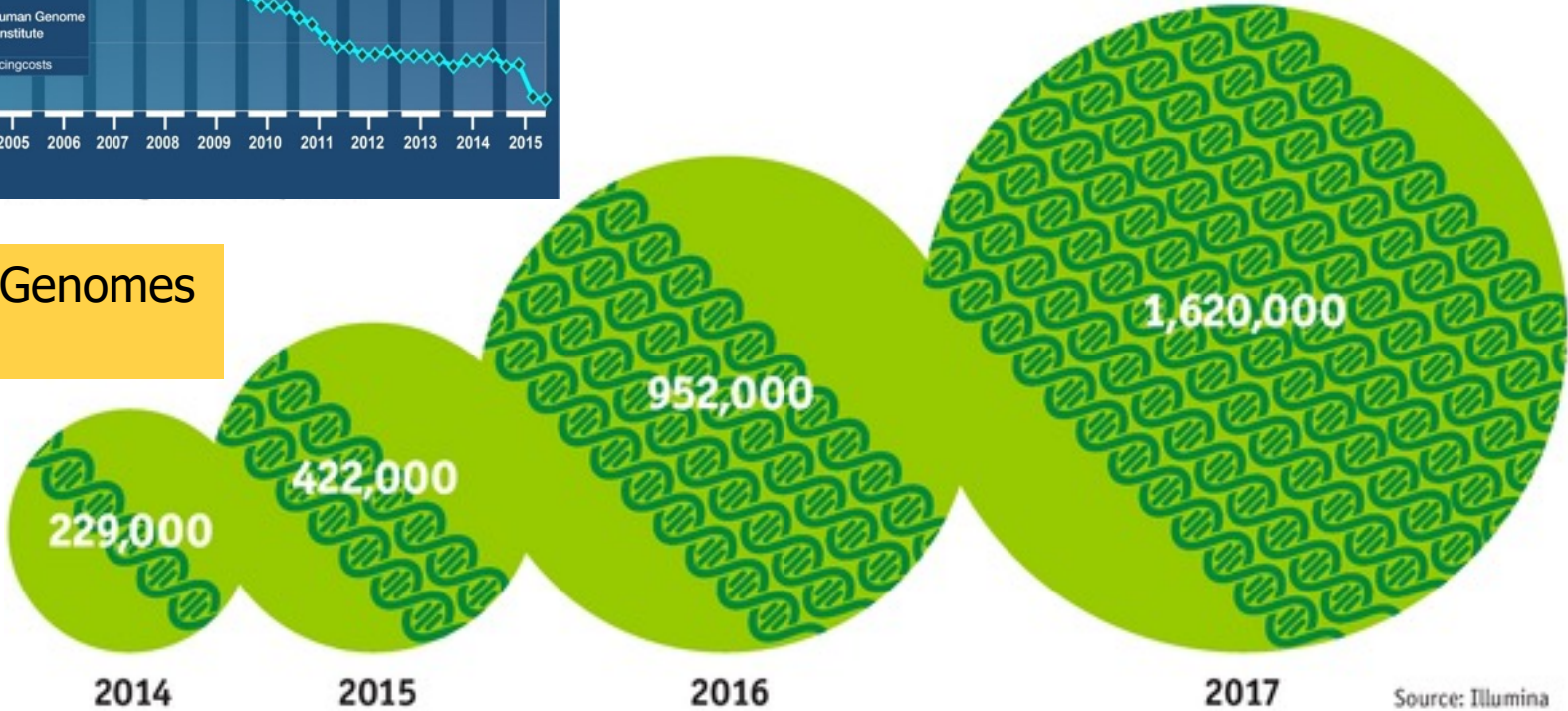
Source: <https://blog.wego.com/7-crowded-places-and-events-that-you-will-love/>

Many Difficult Problems: Genome Analysis



development of high-throughput sequencing (HTS) technologies

Number of Genomes Sequenced

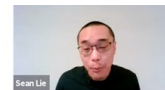


The Economist

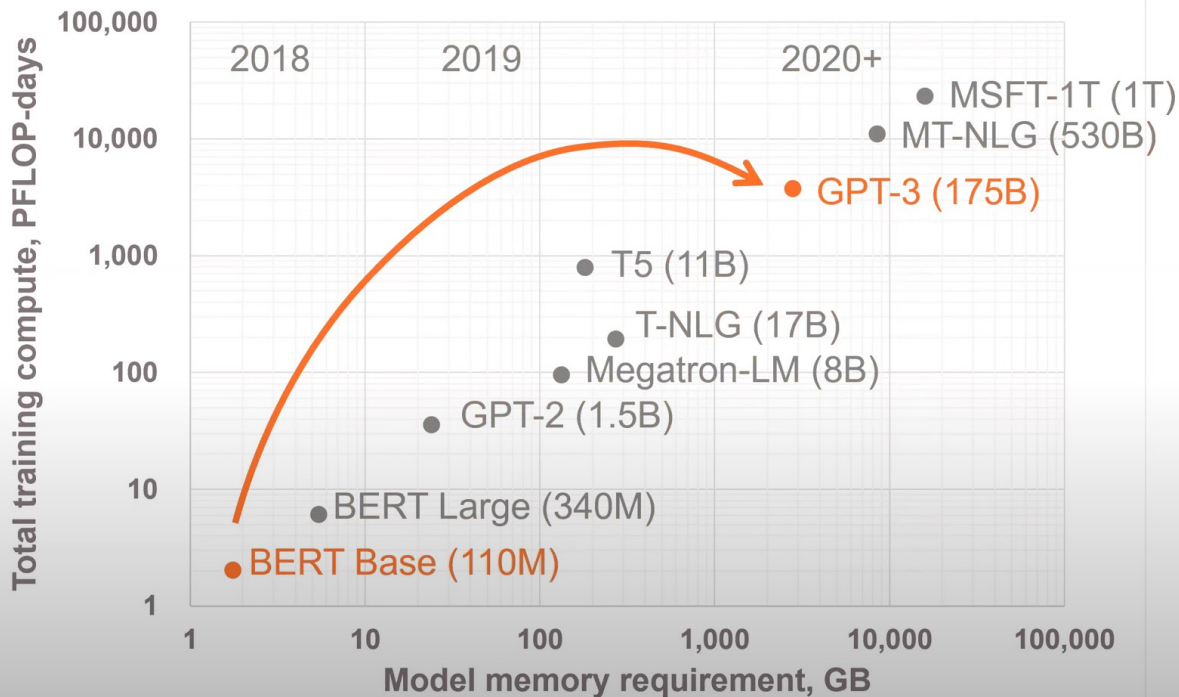
Source: Illumina

Huge Demand for Performance & Efficiency

Exponential Growth of Neural Networks



Memory and compute requirements



1800x more compute
In just 2 years

Tomorrow, **multi-trillion** parameter models

Computation vs. Data Storage Dichotomy

Sensors



SoC
with lots of
compute
& caches

Storage

Main Memory

Main Memory

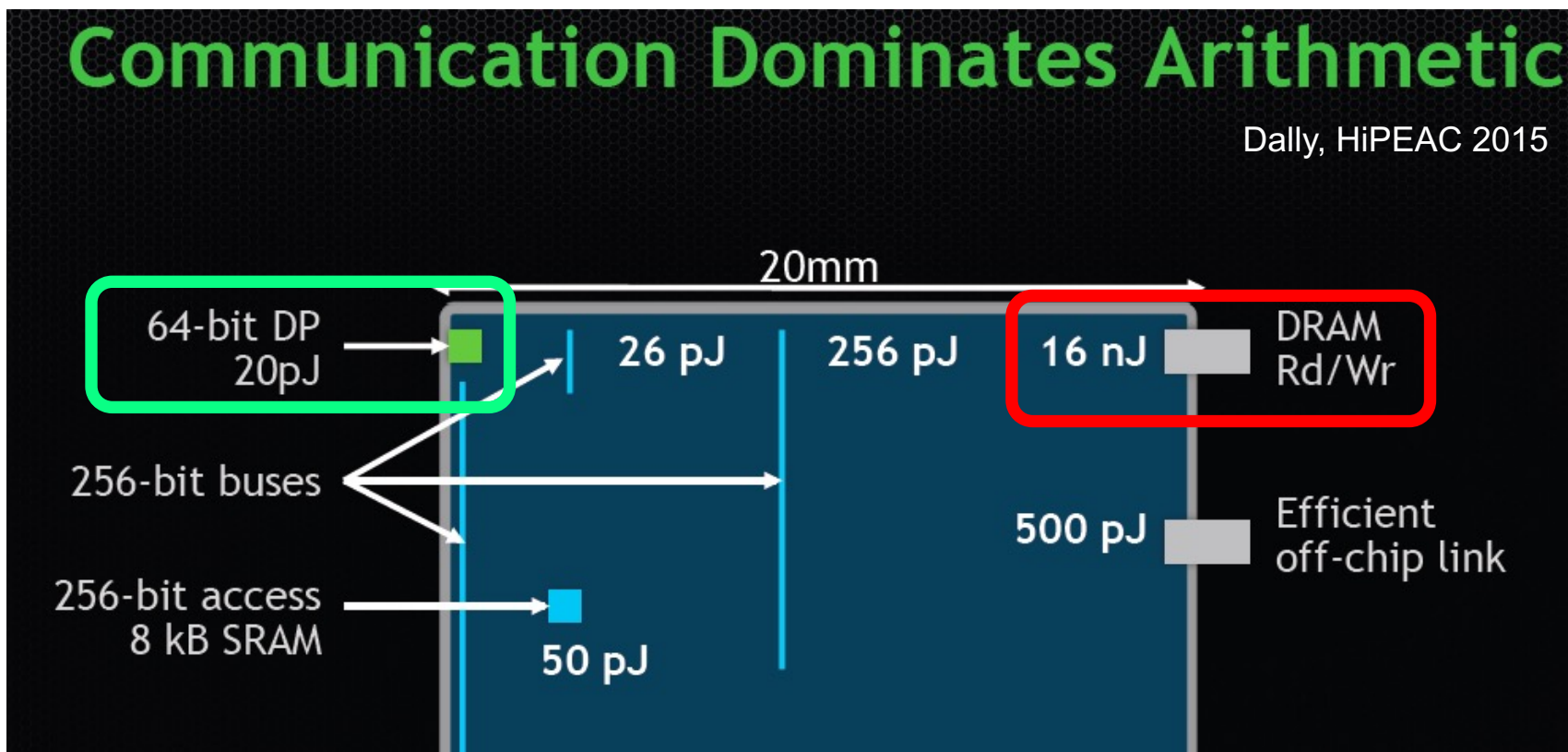
Storage

Apple M1 Ultra System (2022)

Data Movement vs. Computation Energy

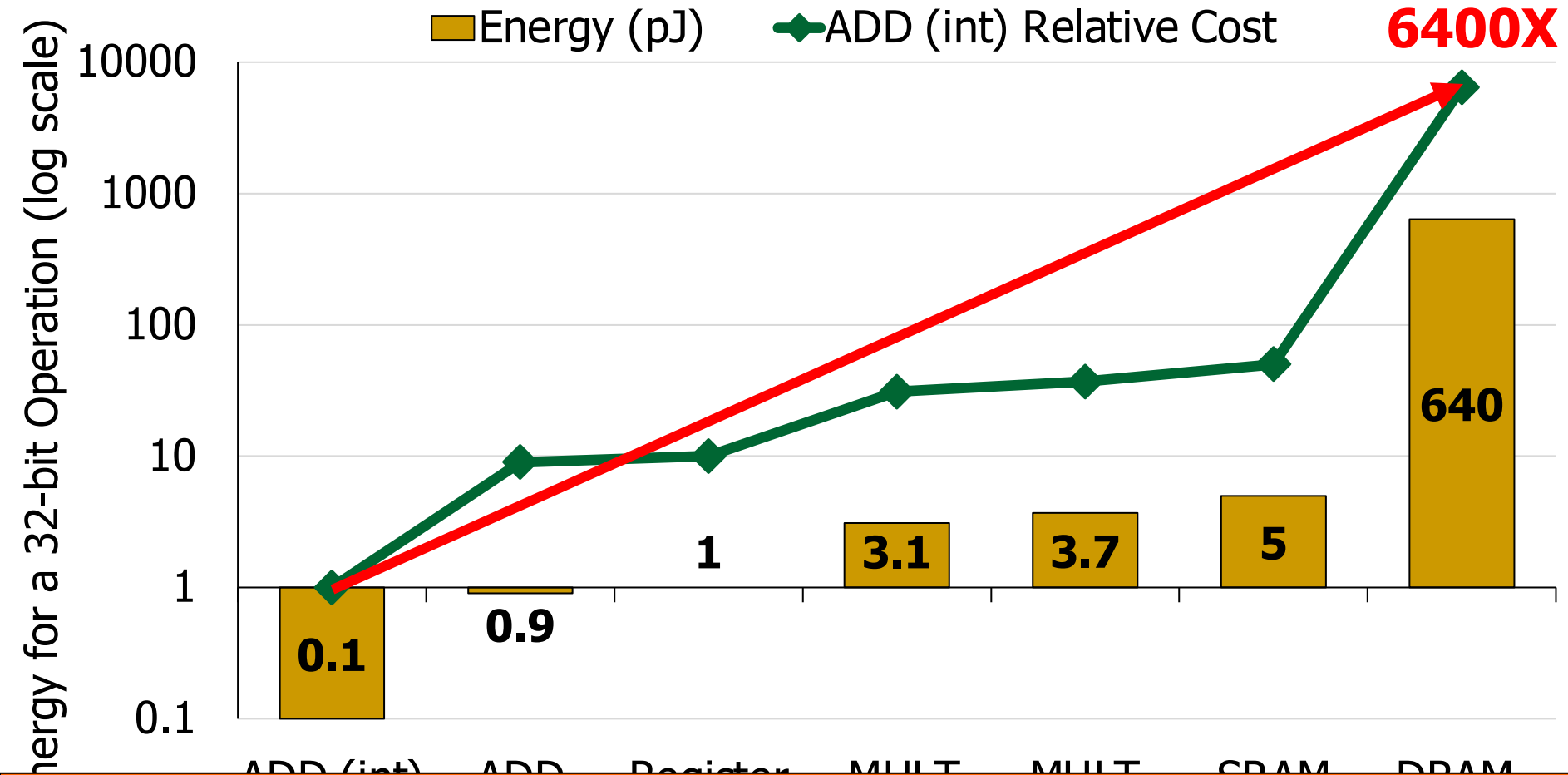
Communication Dominates Arithmetic

Dally, HiPEAC 2015



A memory access consumes $\sim 100-1000X$ the energy of a complex addition

Data Movement vs. Computation Energy

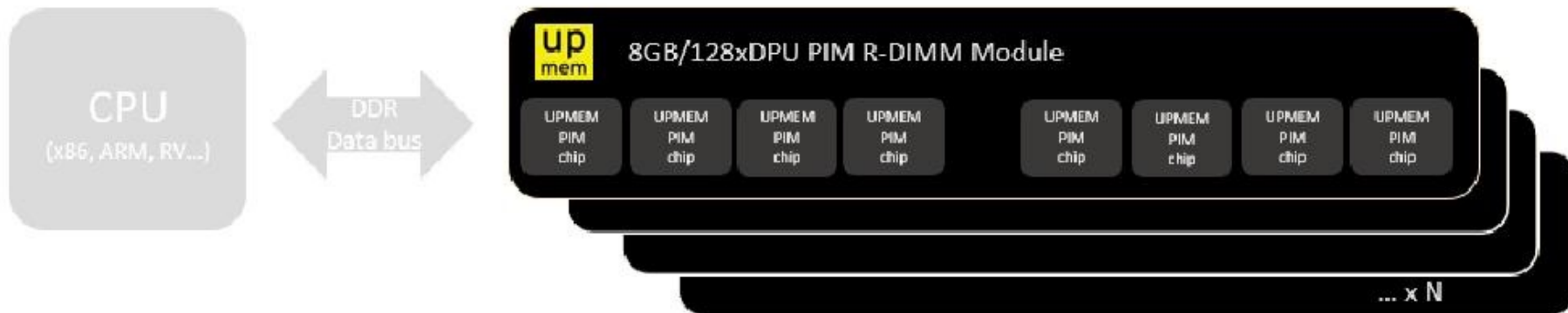


A memory access consumes 6400X the energy of a simple integer addition

Computing Architectures
with
Minimal Data Movement

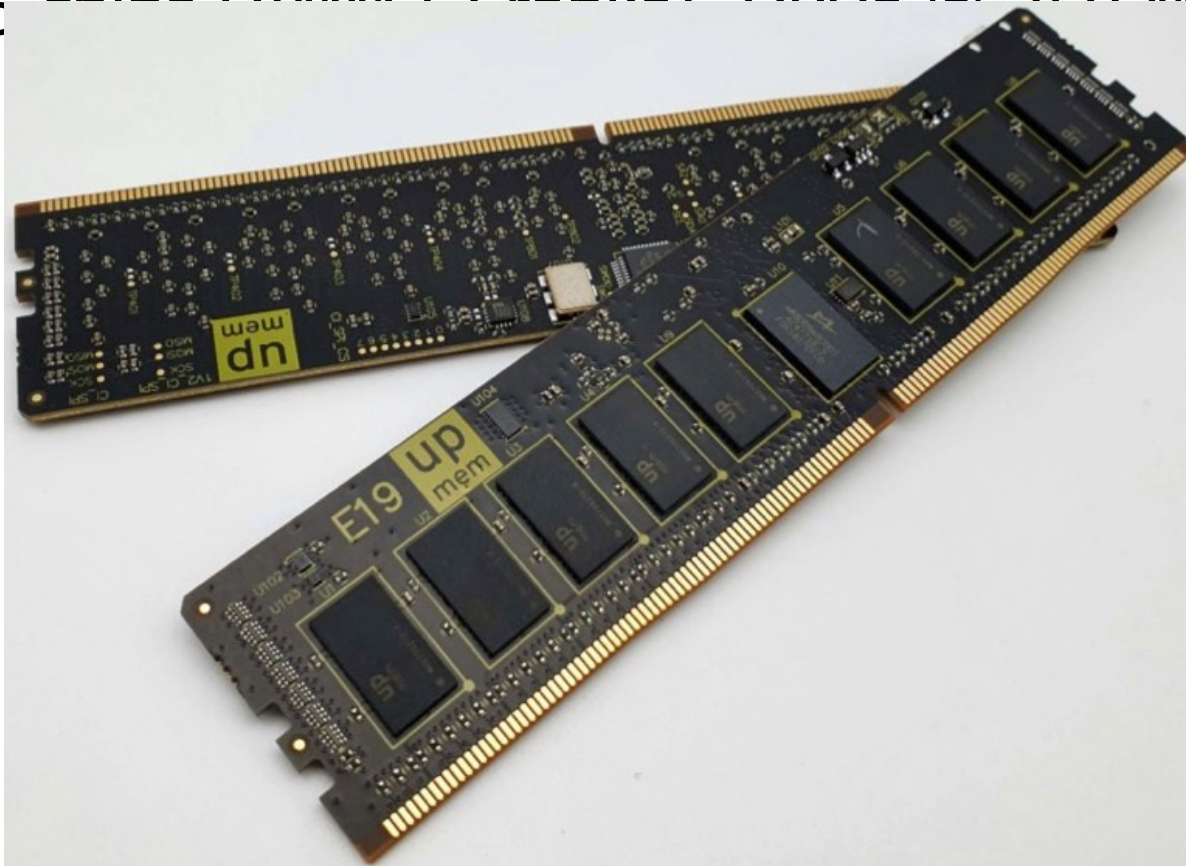
UPMEM Processing-in-DRAM Engine (2019)

- **Processing in DRAM Engine**
- Includes **standard DIMM modules**, with a **large number of DPU processors** combined with DRAM chips.
- Replaces **standard DIMMs**
 - DDR4 R-DIMM modules
 - 8GB+128 DPUs (16 PIM chips)
 - Standard 2x-nm DRAM process
 - **Large amounts of** compute & memory bandwidth

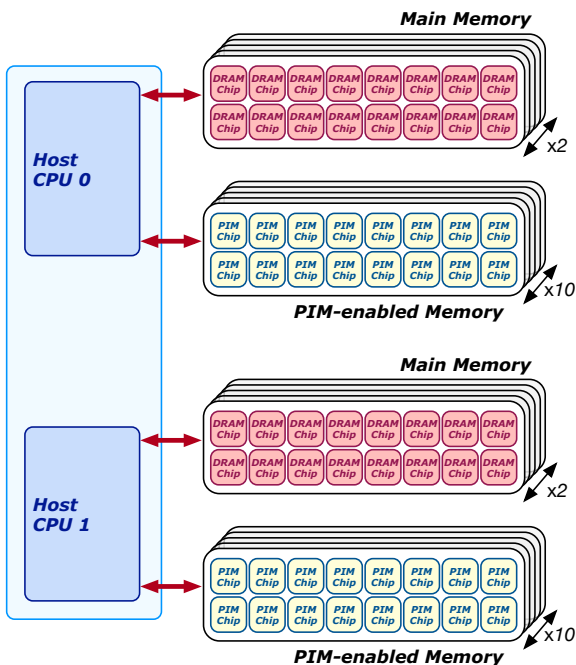


UPMEM Memory Modules

- E19: 8 chips DIMM (1 rank). DPUs @ 267 MHz
- P21: 16 chips DIMM (2 ranks). DPUs @ 250 MHz



2,560-DPU Processing-in-Memory System



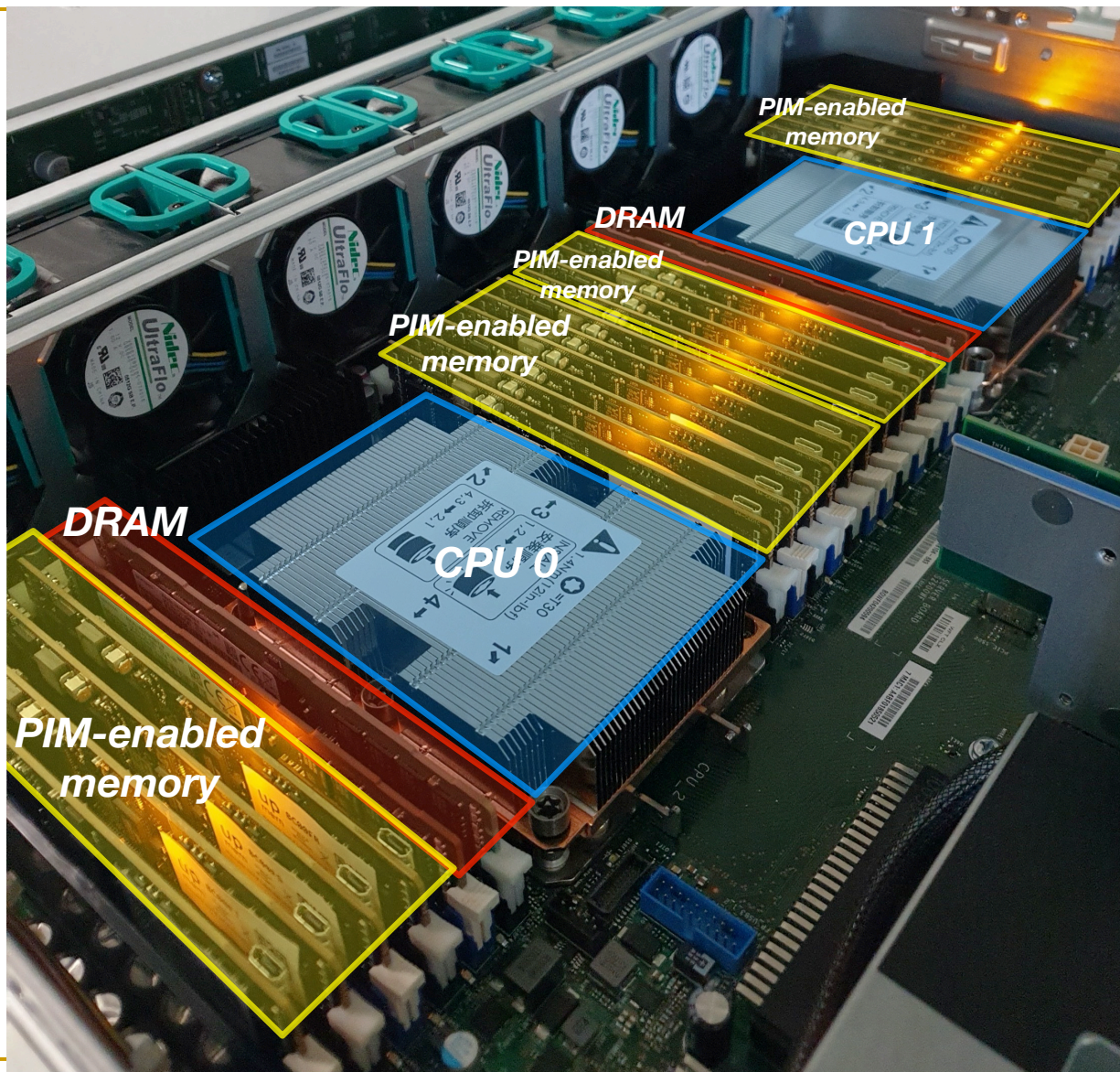
Benchmarking a New Paradigm: An Experimental Analysis of a Real Processing-in-Memory Architecture

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 440 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>

FPGA-based Processing Near Memory

- Gagandeep Singh, Mohammed Alser, Damla Senol Cali, Dionysios Diamantopoulos, Juan Gómez-Luna, Henk Corporaal, and Onur Mutlu, ["FPGA-based Near-Memory Acceleration of Modern Data-Intensive Applications"](#) *IEEE Micro (IEEE MICRO)*, to appear, 2021.

FPGA-based Near-Memory Acceleration of Modern Data-Intensive Applications

Gagandeep Singh[◇] Mohammed Alser[◇] Damla Senol Cali[✕]

Dionysios Diamantopoulos[▽] Juan Gómez-Luna[◇]

Henk Corporaal^{*} Onur Mutlu^{◇✕}

[◇]*ETH Zürich* [✕]*Carnegie Mellon University*

^{*}*Eindhoven University of Technology* [▽]*IBM Research Europe*

Samsung Function-in-Memory DRAM (2021)



Samsung Develops Industry's First High Bandwidth Memory with AI Processing Power

Korea on February 17, 2021

Audio



Share



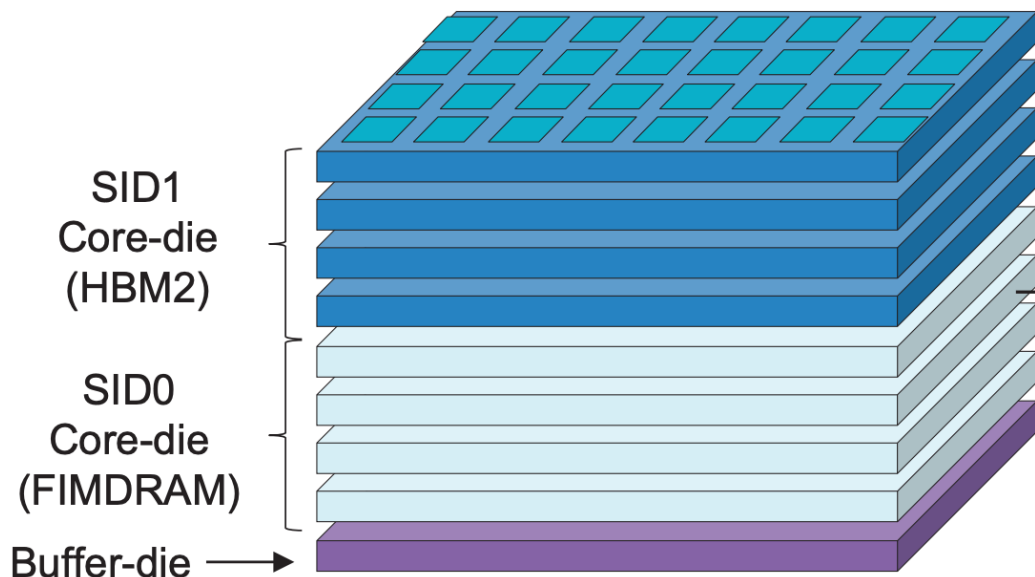
The new architecture will deliver over twice the system performance and reduce energy consumption by more than 70%

Samsung Electronics, the world leader in advanced memory technology, today announced that it has developed the industry's first High Bandwidth Memory (HBM) integrated with artificial intelligence (AI) processing power – the HBM-PIM. **The new processing-in-memory (PIM) architecture brings powerful AI computing capabilities inside high-performance memory, to accelerate large-scale processing in data centers, high performance computing (HPC) systems and AI-enabled mobile applications.**

Kwangil Park, senior vice president of Memory Product Planning at Samsung Electronics stated, "Our groundbreaking HBM-PIM is the industry's first programmable PIM solution tailored for diverse AI-driven workloads such as HPC, training and inference. We plan to build upon this breakthrough by further collaborating with AI solution providers for even more advanced PIM-powered applications."

Samsung Function-in-Memory DRAM (2021)

■ FIMDRAM based on HBM2



[3D Chip Structure of HBM with FIMDRAM]

Chip Specification

128DQ / 8CH / 16 banks / BL4

32 PCU blocks (1 FIM block/2 banks)

1.2 TFLOPS (4H)

**FP16 ADD /
Multiply (MUL) /
Multiply-Accumulate (MAC) /
Multiply-and- Add (MAD)**

ISSCC 2021 / SESSION 25 / DRAM / 25.4

25.4 A 20nm 6GB Function-In-Memory DRAM, Based on HBM2 with a 1.2TFLOPS Programmable Computing Unit Using Bank-Level Parallelism, for Machine Learning Applications

Young-Cheon Kwon¹, Suk Han Lee¹, Jaehoon Lee¹, Sang-Hyuk Kwon¹, Je Min Ryu¹, Jong-Pil Son¹, Seongil O¹, Hak-Soo Yu¹, Haesuk Lee¹, Soo Young Kim¹, Youngmin Cho¹, Jin Guk Kim¹, Jongyoon Choi¹, Hyun-Sung Shin¹, Jin Kim¹, BengSeng Phuah¹, HyoungMin Kim¹, Myeong Jun Song¹, Ahn Choi¹, Daeho Kim¹, SooYoung Kim¹, Eun-Bong Kim¹, David Wang², Shinhaeng Kang¹, Yuhwan Ro³, Seungwoo Seo³, JoonHo Song³, Jaeyoun Youn¹, Kyomin Sohn¹, Nam Sung Kim¹

¹Samsung Electronics, Hwaseong, Korea

²Samsung Electronics, San Jose, CA

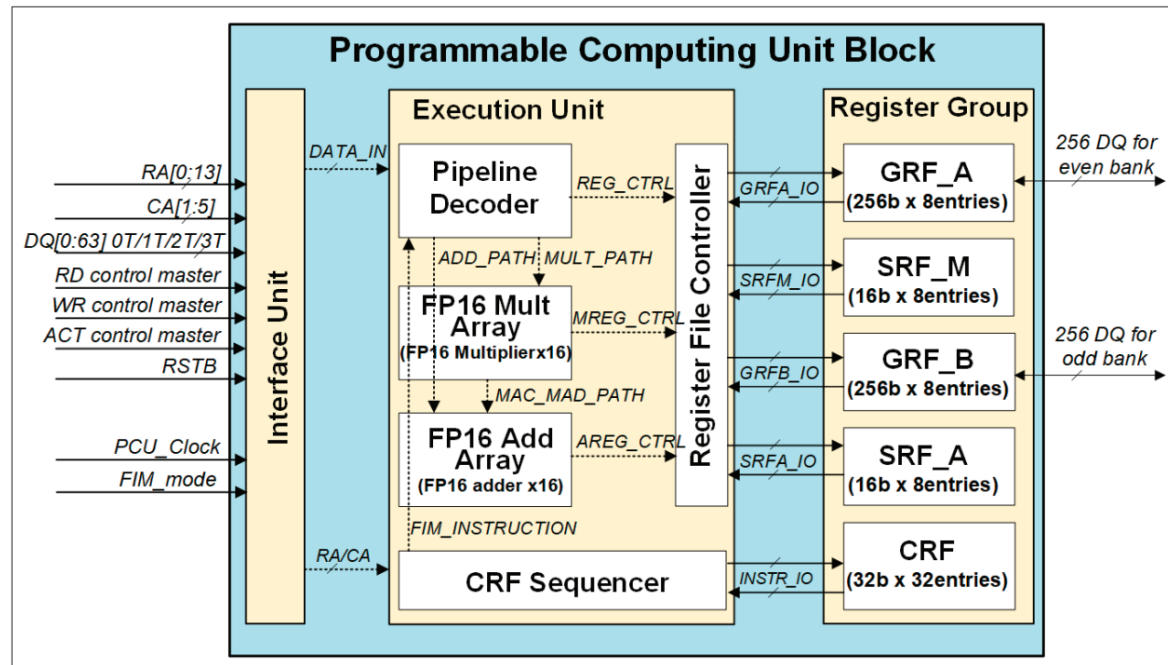
³Samsung Electronics, Suwon, Korea

Samsung Function-in-Memory DRAM (2021)

Programmable Computing Unit

■ Configuration of PCU block

- Interface unit to control data flow
- Execution unit to perform operations
- Register group
 - 32 entries of CRF for instruction memory
 - 16 GRF for weight and accumulation
 - 16 SRF to store constants for MAC operations



[Block diagram of PCU in FIMDRAM]

ISSCC 2021 / SESSION 25 / DRAM / 25.4

25.4 A 20nm 6GB Function-in-Memory DRAM, Based on HBM2 with a 1.2TFLOPS Programmable Computing Unit Using Bank-Level Parallelism, for Machine Learning Applications

Young-Cheon Kwon¹, Suk Han Lee¹, Jaehoon Lee¹, Sang-Hyuk Kwon¹, Je Min Ryu¹, Jong-Pil Son¹, Seongil O¹, Hak-Soo Yu¹, Haesuk Lee¹, Soo Young Kim¹, Youngmin Cho¹, Jin Guk Kim¹, Jongyoon Choi¹, Hyun-Sung Shin¹, Jin Kim¹, BengSeng Phuah¹, HyoungMin Kim¹, Myeong Jun Song¹, Ahn Choi¹, Daeho Kim¹, SooYoung Kim¹, Eun-Bong Kim¹, David Wang¹, Shinhaeng Kang¹, Yuhwan Ro¹, Seungwoo Seo¹, JoonHo Song¹, Jaeyoun Youn¹, Kyomin Sohn¹, Nam Sung Kim¹

¹Samsung Electronics, Hwasong, Korea
²Samsung Electronics, San Jose, CA
³Samsung Electronics, Suwon, Korea

Samsung Function-in-Memory DRAM (2021)

[Available instruction list for FIM operation]

Type	CMD	Description
Floating Point	ADD	FP16 addition
	MUL	FP16 multiplication
	MAC	FP16 multiply-accumulate
	MAD	FP16 multiply and add
Data Path	MOVE	Load or store data
	FILL	Copy data from bank to GRFs
Control Path	NOP	Do nothing
	JUMP	Jump instruction
	EXIT	Exit instruction

ISSCC 2021 / SESSION 25 / DRAM / 25.4

25.4 A 20nm 6GB Function-in-Memory DRAM, Based on HBM2 with a 1.2TFLOPS Programmable Computing Unit Using Bank-Level Parallelism, for Machine Learning Applications

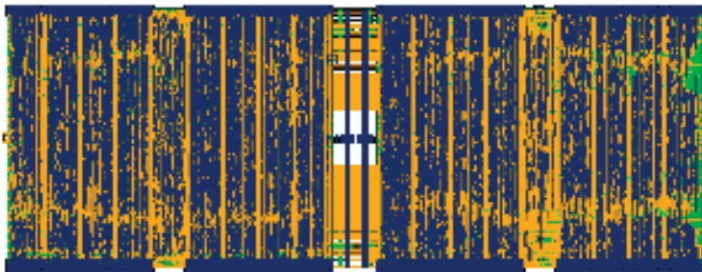
Young-Cheon Kwon¹, Suk Han Lee¹, Jaehoon Lee¹, Sang-Hyuk Kwon¹, Je Min Ryu¹, Jong-Pil Son¹, Seongil O¹, Hak-Soo Yu¹, Haesuk Lee¹, Soo Young Kim¹, Youngmin Cho¹, Jin Guk Kim¹, Jongyoon Choi¹, Hyun-Sung Shin¹, Jin Kim¹, BengSeng Phuah¹, HyoungMin Kim¹, Myeong Jun Song¹, Ahn Choi¹, Daeho Kim¹, SooYoung Kim¹, Eun-Bong Kim¹, David Wang¹, Shinhaeng Kang¹, Yuhwan Ro¹, Seungwoo Seo¹, JoonHo Song¹, Jaeyoun Youn¹, Kyomin Sohn¹, Nam Sung Kim¹

¹Samsung Electronics, Hwasong, Korea
²Samsung Electronics, San Jose, CA
³Samsung Electronics, Suwon, Korea

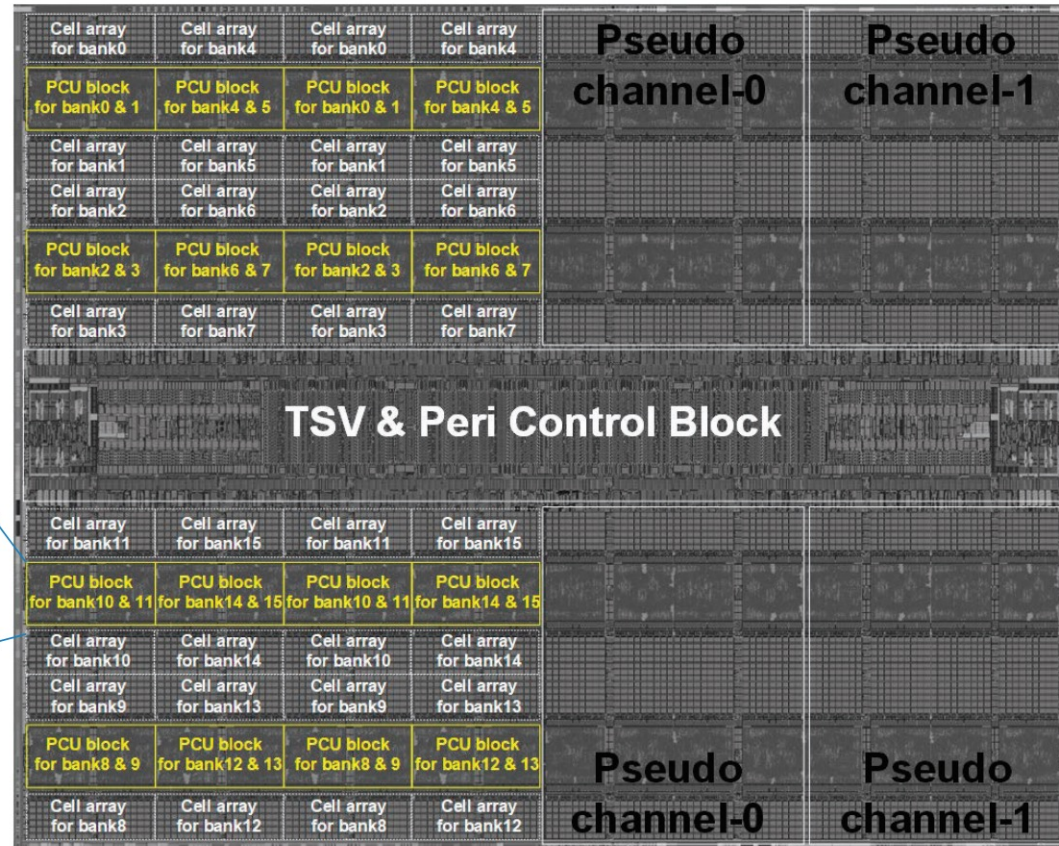
Samsung Function-in-Memory DRAM (2021)

Chip Implementation

- Mixed design methodology to implement FIMDRAM
 - Full-custom + Digital RTL



[Digital RTL design for PCU block]



ISSCC 2021 / SESSION 25 / DRAM / 25.4

25.4 A 20nm 6GB Function-In-Memory DRAM, Based on HBM2 with a 1.2TFLOPS Programmable Computing Unit Using Bank-Level Parallelism, for Machine Learning Applications

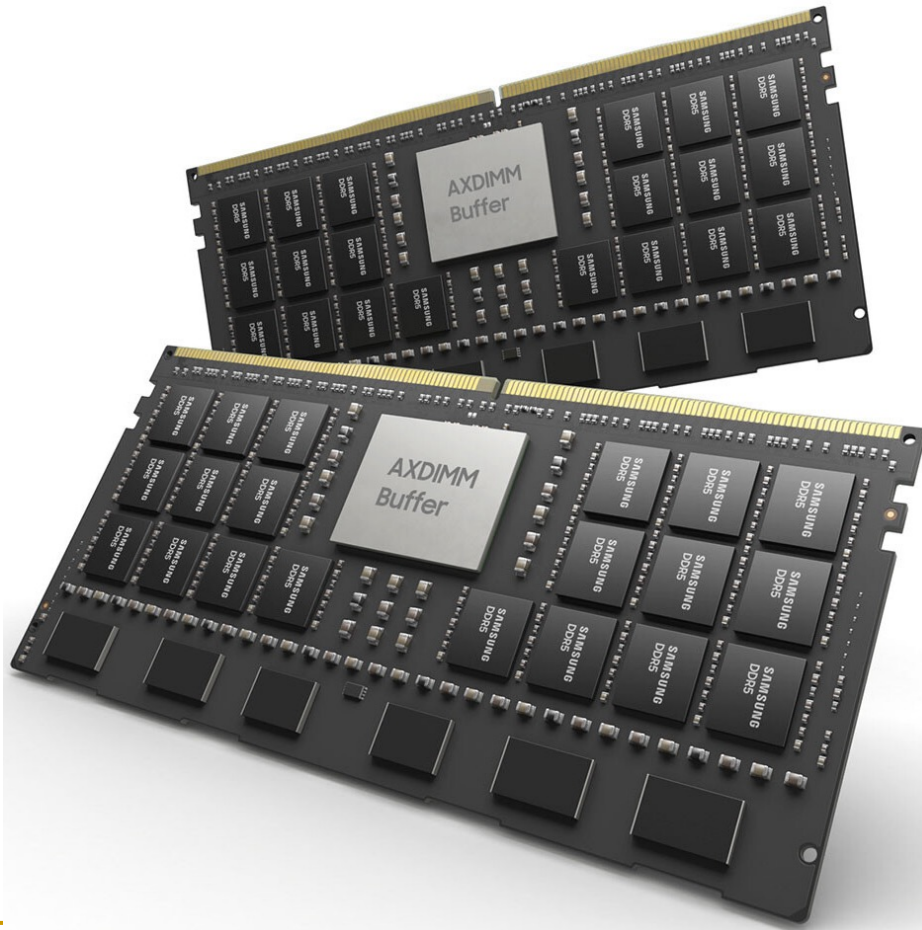
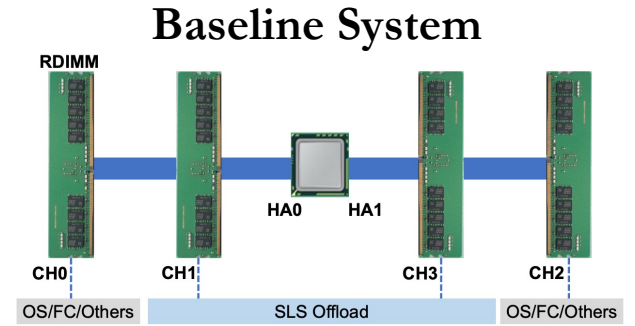
Young-Cheon Kwon¹, Suk Han Lee¹, Jaehoon Lee¹, Sang-Hyuk Kwon¹, Je Min Ryu¹, Jong-Pil Son¹, Seongil O¹, Hak-Soo Yu¹, Haesuk Lee¹, Soo Young Kim¹, Youngmin Cho¹, Jin Guk Kim¹, Jongyeon Choi¹, Hyun-Sung Shim¹, Jin Kim¹, BengSeng Phuah¹, HyoungMin Kim¹, Myeong Jun Song¹, Ahn Choi¹, Daeho Kim¹, SooYoung Kim¹, Eun-Bong Kim¹, David Wang², Shrinraeng Kang³, Yulwan Ro³, Seungwoo Seo³, JoonHo Song³, Jaeyoun Yoon¹, Kyomin Sohn¹, Nam Sung Kim¹

¹Samsung Electronics, Hwaseong, Korea
²Samsung Electronics, San Jose, CA
³Samsung Electronics, Suwon, Korea

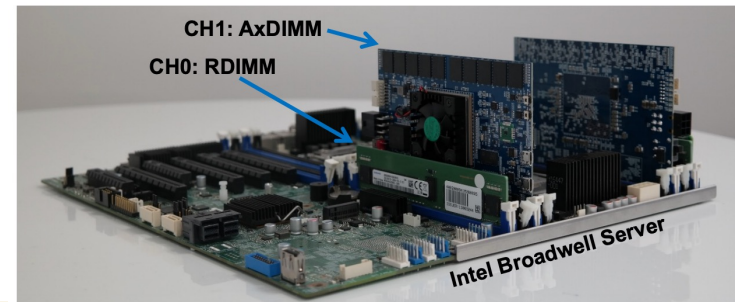
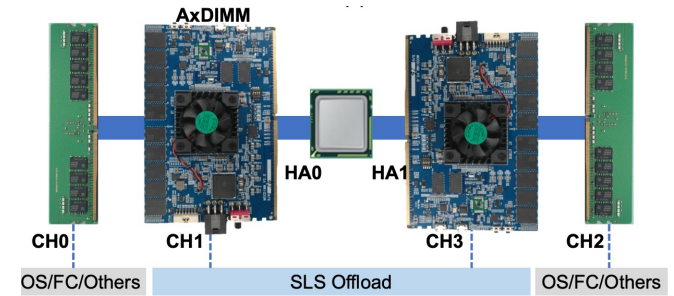
Samsung AxDIMM (2021)

- DDRx-PIM

Baseline system



AxDIMM System



SK Hynix Accelerator-in-Memory (2022)

SK hynix Develops PIM, Next-Generation AI Accelerator

February 16, 2022



Seoul, February 16, 2022

SK hynix (or “the Company”, www.skhynix.com) announced on February 16 that it has developed PIM*, a next-generation memory chip with computing capabilities.

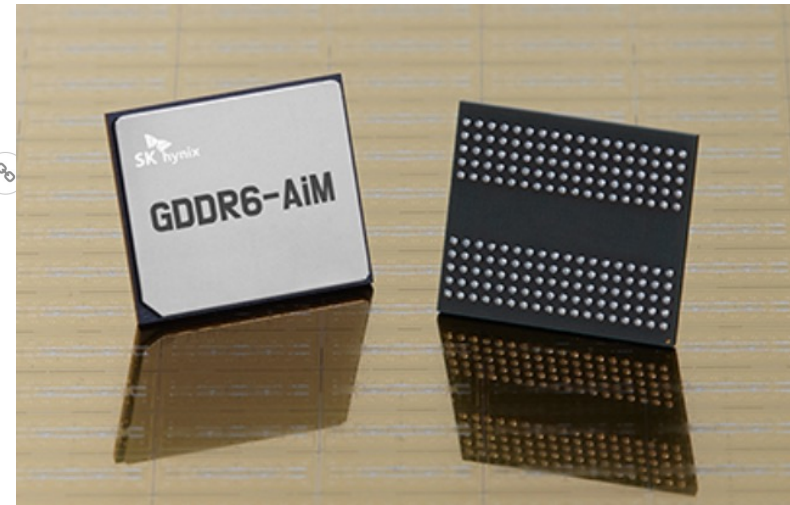
**PIM(Processing In Memory): A next-generation technology that provides a solution for data congestion issues for AI and big data by adding computational functions to semiconductor memory*

It has been generally accepted that memory chips store data and CPU or GPU, like human brain, process data. SK hynix, following its challenge to such notion and efforts to pursue innovation in the next-generation smart memory, has found a breakthrough solution with the development of the latest technology.

SK hynix plans to showcase its PIM development at the world’s most prestigious semiconductor conference, 2022 ISSCC*, in San Francisco at the end of this month. The company expects continued efforts for innovation of this technology to bring the memory-centric computing, in which semiconductor memory plays a central role, a step closer to the reality in devices such as smartphones.

**ISSCC: The International Solid-State Circuits Conference will be held virtually from Feb. 20 to Feb. 24 this year with a theme of “Intelligent Silicon for a Sustainable World”*

For the first product that adopts the PIM technology, SK hynix has developed a sample of GDDR6-AiM (Accelerator* in memory). The GDDR6-AiM adds computational functions to GDDR6* memory chips, which process data at 16Gbps. A combination of GDDR6-AiM with CPU or GPU instead of a typical DRAM makes certain computation speed 16 times faster. GDDR6-AiM is widely expected to be adopted for machine learning, high-performance computing, and big data computation and storage.



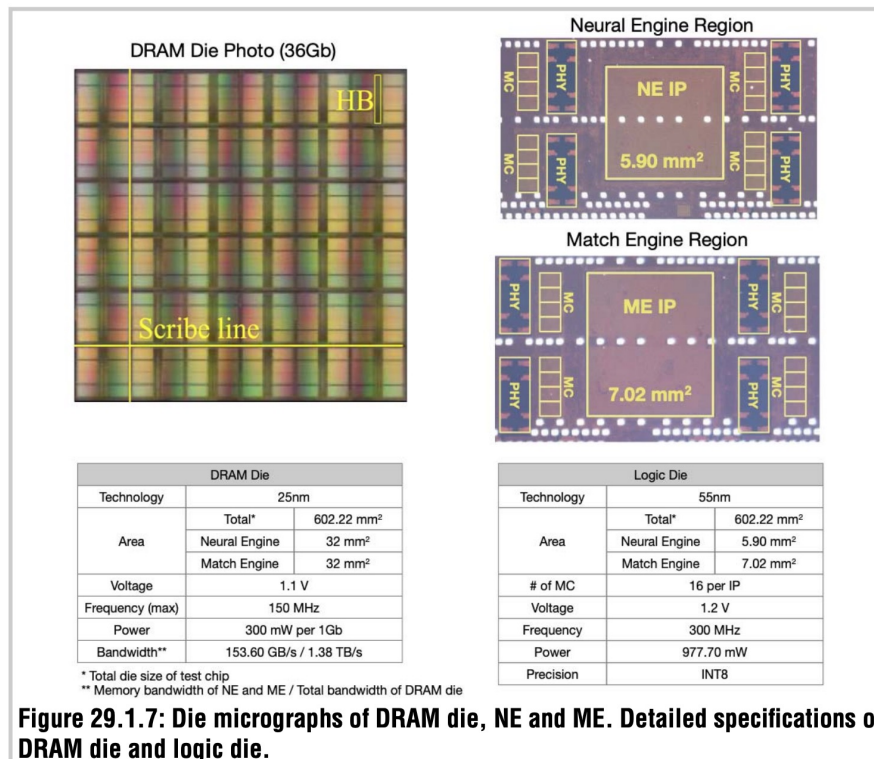
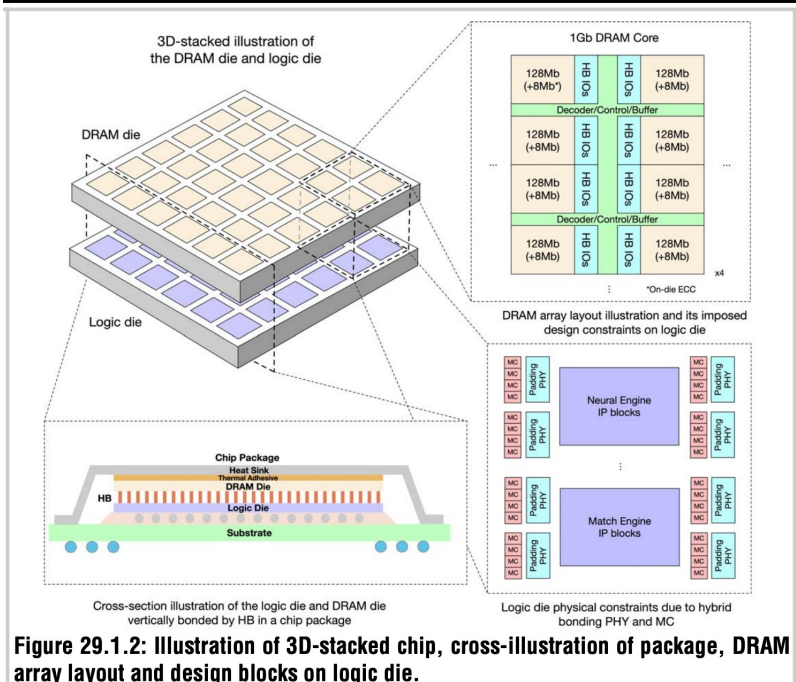
11.1 A 1ynm 1.25V 8Gb, 16Gb/s/pin GDDR6-based Accelerator-in-Memory supporting 1TFLOPS MAC Operation and Various Activation Functions for Deep-Learning Applications

Seongju Lee, SK hynix, Icheon, Korea

In Paper 11.1, SK Hynix describes a 1ynm, GDDR6-based accelerator-in-memory with a command set for deep-learning operation. The 8Gb design achieves a peak throughput of 1TFLOPS with 1GHz MAC operations and supports major activation functions to improve accuracy.

AliBaba PIM Recommendation System (2022)

ISSCC 2022 / February 24, 2022 / 8:30 AM



29.1 184QPS/W 64Mb/mm² 3D Logic-to-DRAM Hybrid Bonding with Process-Near-Memory Engine for Recommendation System

Dimin Niu¹, Shuangchen Li¹, Yuhao Wang¹, Wei Han¹, Zhe Zhang², Yijin Guan², Tianchan Guan³, Fei Sun¹, Fei Xue¹, Lide Duan¹, Yuanwei Fang¹, Hongzhong Zheng¹, Xiping Jiang⁴, Song Wang⁴, Fengguo Zuo⁴, Yubing Wang⁴, Bing Yu⁴, Qiwei Ren⁴, Yuan Xie¹

PIM Review and Open Problems

A Modern Primer on Processing in Memory

Onur Mutlu^{a,b}, Saugata Ghose^{b,c}, Juan Gómez-Luna^a, Rachata Ausavarungnirun^d

SAFARI Research Group

^a*ETH Zürich*

^b*Carnegie Mellon University*

^c*University of Illinois at Urbana-Champaign*

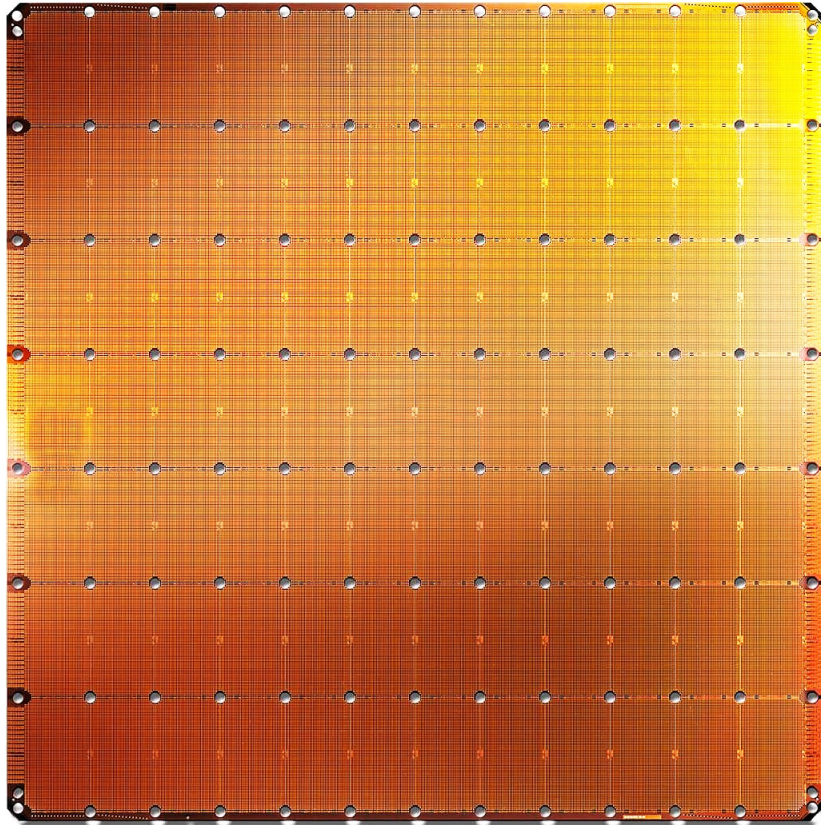
^d*King Mongkut's University of Technology North Bangkok*

Onur Mutlu, Saugata Ghose, Juan Gomez-Luna, and Rachata Ausavarungnirun,

"A Modern Primer on Processing in Memory"

*Invited Book Chapter in **Emerging Computing: From Devices to Systems - Looking Beyond Moore and Von Neumann**, Springer, to be published in 2021.*

Cerebras's Wafer Scale ML Engine (2019)



Cerebras WSE

1.2 Trillion transistors
46,225 mm²

- The largest ML accelerator chip
- 400,000 cores



Largest GPU

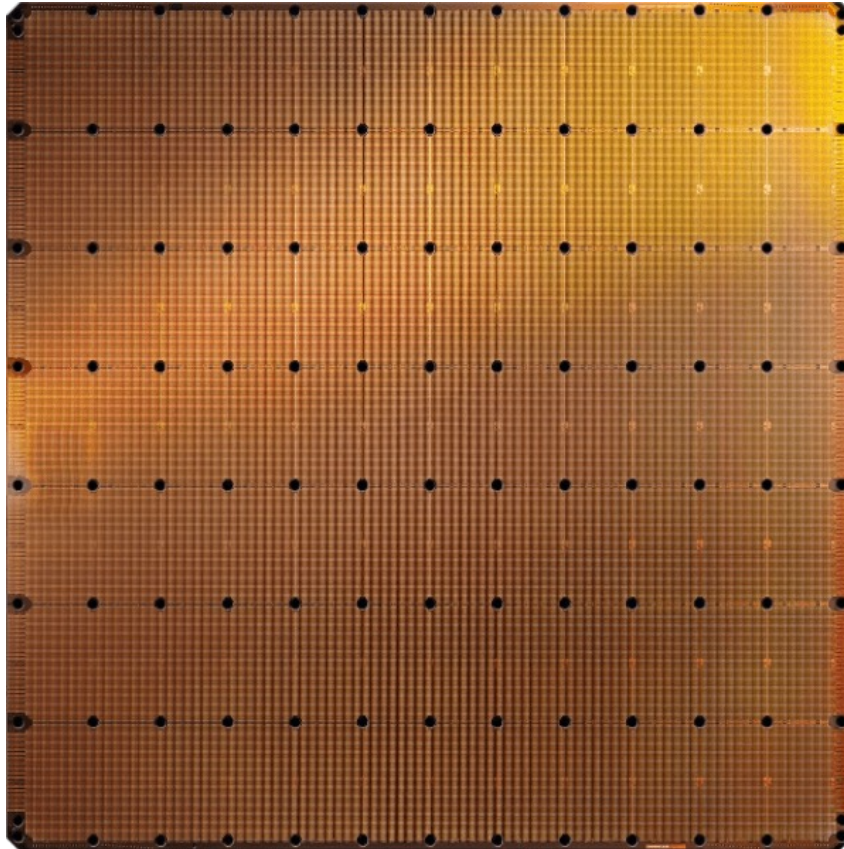
21.1 Billion transistors
815 mm²

NVIDIA TITAN V

<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 ML Engine-2 (2021)



Cerebras WSE-2
2.6 Trillion transistors
46,225 mm²

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



Largest GPU
54.2 Billion transistors
826 mm²
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/>

Computing Architectures
with
Minimal Data Movement

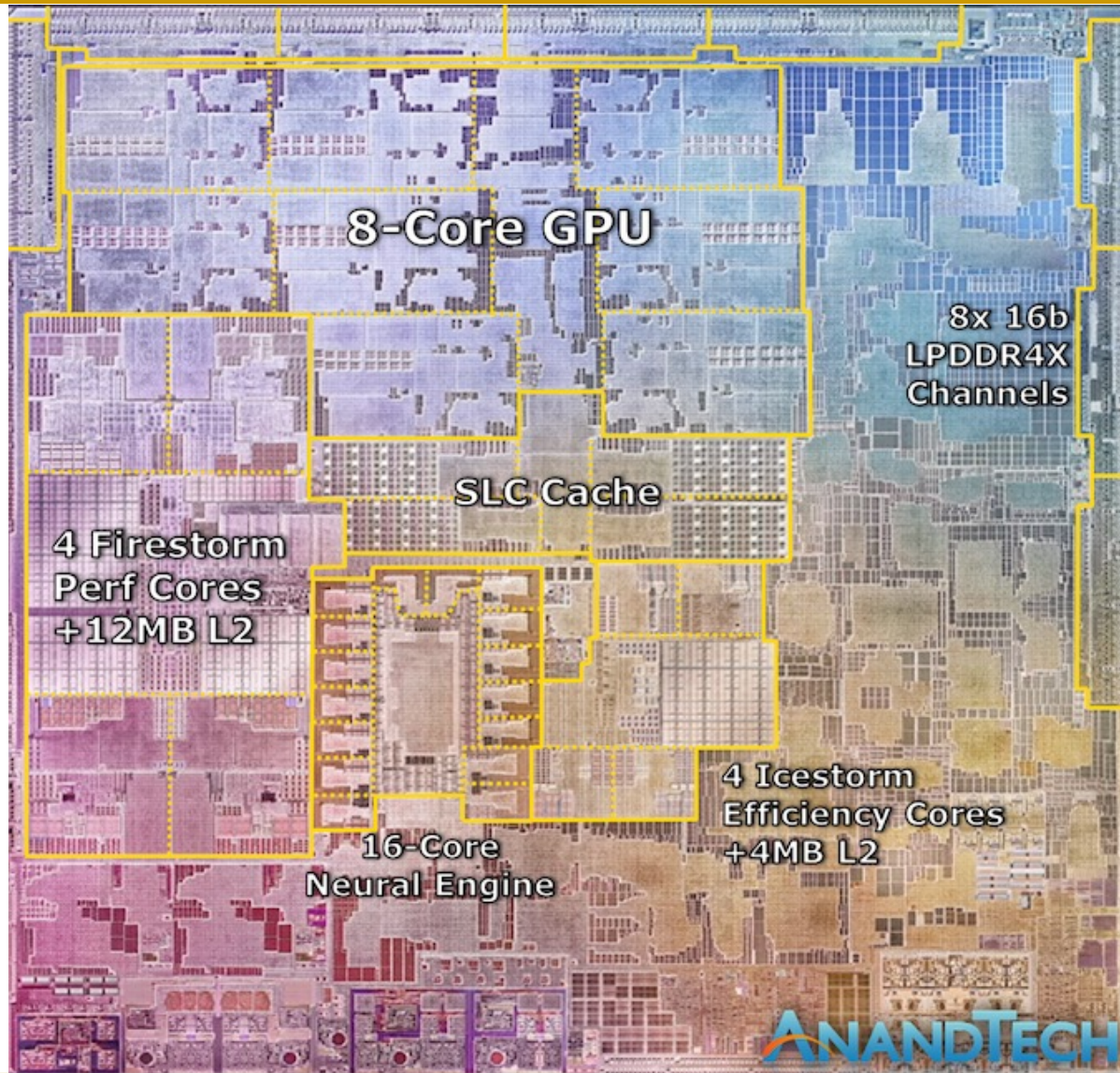
Fundamentally
Energy-Efficient
(Data-Centric)
Computing Architectures

Fundamentally High-Performance **(Data-Centric)** Computing Architectures

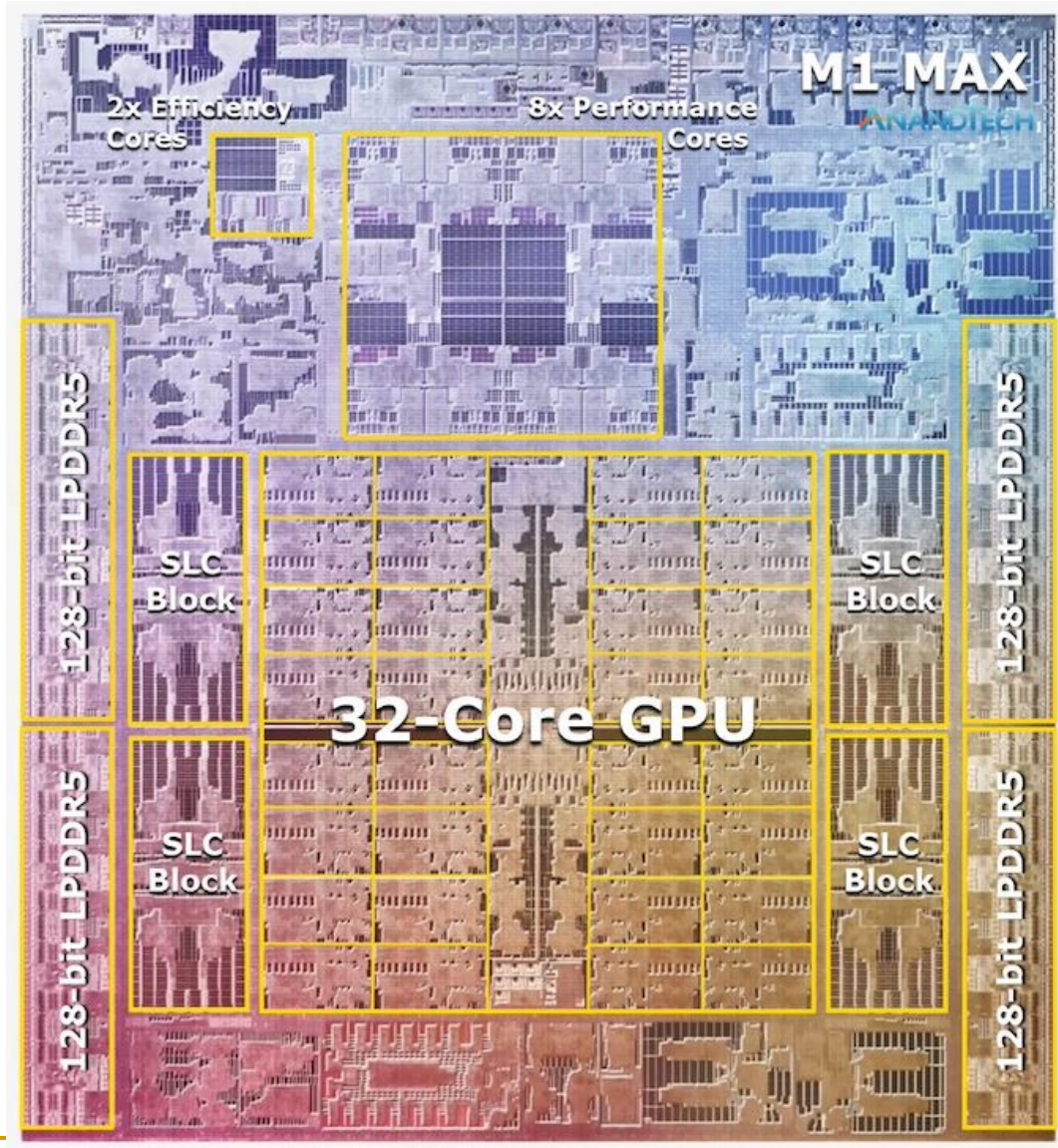
Many Interesting Things
Are Happening Today
in Computer Architecture

Performance
Energy Efficiency
Sustainability
Specialized Accelerators

Apple M1 System on Chip (2021)



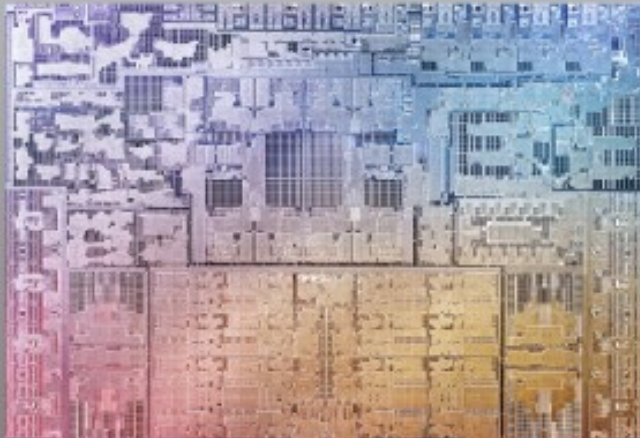
Apple M1 Max System on Chip (2021)



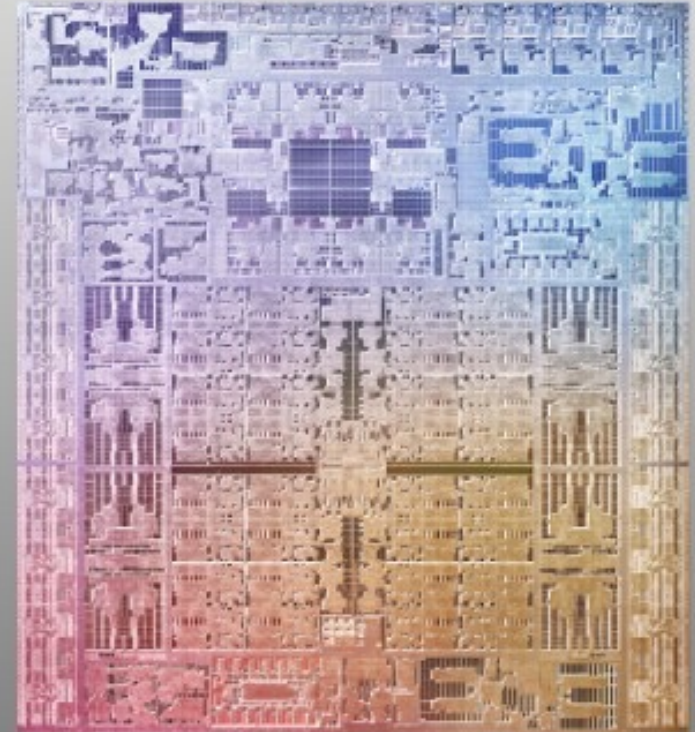
Bigger and More Powerful Systems (2021)



Apple M1

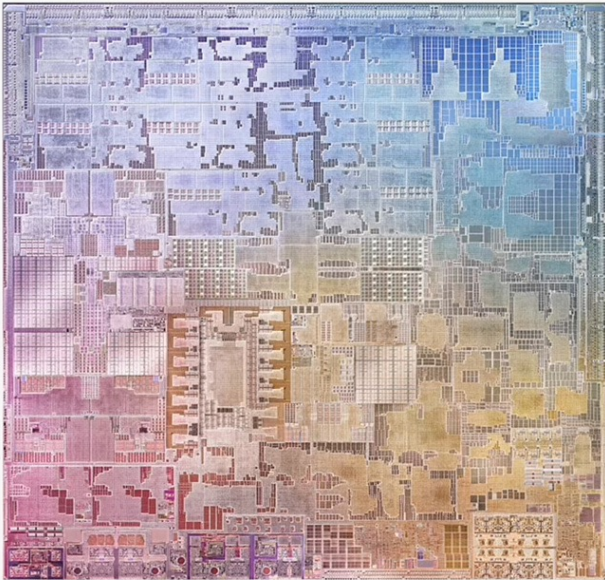


Apple M1 Pro

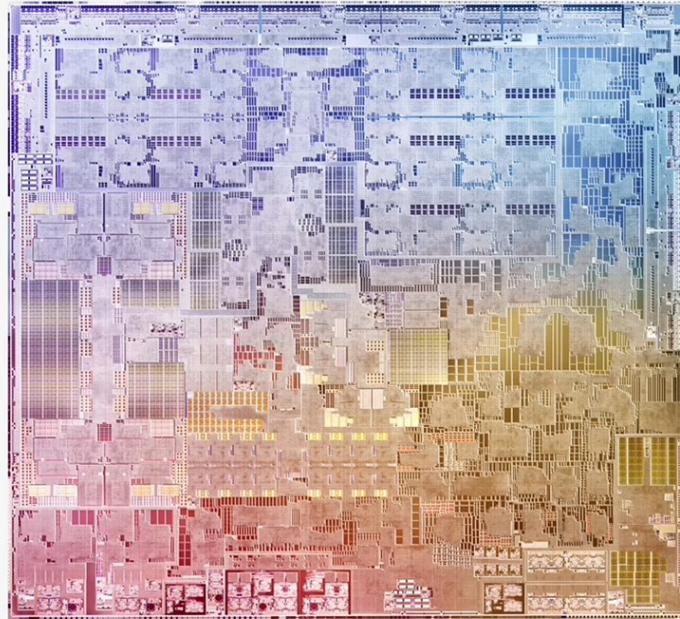


Apple M1 Max

Bigger and More Powerful Systems (2022)



Apple M1

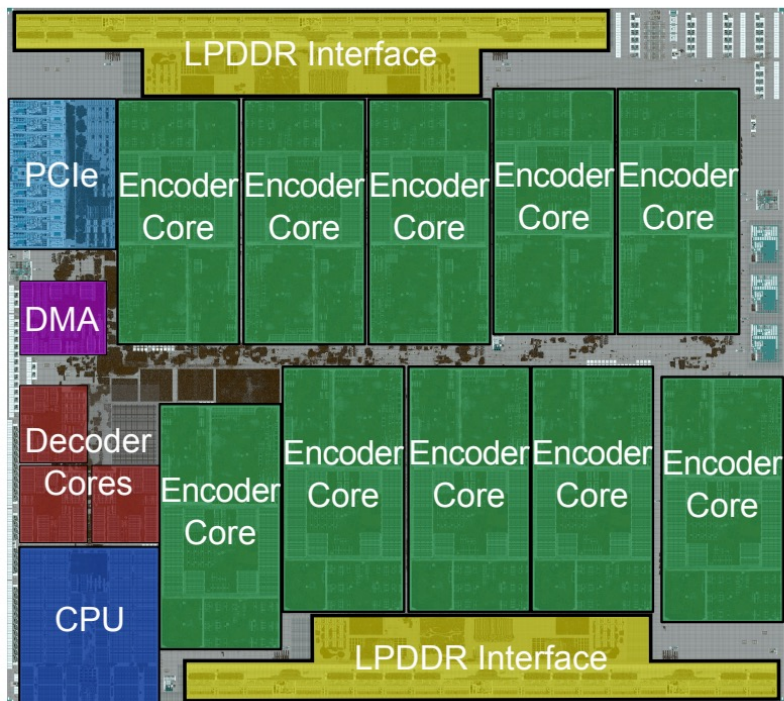


Apple M2

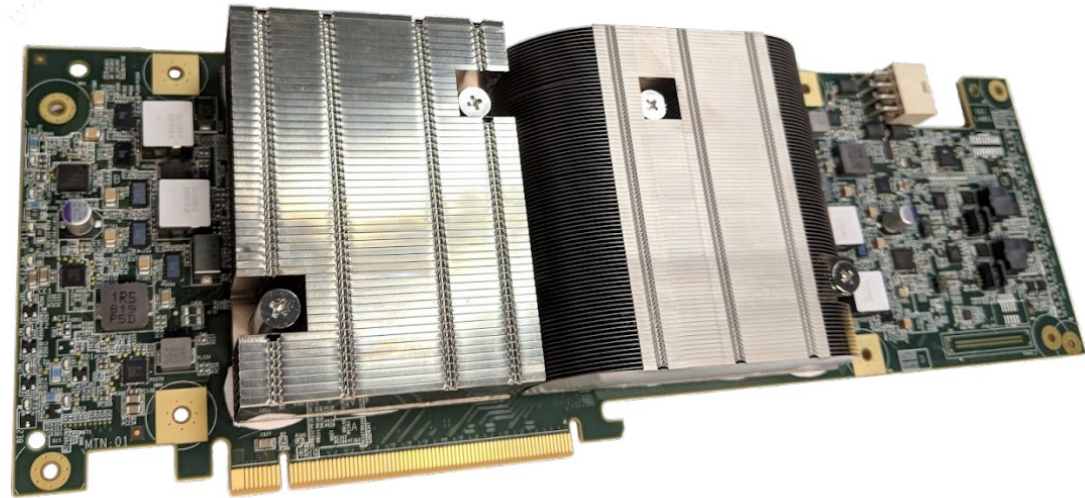
Apple Silicon SoCs		
SoC	M2	M1
CPU	4x High Performance (Avalanche?) 16MB Shared L2	4x High Performance (Firestorm) 12MB Shared L2
	4x High Efficiency (Blizzard?) 4MB Shared L2	4x High Efficiency (Icestorm) 4MB Shared L2
GPU	"Next Generation" 10-Core 3.6 TFLOPS	8-Core 2.6 TFLOPS
Neural Engine	16-Core 15.8 TOPS	16-Core 11 TOPS
Memory Controller	LPDDR5-6400 8x 16-bit CH 100GB/sec Total Bandwidth (Unified)	LPDDR4-4266 8x 16 CH 68GB/sec Total Bandwidth (Unified)
Memory Capacity	24GB	16GB
Encode/ Decode	8K H.264, H.265, ProRes, ProRes RAW	4K H.264, H.265
USB	USB4/Thunderbolt 3 2x Ports	USB4/Thunderbolt 3 2x Ports
Transistors	20 Billion	16 Billion
Mfc. Process	"Second Generation 5nm" TSMC N5P?	TSMC N5

Google's Video Coding Unit (2021)

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

Google's Video Coding Unit (2021)

I WONDER IF NETFLIX WANTS TO BUY SOME —

YouTube is now building its own video-transcoding chips

Google throws custom silicon at YouTube's massive video-transcoding workload.

RON AMADEO - 4/22/2021, 8:24 PM

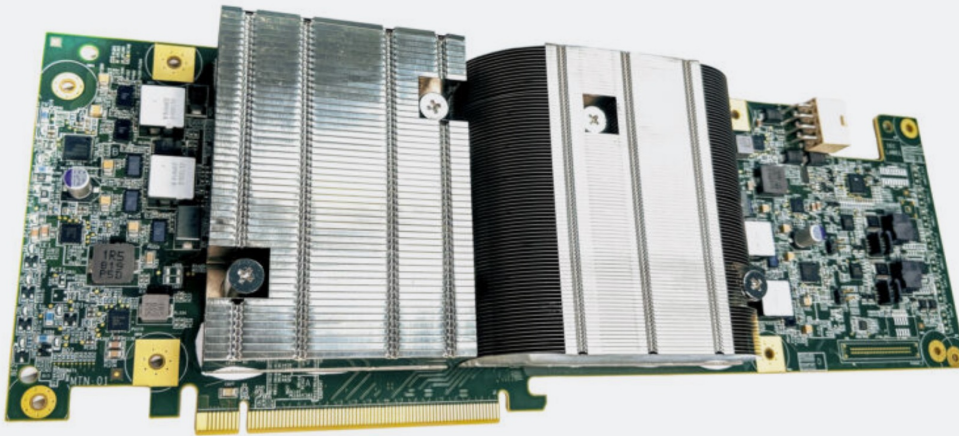


Table 1: Offline two-pass single output (SOT) throughput in VCU vs. CPU and GPU systems

System	Throughput [Mpix/s]		Perf/TCO ⁸	
	H.264	VP9	H.264	VP9
Skylake	714	154	1.0x	1.0x
4xNvidia T4	2,484	—	1.5x	—
8xVCU	5,973	6,122	4.4x	20.8x
20xVCU	14,932	15,306	7.0x	33.3x

Encoding Throughput: Table 1 shows throughput and perf/TCO (performance per total cost of ownership) for the four systems and is normalized to the perf/TCO of the CPU system. The performance is shown for offline two-pass SOT encoding for H.264 and VP9. For H.264, the GPU has 3.5x higher throughput, and the 8xVCU and 20xVCU provide 8.4x and 20.9x more throughput, respectively. For VP9, the 20xVCU system has 99.4x the throughput of the CPU baseline. The two orders of magnitude increase in performance clearly demonstrates the benefits of our VCU system.

Enlarge / A Google Argos VCU. It transcodes video very quickly.

116

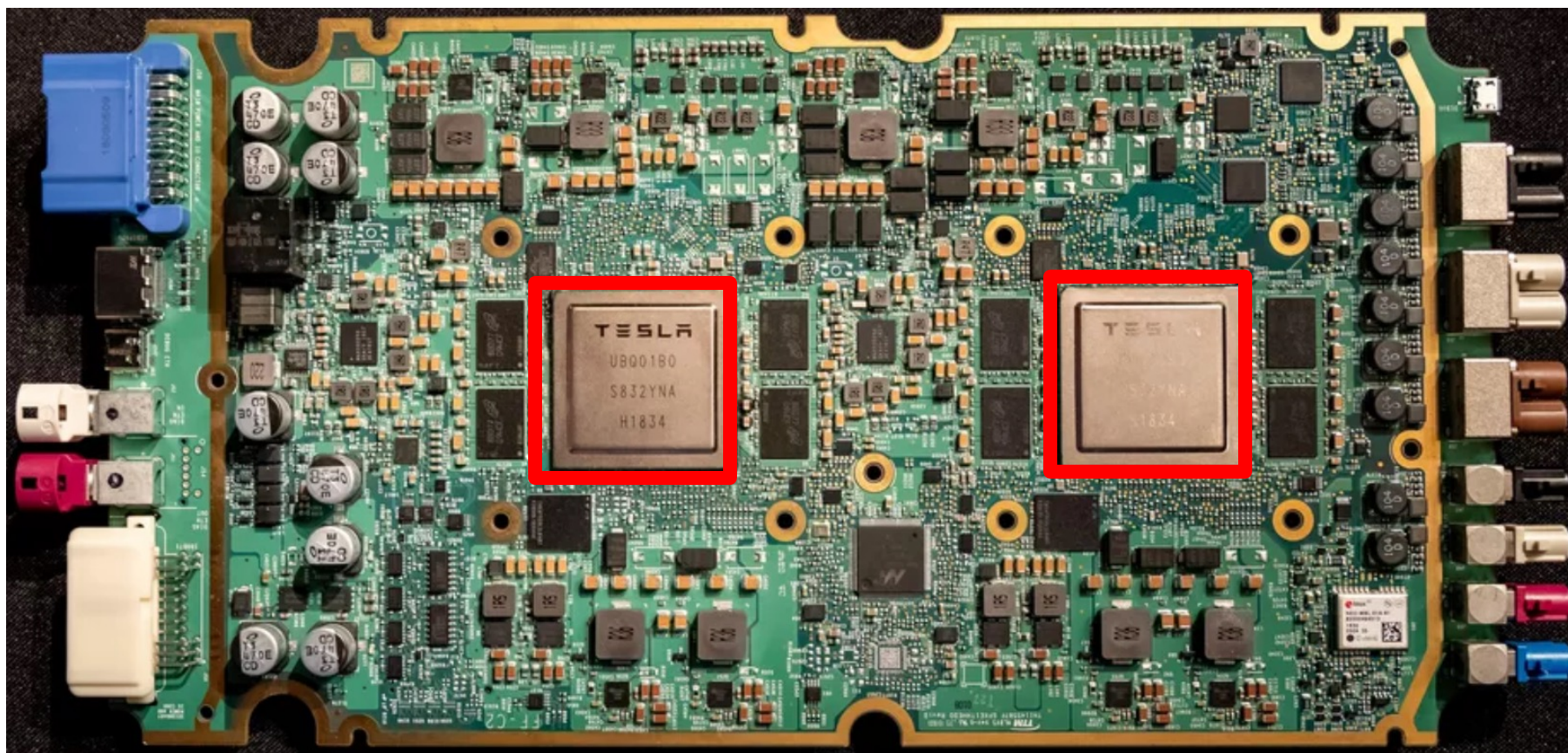
Google has decided that YouTube demands such a huge transcoding workload that it needs to build its own server chips. The company detailed its new "Argos" chips in a [YouTube blog post](#), a [CNET interview](#), and in a [paper](#) for ASPLOS, the Architectural Support for Programming Languages and Operating Systems Conference. Just as there are GPUs for graphics workloads and Google's TPU (tensor processing unit) for AI workloads, the YouTube infrastructure team says it has created the "VCU" or "Video (trans)Coding Unit," which helps YouTube transcode a single video into over a dozen versions that it needs to provide a smooth.

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

Source: <https://arstechnica.com/gadgets/2021/04/youtube-is-now-building-its-own-video-transcoding-chips/>

TESLA Full Self-Driving Computer (2019)

- ML accelerator: 260 mm², 6 billion transistors, 600 GFLOPS GPU, 12 ARM 2.2 GHz CPUs.
- Two redundant chips for better safety.



Tesla Dojo ML Training Chip (2021)



■ Tesla Dojo Chip

D1 Chip

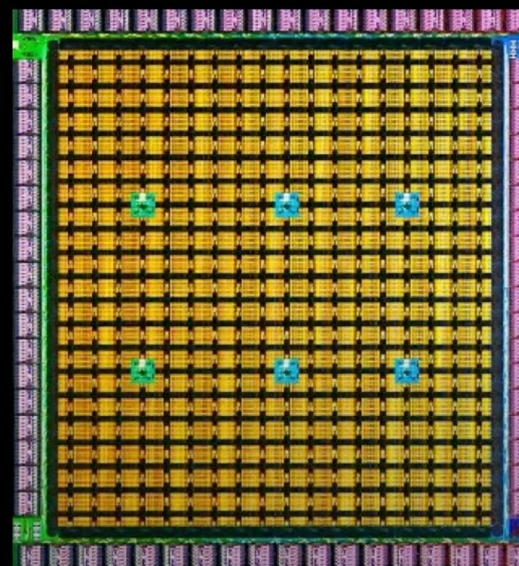
362 TFLOPs BF16/CFP8

22.6 TFLOPs FP32

10TBps/dir. On-Chip Bandwidth

4TBps/edge. Off-Chip Bandwidth

400W TDP



645mm²
7nm Technology

50 Billion
Transistors

11+ Miles
Of Wires

Tesla Dojo ML Training System (2021)



■ Tesla Dojo System

Neural Network Training - Compute



2021: 3x Clusters

1752 GPUs
5PB NVME
Infiniband EDR

Auto-labelling

4032 GPUs
8PB NVME
Infiniband EDR

Training

5760 GPUs
12PB NVME
Infiniband HDR

Training



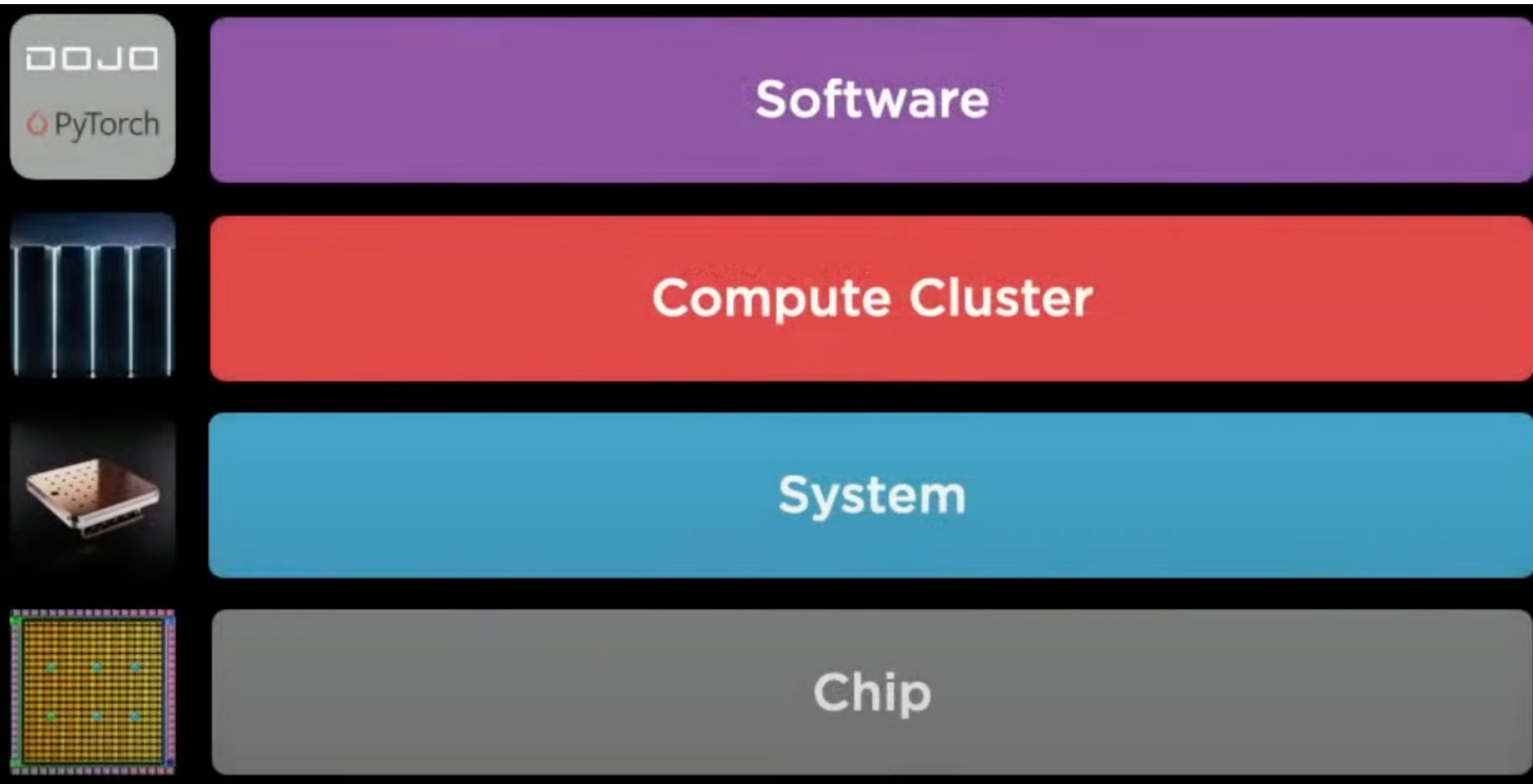
1:45:20 / 3:03:20 · Hardware Integration >



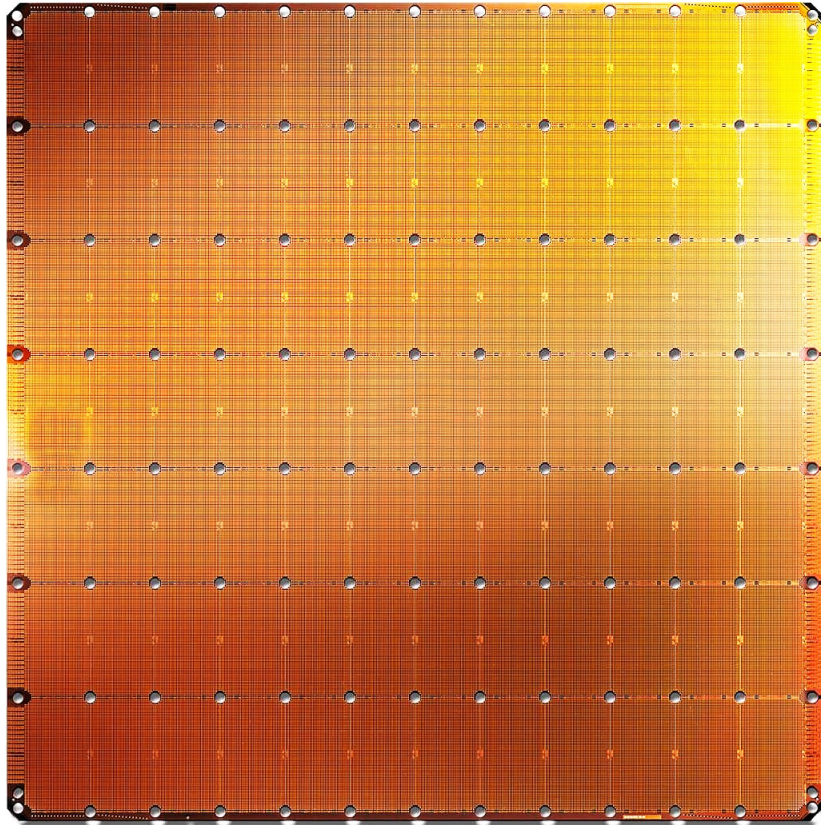
Tesla Dojo ML Training System (2021)



- Tesla Dojo Chip & System



Cerebras's Wafer Scale ML Engine (2019)



Cerebras WSE

1.2 Trillion transistors
46,225 mm²

- The largest ML accelerator chip
- 400,000 cores



Largest GPU

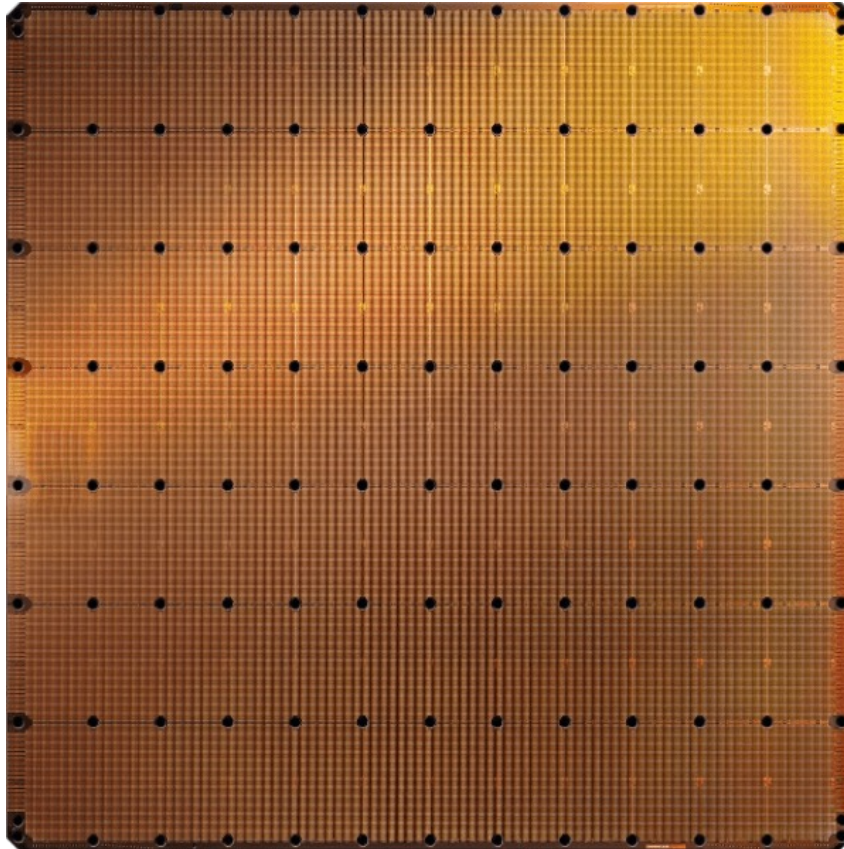
21.1 Billion transistors
815 mm²

NVIDIA TITAN V

<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 ML Engine-2 (2021)



Cerebras WSE-2
2.6 Trillion transistors
46,225 mm²

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



Largest GPU
54.2 Billion transistors
826 mm²
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/>

Google Tensor Processing Unit (~2016)

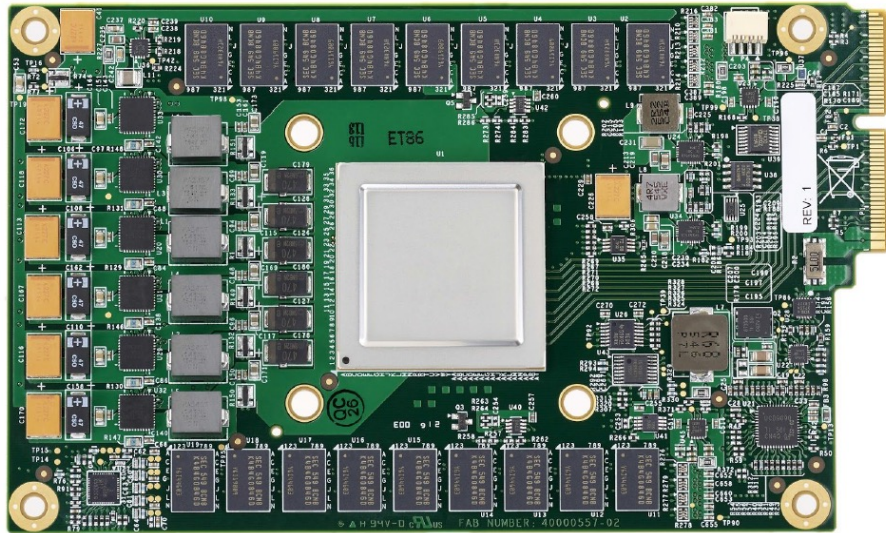


Figure 3. TPU Printed Circuit Board. It can be inserted in the slot for an SATA disk in a server, but the card uses PCIe Gen3 x16.

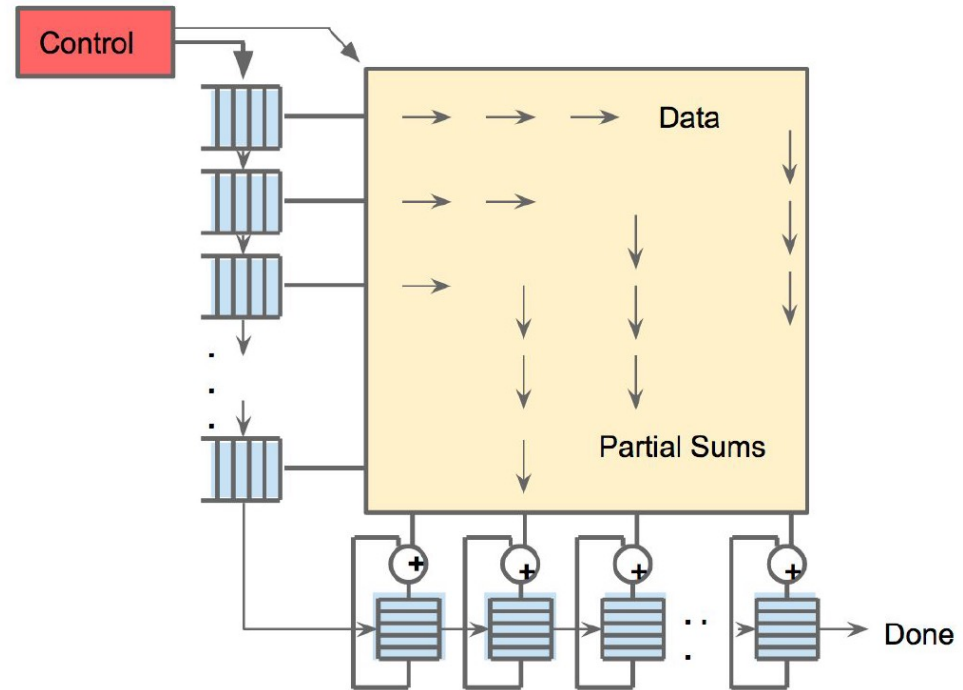
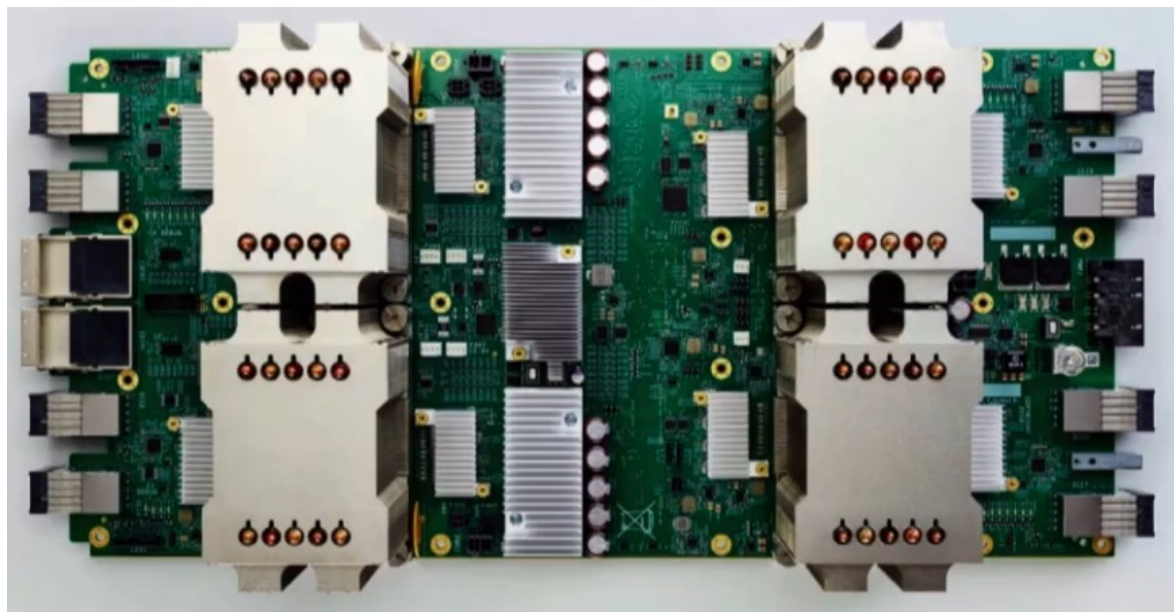


Figure 4. Systolic data flow of the Matrix Multiply Unit. Software has the illusion that each 256B input is read at once, and they instantly update one location of each of 256 accumulator RAMs.

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

Google TPU Generation II (2017)



<https://www.nextplatform.com/2017/05/17/first-depth-look-googles-new-second-generation-tpu/>

4 TPU chips
vs 1 chip in TPU1

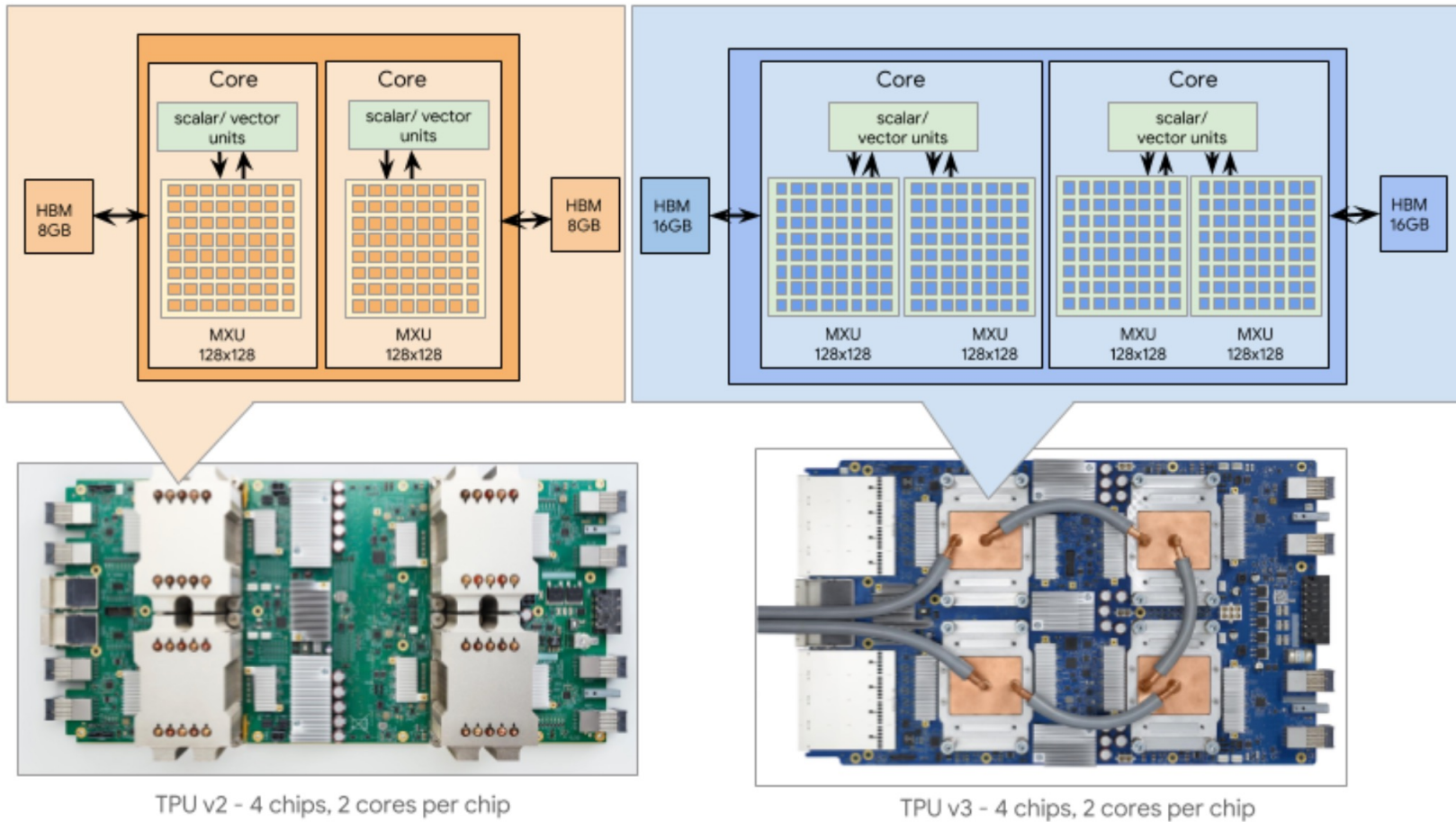
High Bandwidth Memory
vs DDR3

Floating point operations
vs FP16

45 TFLOPS per chip
vs 23 TOPS

Designed for **training**
and **inference**
vs only inference

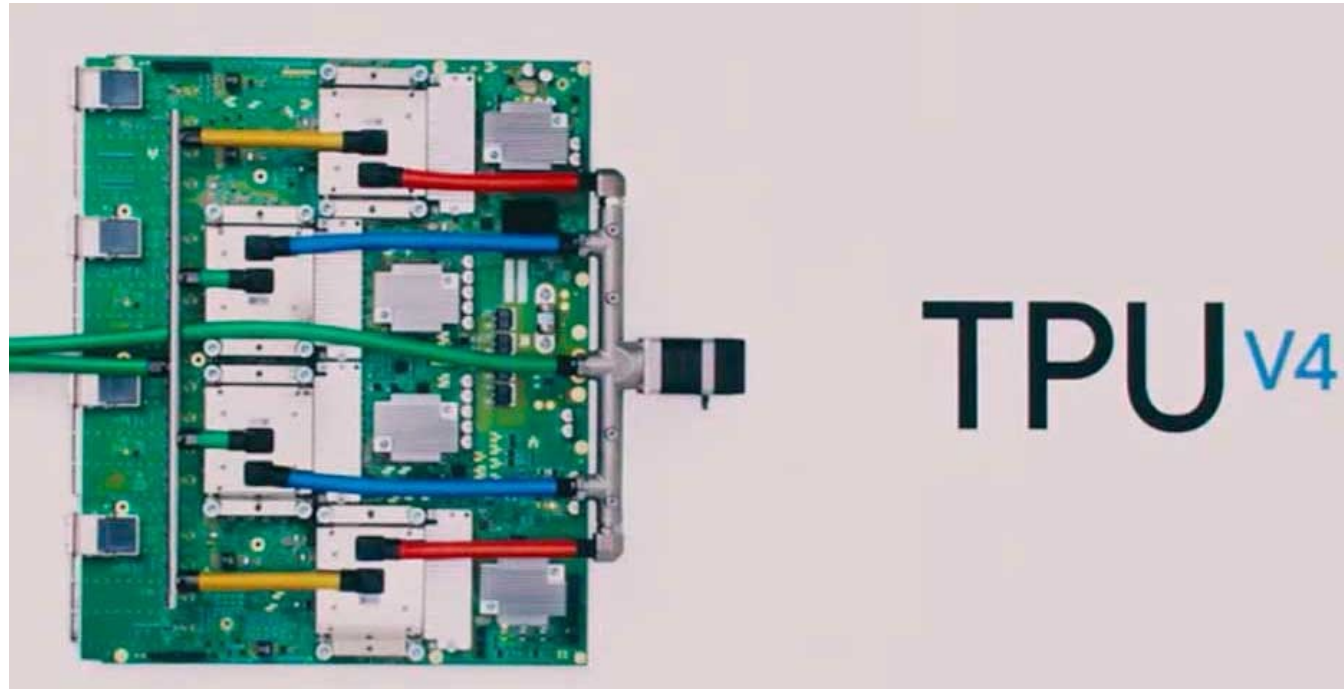
Google TPU Generation III



More
High Bandwidth Memory

More
Systolic Arrays

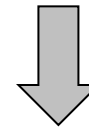
Google TPU Generation IV (2021)



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

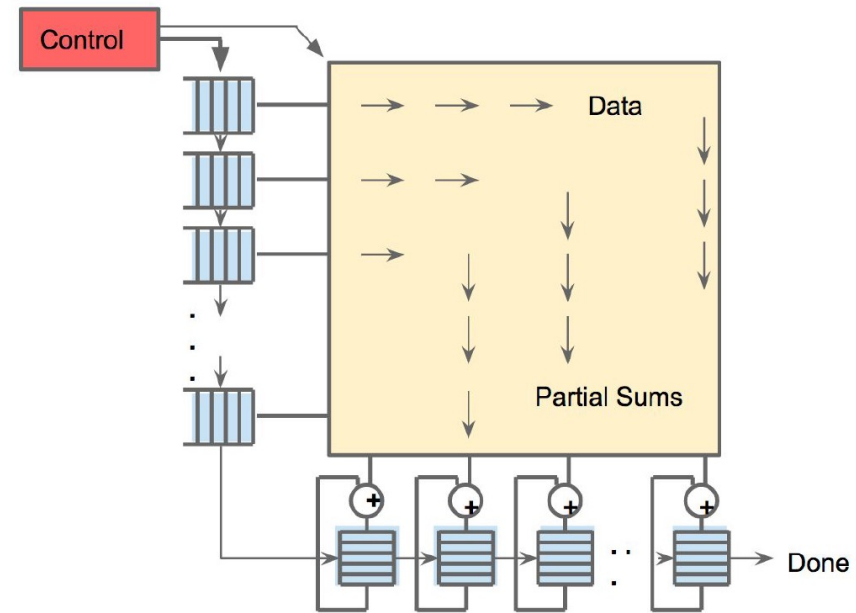


1 ExaFLOPS per board

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

An Example Modern Systolic Array: TPU (II)

As reading a large SRAM uses much more power than arithmetic, the matrix unit uses systolic execution to save energy by reducing reads and writes of the Unified Buffer [Kun80][Ram91][Ovt15b]. Figure 4 shows that data flows in from the left, and the weights are loaded from the top. A given 256-element multiply-accumulate operation moves through the matrix as a diagonal wavefront. The weights are preloaded, and take effect with the advancing wave alongside the first data of a new block. Control and data are pipelined to give the illusion that the 256 inputs are read at once, and that they instantly update one location of each of 256 accumulators. From a correctness perspective, software is unaware of the systolic nature of the matrix unit, but for performance, it does worry about the latency of the unit.



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

An Example Modern Systolic Array: TPU (III)

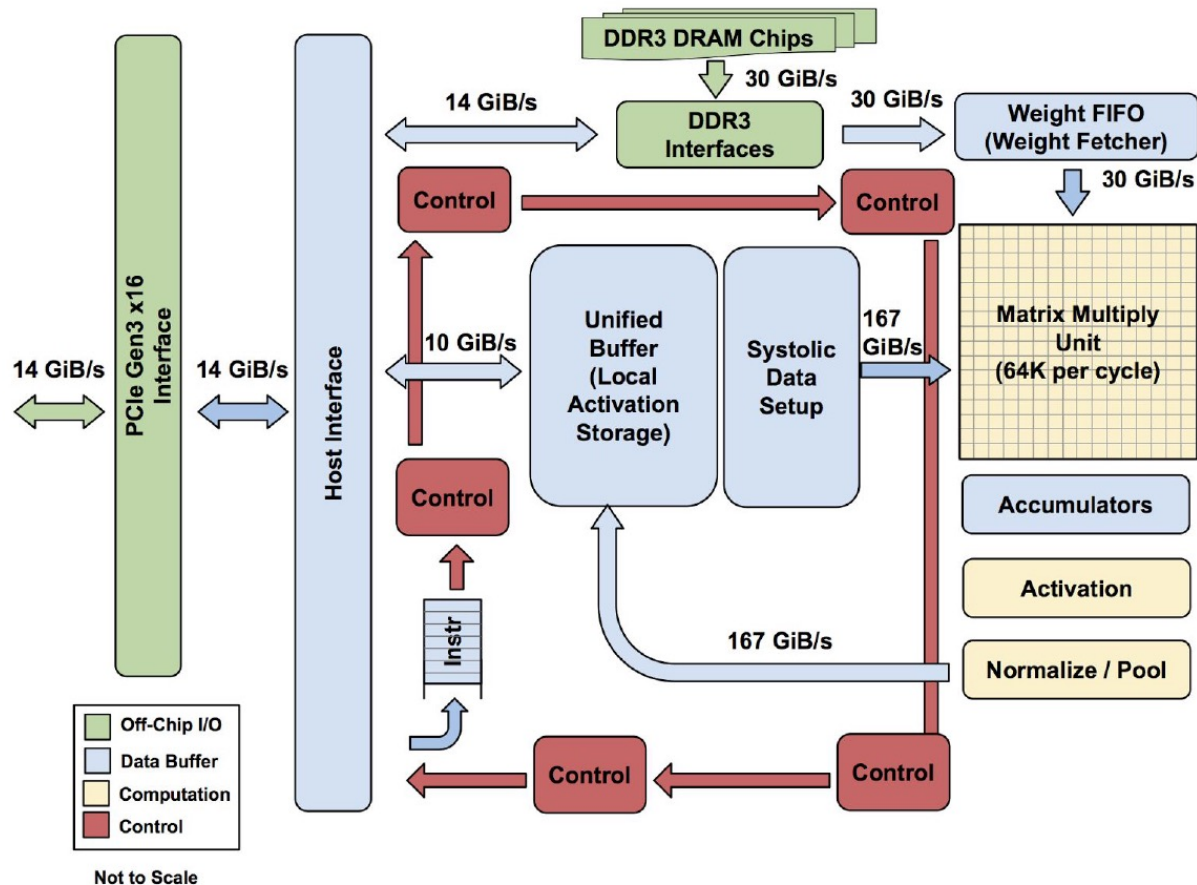


Figure 1. TPU Block Diagram. The main computation part is the yellow Matrix Multiply unit in the upper right hand corner. Its inputs are the blue Weight FIFO and the blue Unified Buffer (UB) and its output is the blue Accumulators (Acc). The yellow Activation Unit performs the nonlinear functions on the Acc, which go to the UB.

Many (Other) AI/ML Chips

- Alibaba
- Amazon
- Facebook
- Google
- Huawei
- Intel
- Microsoft
- NVIDIA
- Tesla
- Many Others and Many Startups...
- **Many More to Come...**

Many (Other) AI/ML Chips (2021)

AI Chip Landscape

V0.7 Dec., 2019

S.T.

■ MLPerf results available ■ AI-Benchmark results available

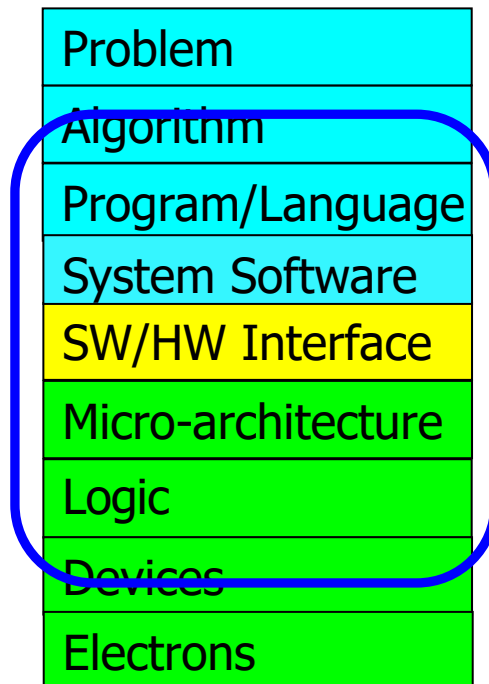


All information contained within this infographic is gathered from the internet and periodically updated, no guarantee is given that the information provided is correct, complete, and up-to-date.

Recall Our 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

Many Interesting Things
Are Happening Today
in Computer Architecture

Reliability
Safety
Security
Privacy

Collapse of the “Galloping Gertie”



Another View



How Secure Are These People?



Security is about preventing unforeseen consequences

How Safe & Secure Is **This** Platform?



Security: RowHammer (2014)



The Story of RowHammer

- One can **predictably induce bit flips** in commodity DRAM chips
 - All tested DRAM chips are vulnerable
- First example of how a **simple hardware failure mechanism** can create a **widespread system security vulnerability**

WIRED

Forget Software—Now Hackers Are Exploiting Physics

BUSINESS	CULTURE	DESIGN	GEAR	SCIENCE
----------	---------	--------	------	---------

ANDY GREENBERG SECURITY 08.31.16 7:00 AM

SHARE



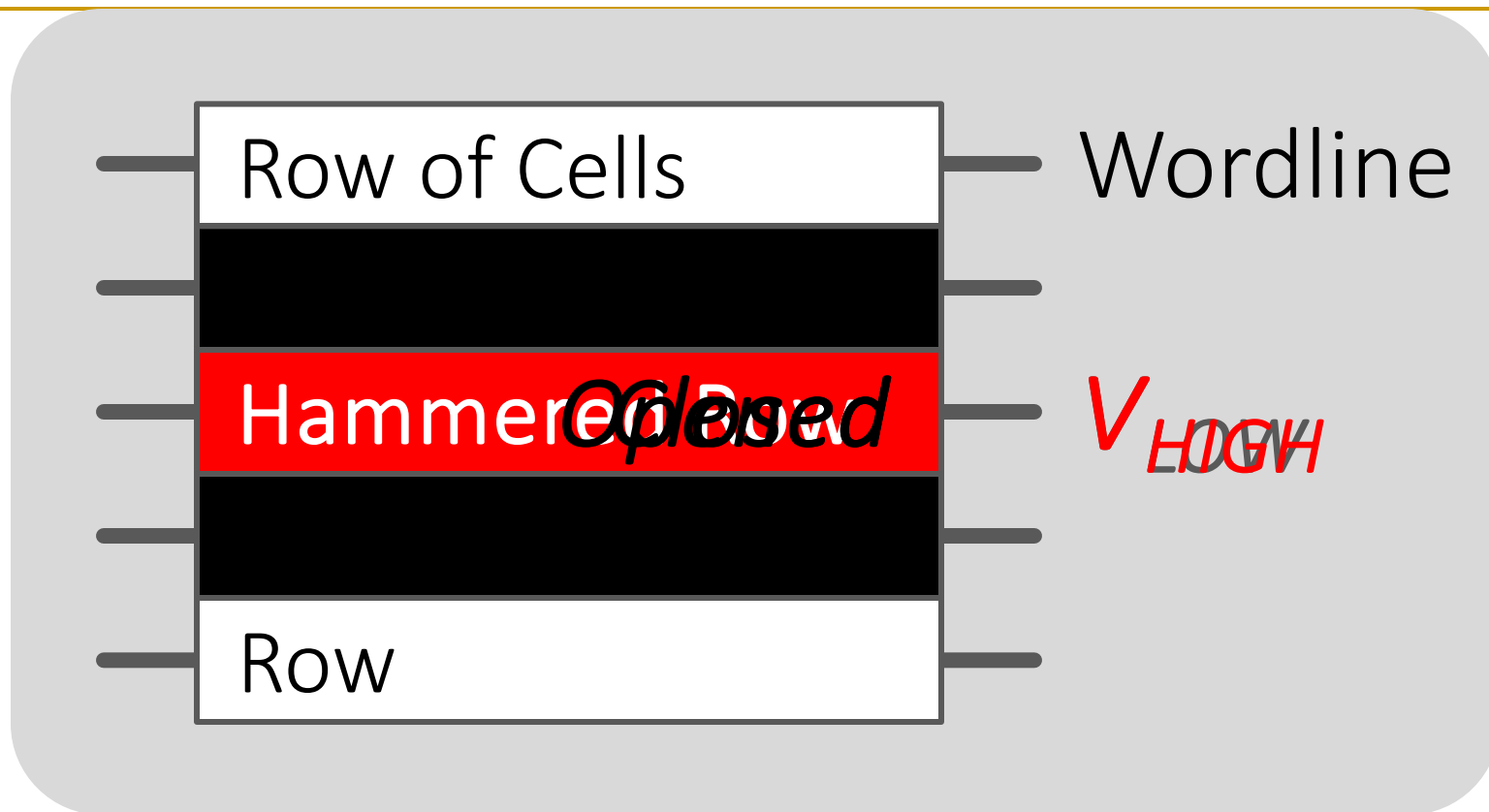
SHARE
18276



TWEET

FORGET SOFTWARE—NOW HACKERS ARE EXPLOITING PHYSICS

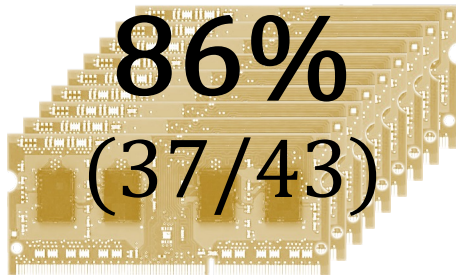
Modern DRAM is Prone to Disturbance Errors



Repeatedly reading a row enough times (before memory gets refreshed) induces **disturbance errors** in adjacent rows in **most real DRAM chips you can buy today**

Most DRAM Modules Are Vulnerable

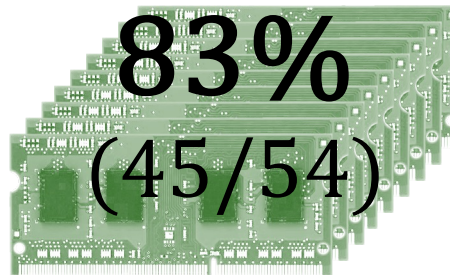
A company



Up to
 1.0×10^7

errors

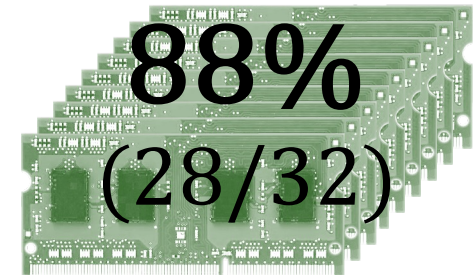
B company



Up to
 2.7×10^6

errors

C company



Up to
 3.3×10^5

errors

One Can Take Over an Otherwise-Secure System

Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors

Abstract. Memory isolation is a key property of a reliable and secure computing system — an access to one memory address should not have unintended side effects on data stored in other addresses. However, as DRAM process technology

Project Zero

[Flipping Bits in Memory Without Accessing Them:
An Experimental Study of DRAM Disturbance Errors](#)
(Kim et al., ISCA 2014)

News and updates from the Project Zero team at Google

[Exploiting the DRAM rowhammer bug to
gain kernel privileges](#) (Seaborn+, 2015)

Monday, March 9, 2015

Exploiting the DRAM rowhammer bug to gain kernel privileges

Security: RowHammer (2014)



It's like breaking into an apartment by repeatedly slamming a neighbor's door until the vibrations open the door you were after

More Security Implications (II)

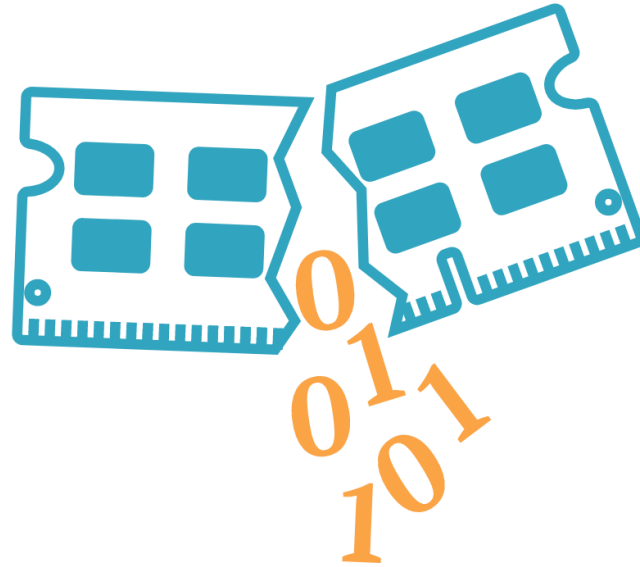
"Can gain control of a smart phone deterministically"



Drammer: Deterministic Rowhammer Attacks on Mobile Platforms, CCS'16

More Security Implications (VI)

- IEEE S&P 2020



RAMBleed

RAMBleed: Reading Bits in Memory Without Accessing Them

Andrew Kwong

University of Michigan
ankwong@umich.edu

Daniel Genkin

University of Michigan
genkin@umich.edu

Daniel Gruss

Graz University of Technology
daniel.gruss@iaik.tugraz.at

Yuval Yarom

University of Adelaide and Data61
yval@cs.adelaide.edu.au

More Security Implications (VII)

Terminal Brain Damage: Exposing the Graceless Degradation in Deep Neural Networks Under Hardware Fault Attacks

Sanghyun Hong, Pietro Frigo[†], Yiğitcan Kaya, Cristiano Giuffrida[†], Tudor Dumitraş

University of Maryland, College Park

[†]Vrije Universiteit Amsterdam



A Single Bit-flip Can Cause Terminal Brain Damage to DNNs

One specific bit-flip in a DNN's representation leads to accuracy drop over 90%

Our research found that a specific bit-flip in a DNN's bitwise representation can cause the accuracy loss up to 90%, and the DNN has 40-50% parameters, on average, that can lead to the accuracy drop over 10% when individually subjected to such single bitwise corruptions...

[Read More](#)

More Security Implications (VIII)

■ DeepHammer: Depleting the Intelligence of Deep Neural Networks through Targeted Chain of Bit Flips

Fan Yao
University of Central Florida
fan.yao@ucf.edu

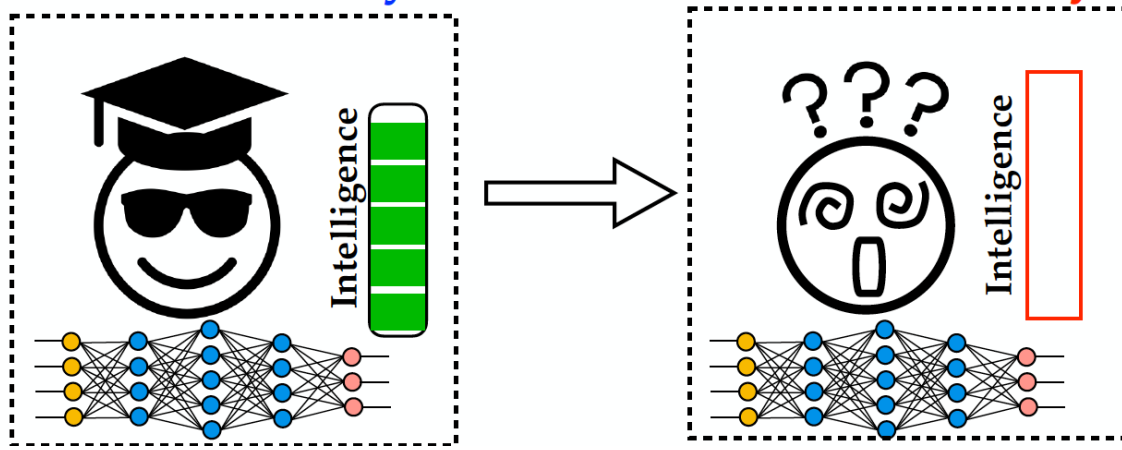
Adnan Siraj Rakin
Arizona State University
asrakin@asu.edu

Deliang Fan
Arizona State University
dfan@asu.edu

Degrade the inference accuracy to the level of Random Guess

Example: ResNet-20 for CIFAR-10, 10 output classes

Before attack, **Accuracy: 90.2%** After attack, **Accuracy: ~10% (1/10)**



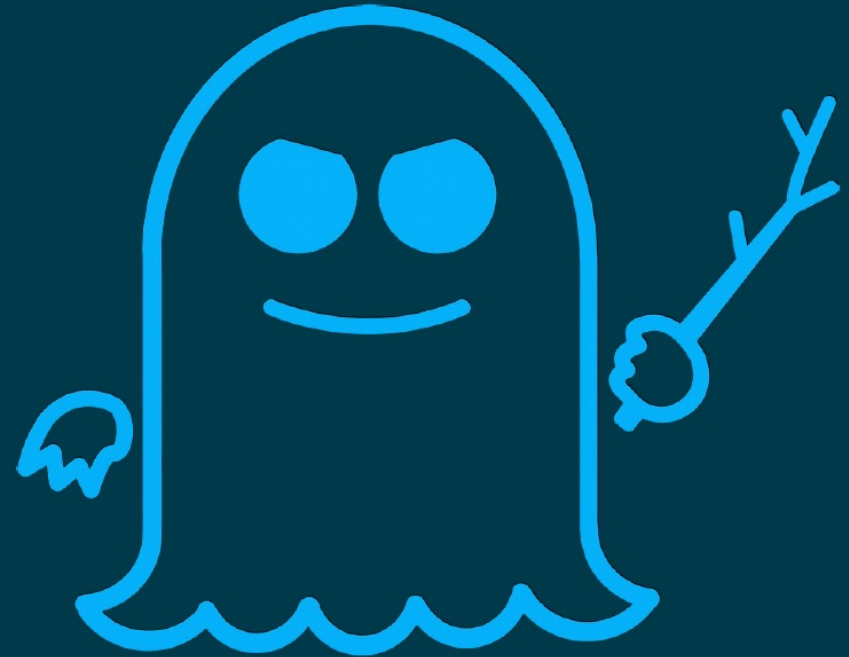
Can We Truly Depend on Computers?



Security: Meltdown and Spectre (2018)



MELTDOWN



SPECTRE

Silent Data Corruption In-the-Field (2021)

We have a *new* problem: cores that disobey instructions

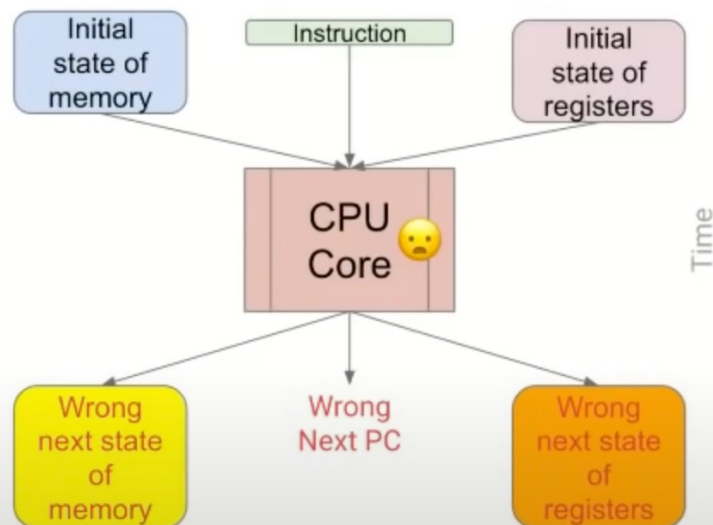
CPU cores that

- repeatedly
- but not always
- mis-calculate
- certain computations
- without giving any obvious signal

"Mercurial cores" committing

"Corrupt Execution Errors"

Due to local silicon defects, not eg cosmic rays



Google

2

▶ ⏪ 🔊 0:19 / 9:14 • We have a new problem cores that disobey instructions >

HotOS 2021: Cores That Don't Count (Fun Hardware)

<https://www.youtube.com/watch?v=QMF3rqhjYuM>

Silent Data Corruption In-the-Field (2021)

Silent Data Corruptions at Scale

Harish Dattatraya
Dixit
Facebook, Inc.
hdd@fb.com

Sneha Pendharkar
Facebook, Inc.
spendharkar@fb.com

Matt Beadon
Facebook, Inc.
mbeadon@fb.com

Chris Mason
Facebook, Inc.
clm@fb.com

Tejasvi Chakravarthy
Facebook, Inc.
teju@fb.com

Bharath Muthiah
Facebook, Inc.
bharathm@fb.com

Sriram Sankar
Facebook Inc.
sriramsankar@fb.com

Cores that don't count

Peter H. Hochschild
Paul Turner
Jeffrey C. Mogul
Google
Sunnyvale, CA, US

Rama Govindaraju
Parthasarathy
Ranganathan
Google
Sunnyvale, CA, US

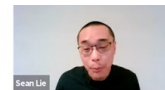
David E. Culler
Amin Vahdat
Google
Sunnyvale, CA, US

Many Interesting Things
Are Happening Today
in Computer Architecture

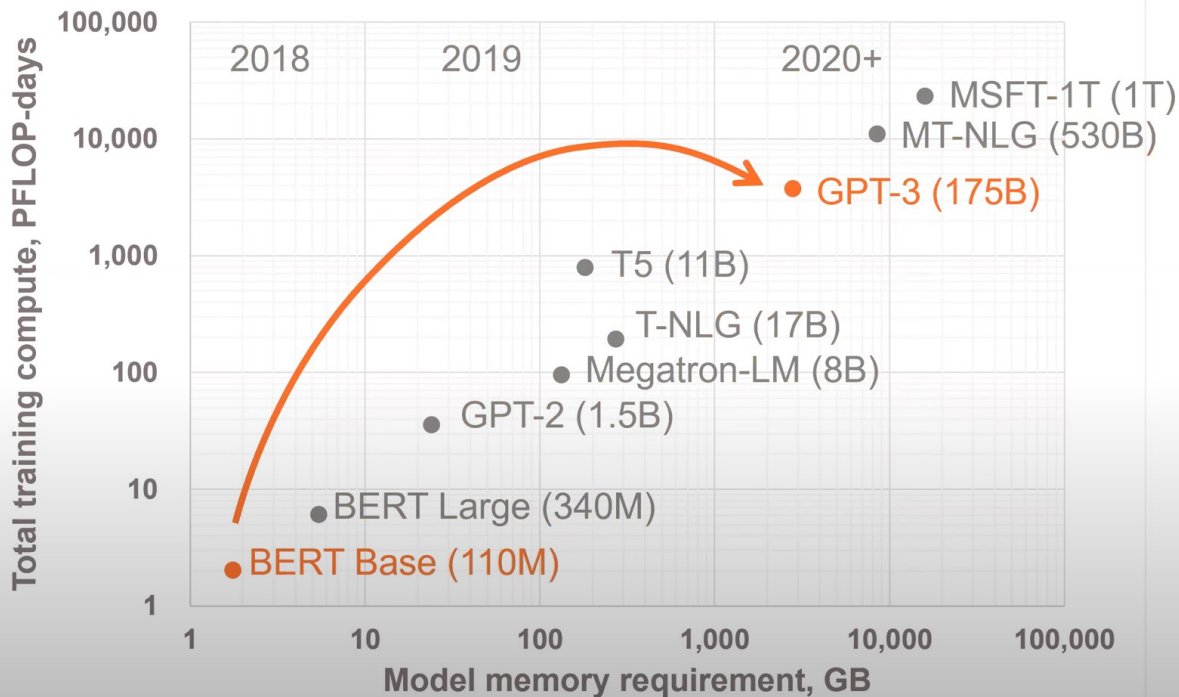
More Demanding Workloads

Huge Demand for Performance & Efficiency

Exponential Growth of Neural Networks



Memory and compute requirements



1800x more compute
In just 2 years

Tomorrow, **multi-trillion** parameter models

Increasingly Demanding Applications

Dream

and, they will come

As applications push boundaries, computing platforms will become increasingly strained.

New Genome Sequencing Technologies

Nanopore sequencing technology and tools for genome assembly: computational analysis of the current state, bottlenecks and future directions

Damla Senol Cali ✉, Jeremie S Kim, Saugata Ghose, Can Alkan, Onur Mutlu

Briefings in Bioinformatics, bby017, <https://doi.org/10.1093/bib/bby017>

Published: 02 April 2018 **Article history** ▼



Oxford Nanopore MinION

Data → performance & energy bottleneck

Why Do We Care? An Example

200 Oxford Nanopore sequencers have left UK for China, to support rapid, near-sample coronavirus sequencing for outbreak surveillance

Fri 31st January 2020

Following extensive support of, and collaboration with, public health professionals in China, Oxford Nanopore has shipped an additional 200 MinION sequencers and related consumables to China. These will be used to support the ongoing surveillance of the current coronavirus outbreak, adding to a large number of the devices already installed in the country.



Each MinION sequencer is approximately the size of a stapler, and can provide rapid sequence information about the coronavirus.



700Kg of Oxford Nanopore sequencers and consumables are on their way for use by Chinese scientists in understanding the current coronavirus outbreak.

Population-Scale Microbiome Profiling



City-Scale Microbiome Profiling

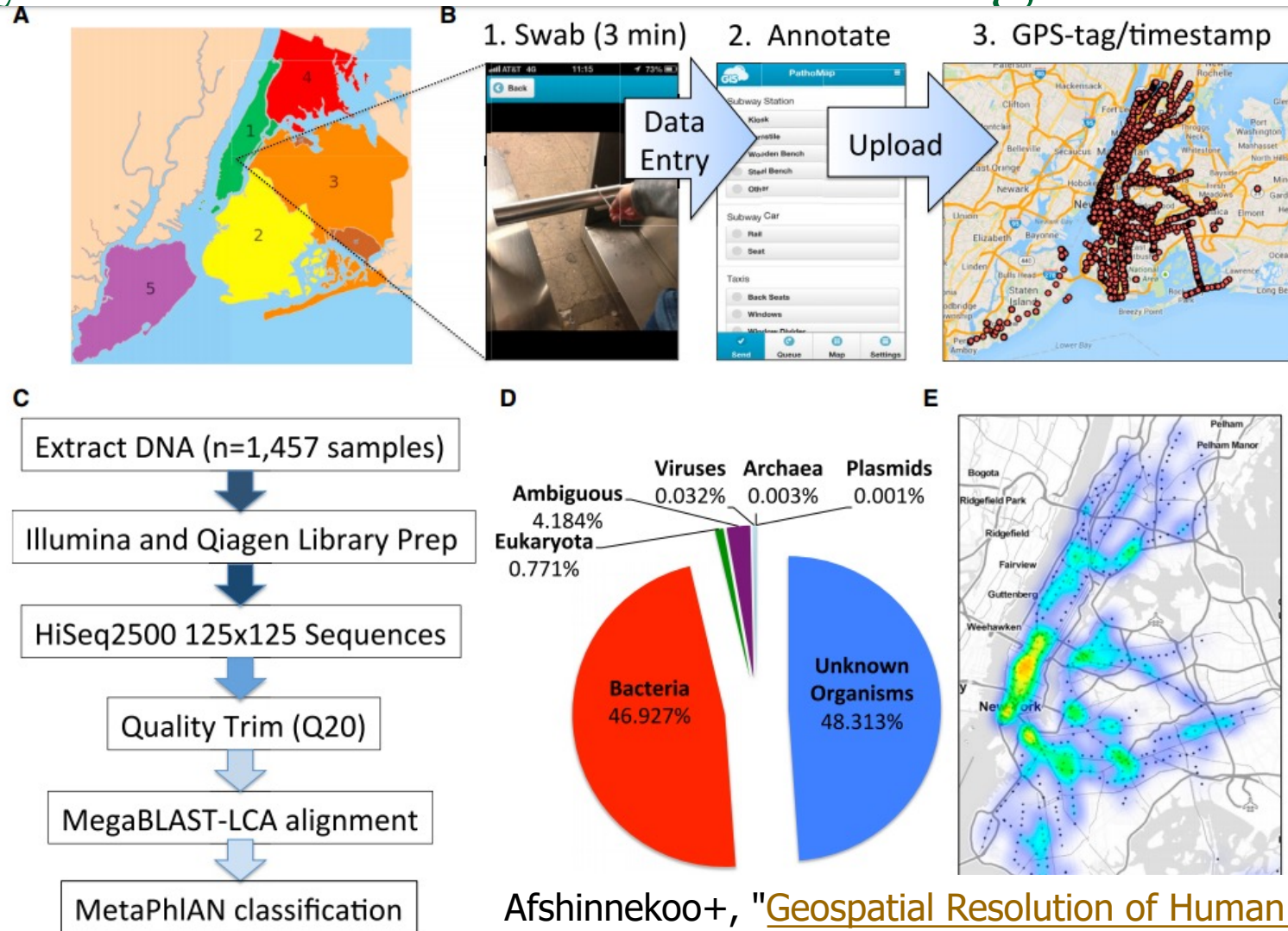


Figure 1. The Metagenome of New York City

(A) The five boroughs of NYC include (1) Manhattan (green)

(B) The collection from the 466 subway stations of NYC across the 24 subway lines involved three main steps: (1) collection with Copan Elution swabs, (2) data entry into the database, and (3) uploading of the data. An image is shown of the current collection database, taken from <http://pathomap.giscloud.com>.

(C) Workflow for sample DNA extraction, library preparation, sequencing, quality trimming of the FASTQ files, and alignment with MegaBLAST and MetaPhlan to discern taxa present

Afshinnekoo+, "Geospatial Resolution of Human and Bacterial Diversity with City-Scale Metagenomics", Cell Systems, 2015

Example: Rapid Surveillance of Ebola Outbreak

Figure 1: Deployment of the portable genome surveillance system in Guinea.



Quick+, "Real-time, portable genome sequencing for Ebola surveillance", *Nature*, 2016

High-Throughput Genome Sequencers



Illumina MiSeq



Pacific
Biosciences
Sequel II

Oxford
Nanopore
PromethION



Oxford Nanopore MinION



Illumina NovaSeq 6000



Pacific Biosciences RS II



Oxford
Nanopore
SmidgION

... and more! All produce data with different properties.

High-Throughput Genome Sequencers

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](https://doi.org/10.1109/MM.2020.3013728)

FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications

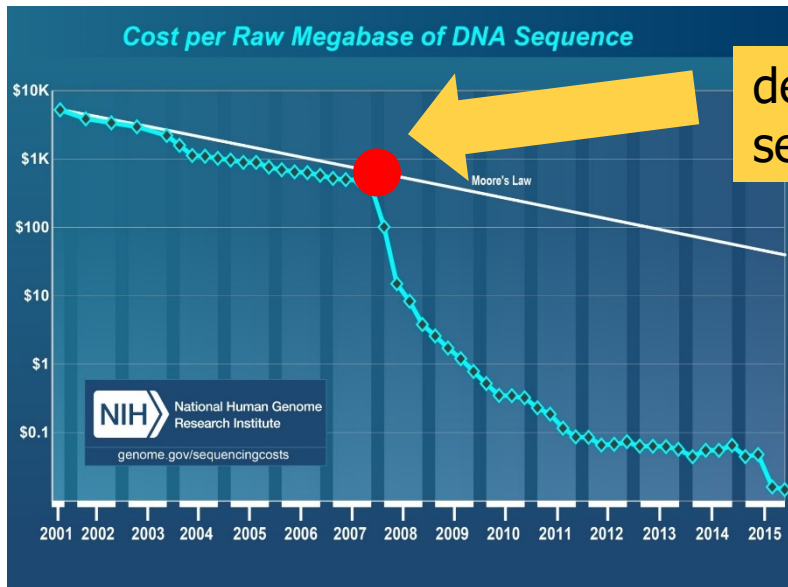
July-Aug. 2021, pp. 39-48, vol. 41

DOI Bookmark: [10.1109/MM.2021.3088396](https://doi.org/10.1109/MM.2021.3088396)



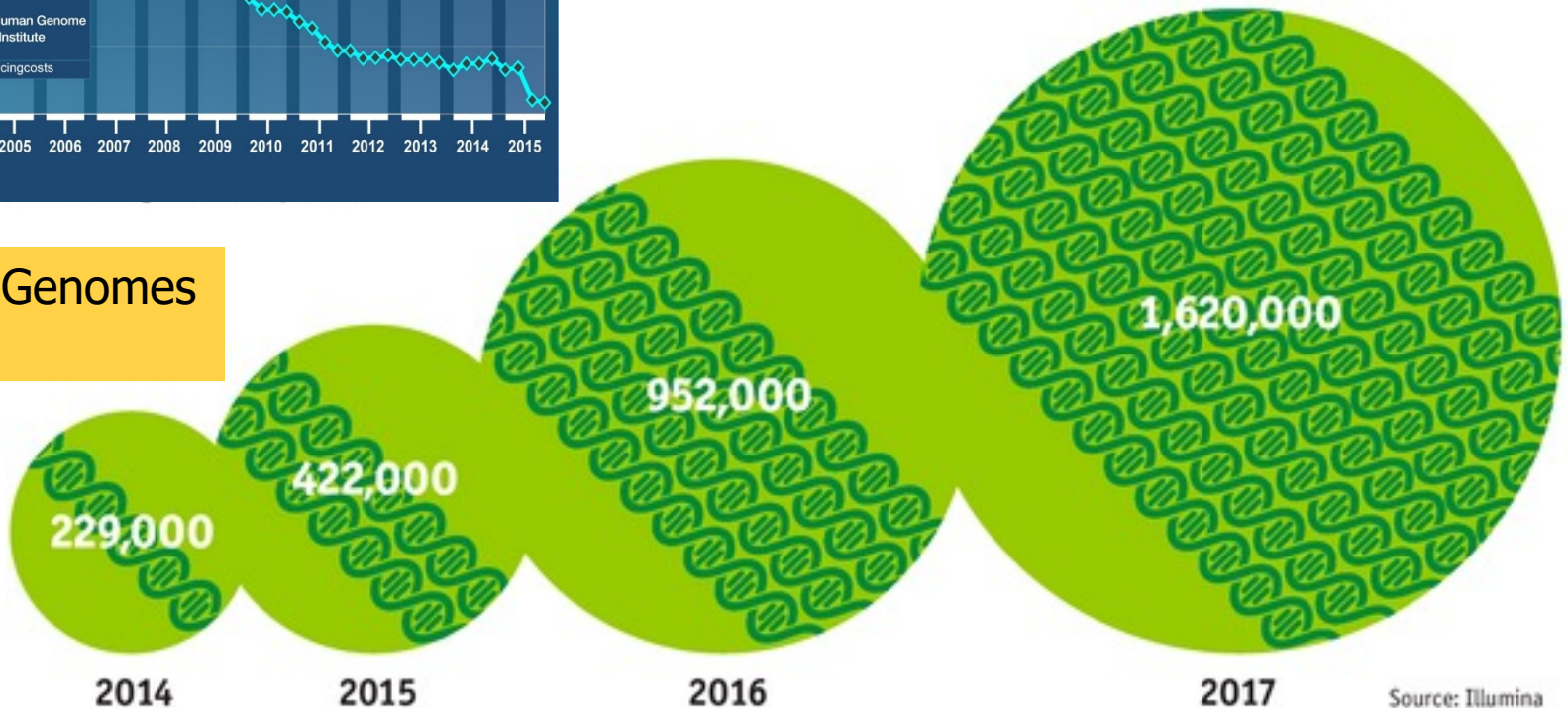
SmidgION from ONT

The Genomic Era



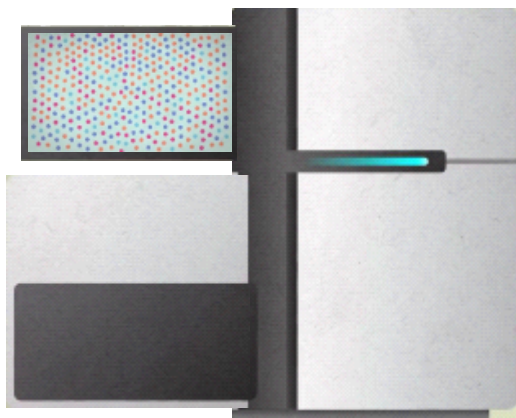
development of high-throughput sequencing (HTS) technologies

Number of Genomes Sequenced



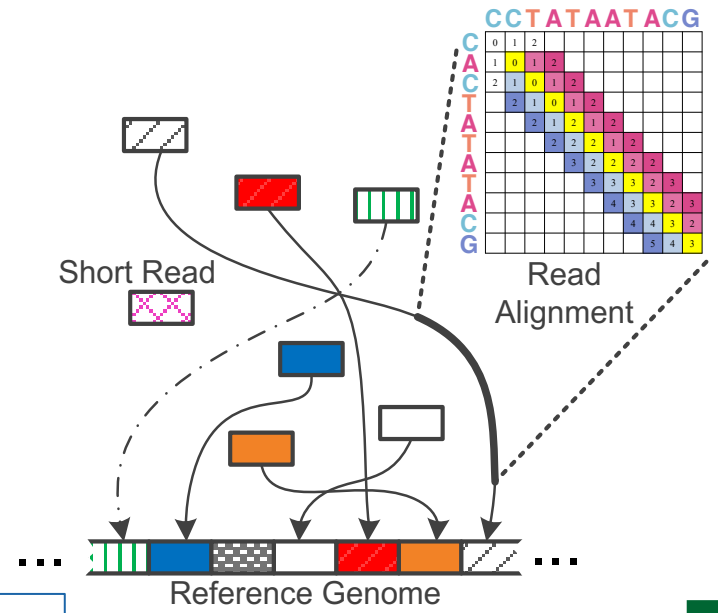
The Economist

Source: Illumina



Billions of Short Reads

ATATATACGTA
 TTTAGTACGTACGT
 ATACGTA
 CG CCCCTACGTA
 CGTACTAGTACGT
 TTAGTACGTACGT
 TACGTA
 TACGTA
 TTTAAACGTA
 CGTACTAGTACGT
 GGGAGTACGTACGT



1 Sequencing

Genome Analysis

2 Read Mapping

Data → performance & energy bottleneck

read4: CGCTTCCAT
 read5: CCATGACGC
 read6: TTCCATGAC



3 Variant Calling

4 Scientific Discovery

We Need Faster & Scalable Genome Analysis



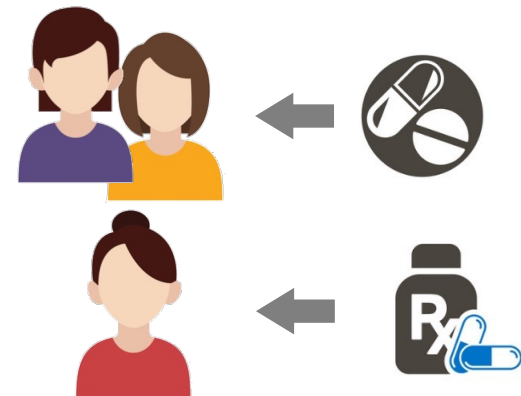
Understanding **genetic variations, species, evolution, ...**



Predicting the **presence and relative abundance of microbes** in a sample



Rapid surveillance of **disease outbreaks**



Developing **personalized medicine**

And, many, many other applications ...

Our Dream (circa 2007)

- An embedded device that can perform comprehensive genome analysis in real time (within a minute)
 - Which of these DNAs does this DNA segment match with?
 - What is the likely genetic disposition of this patient to this drug?
 - What disease/condition might this particular DNA/RNA piece associated with?
 - What potential viruses & variants might be lurking around?
 - . . .

Software Acceleration: Eliminate Useless Work

- Download the source code and try for yourself
 - [Download link to FastHASH](#)

Xin *et al.* *BMC Genomics* 2013, **14**(Suppl 1):S13
<http://www.biomedcentral.com/1471-2164/14/S1/S13>



PROCEEDINGS

Open Access

Accelerating read mapping with FastHASH

Hongyi Xin¹, Donghyuk Lee¹, Farhad Hormozdiari², Samihan Yedkar¹, Onur Mutlu^{1*}, Can Alkan^{3*}

From The Eleventh Asia Pacific Bioinformatics Conference (APBC 2013)
Vancouver, Canada. 21-24 January 2013

Hardware Acceleration: Vectorizable Algorithms

<https://github.com/CMU-SAFARI/Shifted-Hamming-Distance>

Bioinformatics, 31(10), 2015, 1553–1560

doi: 10.1093/bioinformatics/btu856

Advance Access Publication Date: 10 January 2015

Original Paper

OXFORD

Sequence analysis

Shifted Hamming distance: a fast and accurate SIMD-friendly filter to accelerate alignment verification in read mapping

Hongyi Xin^{1,*}, John Greth², John Emmons², Gennady Pekhimenko¹,
Carl Kingsford³, Can Alkan^{4,*} and Onur Mutlu^{2,*}

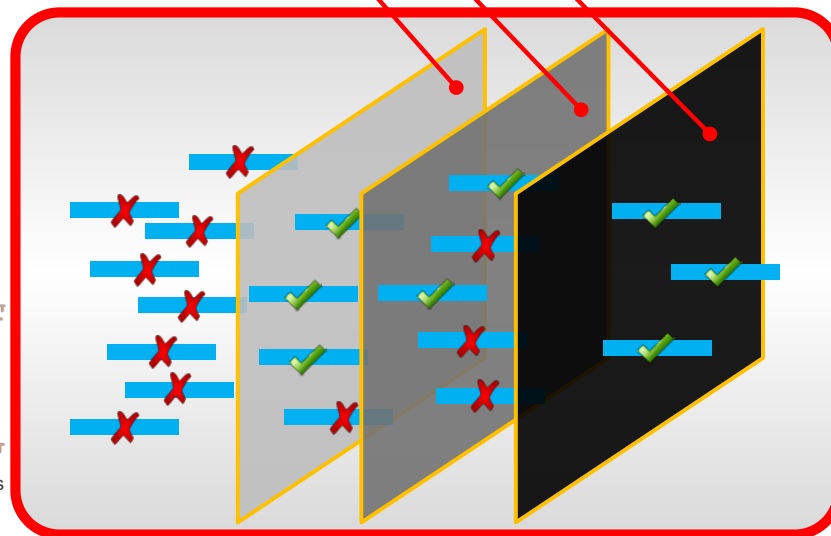
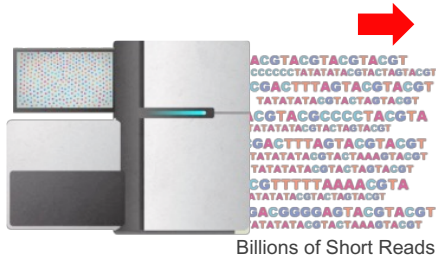
Xin+, ["Shifted Hamming Distance: A Fast and Accurate SIMD-friendly Filter to Accelerate Alignment Verification in Read Mapping"](#), **Bioinformatics 2015.**

GateKeeper: FPGA-Based Acceleration



Low Speed & High Accuracy
 Medium Speed, Medium Accuracy
 High Speed, Low Accuracy

$\times 10^{12}$
mappings



$\times 10^3$
mappings

→

	C	T	A	T	A	A	T	A	C	G
C	0	1	2							
A	1	0	1	2						
C	2	1	0	1	2					
T		2	1	0	1	2				
A			2	1	2	1	2			
T				3	2	2	1	2		
A					3	3	2	2	3	
T						4	3	3	2	3
A							4	4	3	2
C								5	4	3
G										5

- 1 High throughput DNA sequencing (HTS) technologies
- 2 Read Pre-Alignment Filtering
Fast & Low False Positive Rate
- 3 Read Alignment
Slow & Zero False Positives

GateKeeper: FPGA-Based Acceleration

- Mohammed Alser, Hasan Hassan, Hongyi Xin, Oguz Ergin, Onur Mutlu, and Can Alkan
"GateKeeper: A New Hardware Architecture for Accelerating Pre-Alignment in DNA Short Read Mapping"
Bioinformatics, [published online, May 31], 2017.
[\[Source Code\]](#)
[\[Online link at Bioinformatics Journal\]](#)

GateKeeper: a new hardware architecture for accelerating pre-alignment in DNA short read mapping

Mohammed Alser ✉, Hasan Hassan, Hongyi Xin, Oğuz Ergin, Onur Mutlu ✉, Can Alkan ✉

Bioinformatics, Volume 33, Issue 21, 1 November 2017, Pages 3355–3363,

<https://doi.org/10.1093/bioinformatics/btx342>

Published: 31 May 2017 **Article history** ▼

In-Memory DNA Sequence Analysis

- Jeremie S. Kim, Damla Senol Cali, Hongyi Xin, Donghyuk Lee, Saugata Ghose, Mohammed Alser, Hasan Hassan, Oguz Ergin, Can Alkan, and Onur Mutlu,
["GRIM-Filter: Fast Seed Location Filtering in DNA Read Mapping Using Processing-in-Memory Technologies"](#)
BMC Genomics, 2018.
Proceedings of the 16th Asia Pacific Bioinformatics Conference (APBC), Yokohama, Japan, January 2018.
[[Slides \(pptx\) \(pdf\)](#)]
[[Source Code](#)]
[[arxiv.org Version \(pdf\)](#)]
[[Talk Video at AACBB 2019](#)]

GRIM-Filter: Fast seed location filtering in DNA read mapping using processing-in-memory technologies

Jeremie S. Kim^{1,6*}, Damla Senol Cali¹, Hongyi Xin², Donghyuk Lee³, Saugata Ghose¹, Mohammed Alser⁴, Hasan Hassan⁶, Oguz Ergin⁵, Can Alkan^{4*} and Onur Mutlu^{6,1*}

From The Sixteenth Asia Pacific Bioinformatics Conference 2018
Yokohama, Japan. 15-17 January 2018

Shouji (障子) [Alser+, Bioinformatics 2019]

Mohammed Alser, Hasan Hassan, Akash Kumar, Onur Mutlu, and Can Alkan,
"Shouji: A Fast and Efficient Pre-Alignment Filter for Sequence Alignment"
Bioinformatics, [published online, March 28], 2019.

[\[Source Code\]](#)

[\[Online link at Bioinformatics Journal\]](#)

Bioinformatics, 2019, 1–9

doi: 10.1093/bioinformatics/btz234

Advance Access Publication Date: 28 March 2019

Original Paper



Sequence alignment

Shouji: a fast and efficient pre-alignment filter for sequence alignment

**Mohammed Alser^{1,2,3,*}, Hasan Hassan¹, Akash Kumar², Onur Mutlu^{1,3,*}
and Can Alkan^{3,*}**

¹Computer Science Department, ETH Zürich, Zürich 8092, Switzerland, ²Chair for Processor Design, Center For Advancing Electronics Dresden, Institute of Computer Engineering, Technische Universität Dresden, 01062 Dresden, Germany and ³Computer Engineering Department, Bilkent University, 06800 Ankara, Turkey

*To whom correspondence should be addressed.

Associate Editor: Inanc Birol

Received on September 13, 2018; revised on February 27, 2019; editorial decision on March 7, 2019; accepted on March 27, 2019

SneakySnake [Alser+, Bioinformatics 2020]

Mohammed Alser, Taha Shahroodi, Juan-Gomez Luna, Can Alkan, and Onur Mutlu,
**"SneakySnake: A Fast and Accurate Universal Genome Pre-Alignment
Filter for CPUs, GPUs, and FPGAs"**

Bioinformatics, to appear in 2020.

[\[Source Code\]](#)

[\[Online link at Bioinformatics Journal\]](#)

Bioinformatics

doi.10.1093/bioinformatics/xxxxxx

Advance Access Publication Date: Day Month Year

Manuscript Category

OXFORD

Subject Section

SneakySnake: A Fast and Accurate Universal Genome Pre-Alignment Filter for CPUs, GPUs, and FPGAs

**Mohammed Alser^{1,2,*}, Taha Shahroodi¹, Juan Gómez-Luna^{1,2},
Can Alkan^{4,*}, and Onur Mutlu^{1,2,3,4,*}**

¹ Department of Computer Science, ETH Zurich, Zurich 8006, Switzerland

² Department of Information Technology and Electrical Engineering, ETH Zurich, Zurich 8006, Switzerland

³ Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh 15213, PA, USA

⁴ Department of Computer Engineering, Bilkent University, Ankara 06800, Turkey

GenASM Framework [MICRO 2020]

- Damla Senol Cali, Gurpreet S. Kalsi, Zülal Bingöl, Can Firtina, Lavanya Subramanian, Jeremie S. Kim, Rachata Ausavarungnirun, Mohammed Alser, Juan Gomez-Luna, Amirali Boroumand, Anant Nori, Allison Scibisz, Sreenivas Subramoney, Can Alkan, Saugata Ghose, and Onur Mutlu, **"GenASM: A High-Performance, Low-Power Approximate String Matching Acceleration Framework for Genome Sequence Analysis"**
Proceedings of the 53rd International Symposium on Microarchitecture (MICRO), Virtual, October 2020.
[[Lightning Talk Video](#) (1.5 minutes)]
[[Lightning Talk Slides \(pptx\)](#) ([pdf](#))]
[[Talk Video](#) (18 minutes)]
[[Slides \(pptx\)](#) ([pdf](#))]

GenASM: A High-Performance, Low-Power Approximate String Matching Acceleration Framework for Genome Sequence Analysis

Damla Senol Cali[†][✕] Gurpreet S. Kalsi[✕] Zülal Bingöl[∇] Can Firtina[◇] Lavanya Subramanian[‡] Jeremie S. Kim[◇][†]
Rachata Ausavarungnirun[○] Mohammed Alser[◇] Juan Gomez-Luna[◇] Amirali Boroumand[†] Anant Nori[✕]
Allison Scibisz[†] Sreenivas Subramoney[✕] Can Alkan[∇] Saugata Ghose^{*†} Onur Mutlu[◇][†][∇]
[†]Carnegie Mellon University [✕]Processor Architecture Research Lab, Intel Labs [∇]Bilkent University [◇]ETH Zürich
[‡]Facebook [○]King Mongkut's University of Technology North Bangkok ^{*}University of Illinois at Urbana-Champaign

SeGraM Framework [ISCA 2022]

- Damla Senol Cali, Konstantinos Kanellopoulos, Joel Lindegger, Zülal Bingöl, Gurpreet S. Kalsi, Ziyi Zuo, Can Firtina, Meryem Banu Cavlak, Jeremie Kim, Nika Mansouri Ghiasi, Gagandeep Singh, Juan Gomez-Luna, Nour Almadhoun Alserr, Mohammed Alser, Sreenivas Subramoney, Can Alkan, Saugata Ghose, and Onur Mutlu,
"SeGraM: A Universal Hardware Accelerator for Genomic Sequence-to-Graph and Sequence-to-Sequence Mapping"
Proceedings of the 49th International Symposium on Computer Architecture (ISCA), New York, June 2022.
[\[arXiv version\]](#)

SeGraM: A Universal Hardware Accelerator for Genomic Sequence-to-Graph and Sequence-to-Sequence Mapping

Damla Senol Cali¹ Konstantinos Kanellopoulos² Joël Lindegger² Zülal Bingöl³
Gurpreet S. Kalsi⁴ Ziyi Zuo⁵ Can Firtina² Meryem Banu Cavlak² Jeremie Kim²
Nika Mansouri Ghiasi² Gagandeep Singh² Juan Gómez-Luna² Nour Almadhoun Alserr²
Mohammed Alser² Sreenivas Subramoney⁴ Can Alkan³ Saugata Ghose⁶ Onur Mutlu²

¹Bionano Genomics ²ETH Zürich ³Bilkent University ⁴Intel Labs
⁵Carnegie Mellon University ⁶University of Illinois Urbana-Champaign

FPGA-based Near-Memory Analytics

- Gagandeep Singh, Mohammed Alser, Damla Senol Cali, Dionysios Diamantopoulos, Juan Gómez-Luna, Henk Corporaal, and Onur Mutlu, ["FPGA-based Near-Memory Acceleration of Modern Data-Intensive Applications"](#)
IEEE Micro (IEEE MICRO), 2021.

FPGA-based Near-Memory Acceleration of Modern Data-Intensive Applications

Gagandeep Singh[◇] Mohammed Alser[◇] Damla Senol Cali[✕]

Dionysios Diamantopoulos[▽] Juan Gómez-Luna[◇]

Henk Corporaal^{*} Onur Mutlu^{◇✕}

[◇]*ETH Zürich* [✕]*Carnegie Mellon University*

^{*}*Eindhoven University of Technology* [▽]*IBM Research Europe*

In-Storage Genome Filtering [ASPLOS 2022]

- Nika Mansouri Ghiasi, Jisung Park, Harun Mustafa, Jeremie Kim, Ataberk Olgun, Arvid Gollwitzer, Damla Senol Cali, Can Firtina, Haiyu Mao, Nour Almadhoun Alserr, Rachata Ausavarungnirun, Nandita Vijaykumar, Mohammed Alser, and Onur Mutlu, **["GenStore: A High-Performance and Energy-Efficient In-Storage Computing System for Genome Sequence Analysis"](#)**
Proceedings of the 27th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), Virtual, February-March 2022.
[[Lightning Talk Slides \(pptx\)](#)] ([pdf](#))

GenStore: A High-Performance In-Storage Processing System for Genome Sequence Analysis

Nika Mansouri Ghiasi¹ Jisung Park¹ Harun Mustafa¹ Jeremie Kim¹ Ataberk Olgun¹
Arvid Gollwitzer¹ Damla Senol Cali² Can Firtina¹ Haiyu Mao¹ Nour Almadhoun Alserr¹
Rachata Ausavarungnirun³ Nandita Vijaykumar⁴ Mohammed Alser¹ Onur Mutlu¹

¹ETH Zürich ²Bionano Genomics ³KMUTNB ⁴University of Toronto

Future of Genome Sequencing & Analysis

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](https://doi.org/10.1109/MM.2020.3013728)

FPGA-Based Near-Memory Acceleration of Modern Data-Intensive Applications

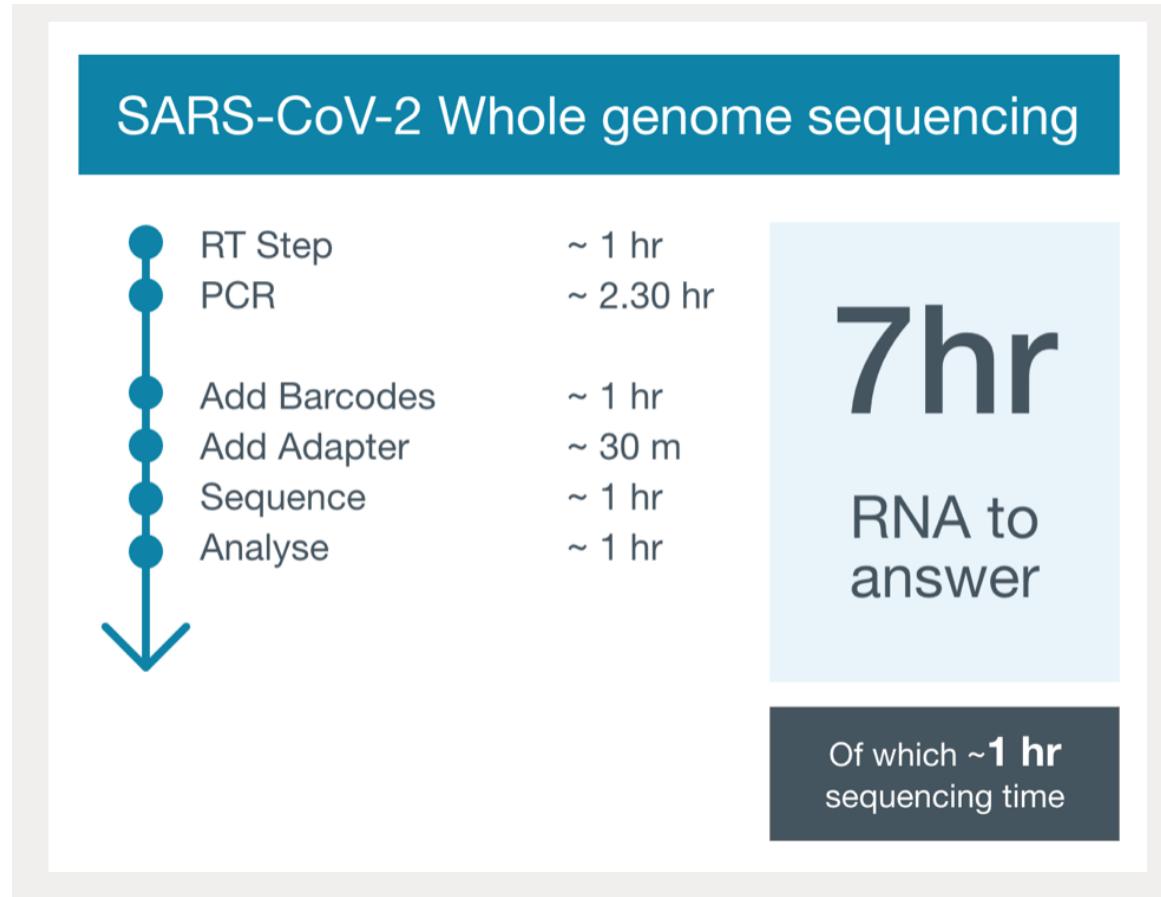
July-Aug. 2021, pp. 39-48, vol. 41

DOI Bookmark: [10.1109/MM.2021.3088396](https://doi.org/10.1109/MM.2021.3088396)



SmidgION from ONT

COVID-19 Nanopore Sequencing (I)



COVID-19 Nanopore Sequencing (II)

How are scientists using nanopore sequencing to research COVID-19?



Samples are collected

Validated SARS-CoV-2 RT-PCR test performed



SARS-CoV-2 positive samples



SARS-CoV-2 negative samples: used as negative controls

How can this be used?
Genomic epidemiology: analyse variants & mutation rate, track spread of virus, identify clusters of transmission

What are the results?
From RNA to full SARS-CoV-2 consensus sequence in ~7 hours

How?
Targeted amplification of SARS-CoV-2 genome + multiplexed, rapid nanopore sequencing

Targeted SARS-CoV-2 nanopore sequencing



Metagenomic nanopore sequencing

How?
1 x RNA metagenomic sequencing run
1 x DNA metagenomic sequencing run

What are the results?
RNA: data for RNA viruses (including SARS-CoV-2) + microbial transcripts
DNA: data for bacteria + DNA viruses

How can this be used?
Characterise co-infecting bacteria & viruses, identify any correlation of risk factors, research potential future treatment implications

SARS-CoV-2 Direct RNA whole genome sequencing: assess viral genome in its native RNA form and the effect of base modifications

Immune repertoire: assess response of the immune system to SARS-CoV-2 infection by sequencing of full-length immune cell receptor genes and transcripts

Whole human genome sequencing: investigate what might cause different responses to the virus in different people based on their genome

What's next?



Find out more at nanoporetech.com/covid19

MinION™

GridION™

PromethION™

Oxford Nanopore Technologies, the Wheel icon, GridION, PromethION and MiniION are registered trademarks of Oxford Nanopore Technologies in various countries. © 2020 Oxford Nanopore Technologies. All rights reserved. Oxford Nanopore Technologies' products are currently for research use only. IG_1061(EN)_V1_03April2020

Accelerating Genome Analysis: Overview

- Mohammed Alser, Zülal Bingöl, Damla Senol Cali, Jeremie Kim, Saugata Ghose, Can Alkan, and Onur Mutlu,
"Accelerating Genome Analysis: A Primer on an Ongoing Journey"
IEEE Micro (IEEE MICRO), Vol. 40, No. 5, pages 65-75, September/October 2020.
[\[Slides \(pptx\)\(pdf\)\]](#)
[\[Talk Video \(1 hour 2 minutes\)\]](#)

Accelerating Genome Analysis: A Primer on an Ongoing Journey

Mohammed Alser
ETH Zürich

Zülal Bingöl
Bilkent University

Damla Senol Cali
Carnegie Mellon University

Jeremie Kim
ETH Zurich and Carnegie Mellon University

Saugata Ghose
University of Illinois at Urbana–Champaign and
Carnegie Mellon University

Can Alkan
Bilkent University

Onur Mutlu
ETH Zurich, Carnegie Mellon University, and
Bilkent University

Beginner Reading on Genome Analysis

Mohammed Alser, Joel Lindegger, Can Firtina, Nour Almadhoun, Haiyu Mao, Gagandeep Singh, Juan Gomez-Luna, Onur Mutlu

"From Molecules to Genomic Variations to Scientific Discovery: Intelligent Algorithms and Architectures for Intelligent Genome Analysis"

Computational and Structural Biotechnology Journal, 2022

[\[Source code\]](#)



ELSEVIER



journal homepage: www.elsevier.com/locate/csbj



Review

From molecules to genomic variations: Accelerating genome analysis via intelligent algorithms and architectures



Mohammed Alser*, Joel Lindegger, Can Firtina, Nour Almadhoun, Haiyu Mao, Gagandeep Singh, Juan Gomez-Luna, Onur Mutlu*

ETH Zurich, Gloriastrasse 35, 8092 Zürich, Switzerland

<https://arxiv.org/pdf/2205.07957.pdf>

Many Interesting Things
Are Happening Today
in Computer Architecture

More Demanding Workloads

The Problem

Computing

is Bottlenecked by Data

Data is Key for AI, ML, Genomics, ...

- Important workloads are all data intensive
- They require rapid and efficient processing of large amounts of data
- Data is increasing
 - We can generate more than we can process

Data is Key for Future Workloads



In-memory Databases

[Mao+, EuroSys'12;
Clapp+ (Intel), IISWC'15]



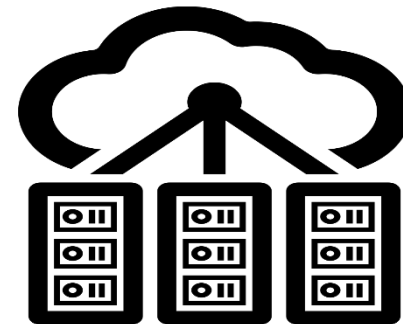
In-Memory Data Analytics

[Clapp+ (Intel), IISWC'15;
Awan+, BDCloud'15]



Graph/Tree Processing

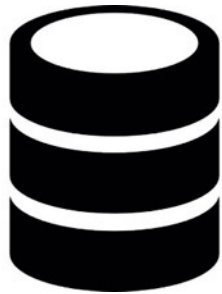
[Xu+, IISWC'12; Umuroglu+, FPL'15]



Datacenter Workloads

[Kanev+ (Google), ISCA'15]

Data Overwhelms Modern Machines



In-memory Databases



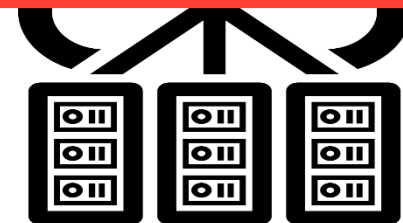
Graph/Tree Processing

Data → performance & energy bottleneck



In-Memory Data Analytics

[Clapp+ (Intel), IISWC'15;
Awan+, BDCloud'15]



Datacenter Workloads

[Kanev+ (Google), ISCA'15]

Data is Key for Future Workloads



Chrome

Google's web browser



TensorFlow Mobile

Google's machine learning framework

VP9



Video Playback

Google's **video codec**

VP9



Video Capture

Google's **video codec**

Data Overwhelms Modern Machines



Chrome



TensorFlow Mobile

Data → performance & energy bottleneck

VP9



Video Playback

Google's **video codec**

VP9



Video Capture

Google's **video codec**

Data Movement Overwhelms Modern Machines

- Amirali Boroumand, Saugata Ghose, Youngsok Kim, Rachata Ausavarungnirun, Eric Shiu, Rahul Thakur, Daehyun Kim, Aki Kuusela, Allan Knies, Parthasarathy Ranganathan, and Onur Mutlu, "[Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks](#)" *Proceedings of the 23rd International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, Williamsburg, VA, USA, March 2018.

62.7% of the total system energy
is spent on **data movement**

Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks

Amirali Boroumand¹

Saugata Ghose¹

Youngsok Kim²

Rachata Ausavarungnirun¹

Eric Shiu³

Rahul Thakur³

Daehyun Kim^{4,3}

Aki Kuusela³

Allan Knies³

Parthasarathy Ranganathan³

Onur Mutlu^{5,1}

Data Movement Overwhelms Accelerators

- Amirali Boroumand, Saugata Ghose, Berkin Akin, Ravi Narayanaswami, Geraldo F. Oliveira, Xiaoyu Ma, Eric Shiu, and Onur Mutlu,
["Google Neural Network Models for Edge Devices: Analyzing and Mitigating Machine Learning Inference Bottlenecks"](#)
Proceedings of the 30th International Conference on Parallel Architectures and Compilation Techniques (PACT), Virtual, September 2021.
[[Slides \(pptx\)](#)] [[pdf](#)]
[[Talk Video](#) (14 minutes)]

> 90% of the total system energy is spent on memory in large ML models

Google Neural Network Models for Edge Devices: Analyzing and Mitigating Machine Learning Inference Bottlenecks

Amirali Boroumand^{†◇}
Geraldo F. Oliveira^{*}

Saugata Ghose[‡]
Xiaoyu Ma[§]

Berkin Akin[§]
Eric Shiu[§]

Ravi Narayanaswami[§]
Onur Mutlu^{*†}

[†]Carnegie Mellon Univ.

[◇]Stanford Univ.

[‡]Univ. of Illinois Urbana-Champaign

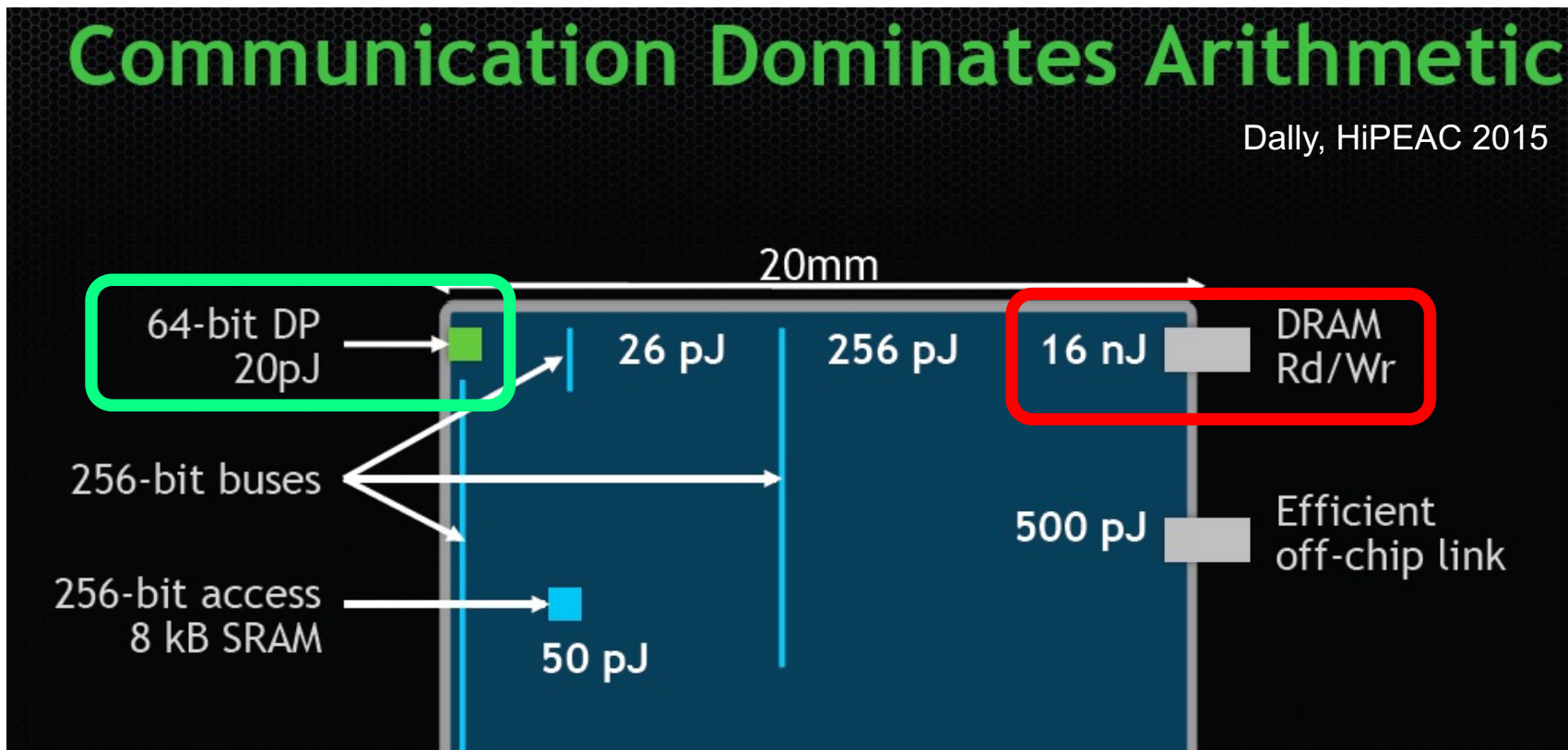
[§]Google

^{*}ETH Zürich

Data Movement vs. Computation Energy

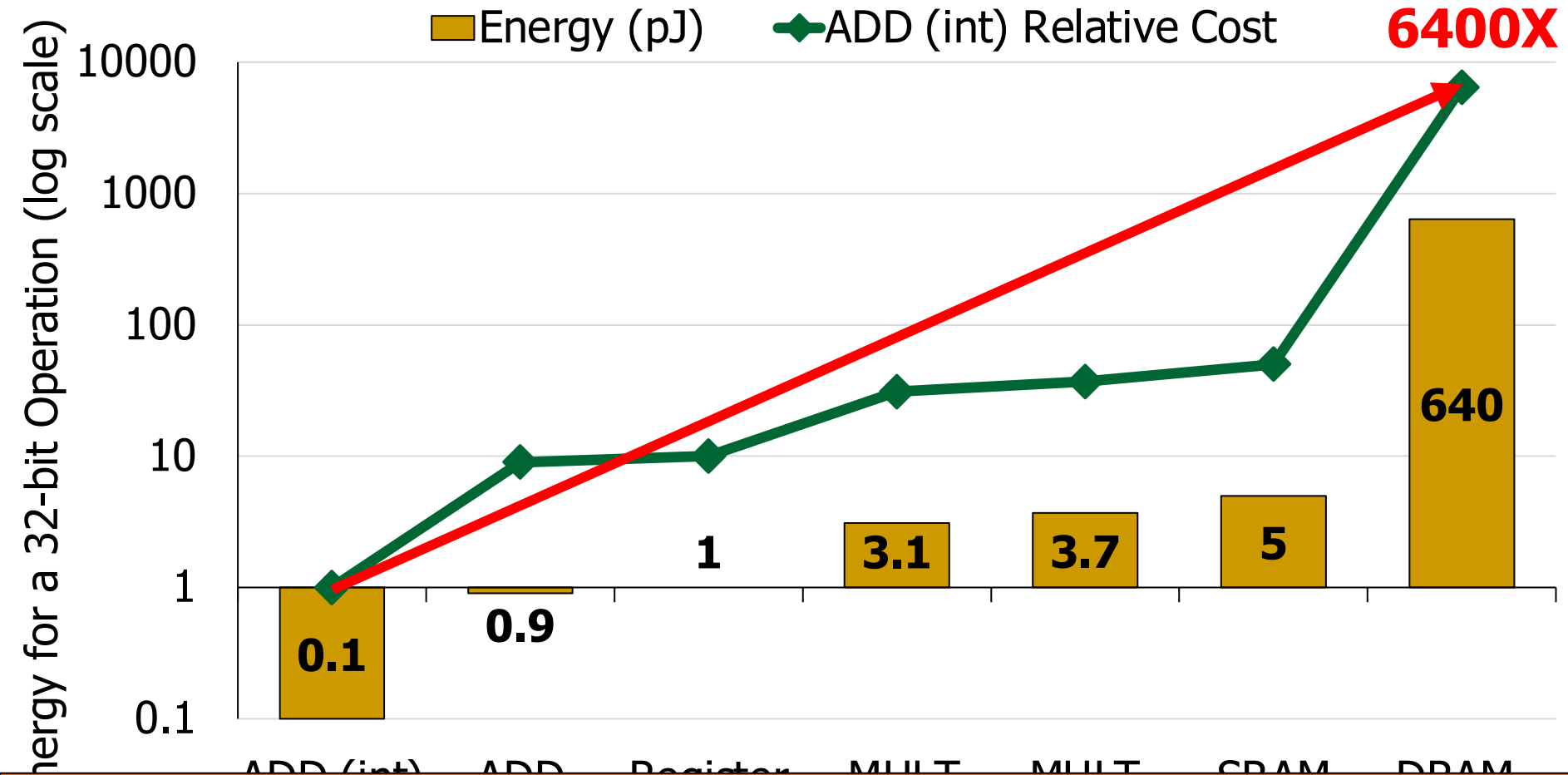
Communication Dominates Arithmetic

Dally, HiPEAC 2015



A memory access consumes $\sim 100-1000X$ the energy of a complex addition

Data Movement vs. Computation Energy



A memory access consumes 6400X
the energy of a simple integer addition

Many Interesting Things
Are Happening Today
in Computer Architecture

Many Novel Concepts Investigated Today

- **New Computing Paradigms (Rethinking the Full Stack)**
 - Processing in Memory, Processing Near Data
 - Neuromorphic Computing, Quantum Computing
 - Fundamentally Secure and Dependable Computers
- **New Accelerators & Systems (Algorithm-Hardware Co-Designs)**
 - Artificial Intelligence & Machine Learning
 - Graph & Data Analytics, Vision, Video
 - Genome Analysis
- **New Memories, Storage Systems, Interconnects, Devices**
 - Non-Volatile Main Memory, Intelligent Memory Systems, Quantum
 - High-Speed Interconnects, Disaggregated Systems

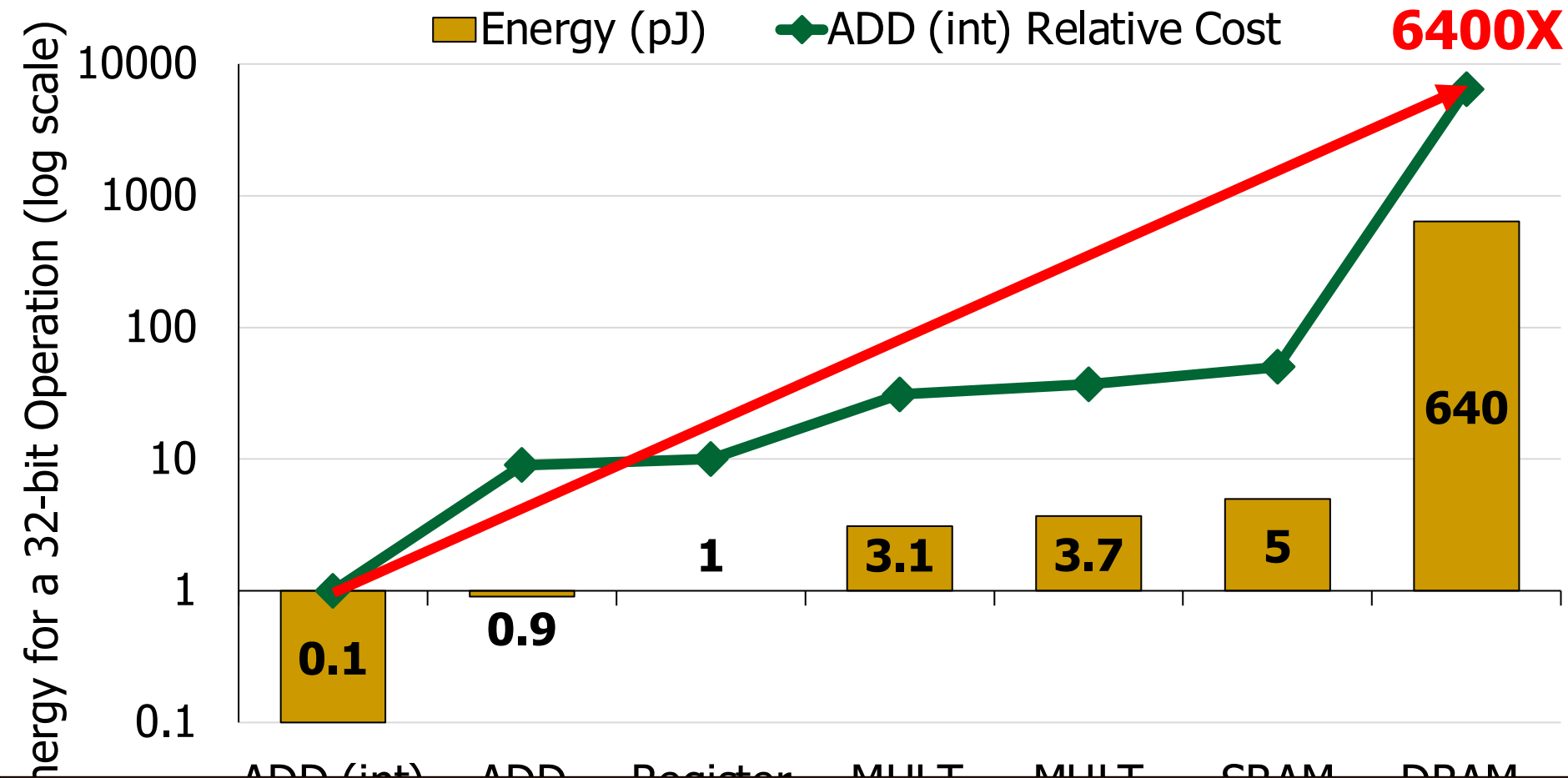
Increasingly Demanding Applications

Dream

and, they will come

As applications push boundaries, computing platforms will become increasingly strained.

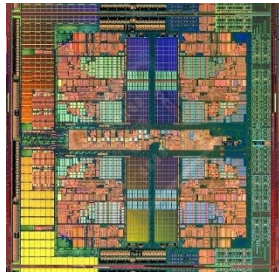
Increasingly Diverging/Complex Tradeoffs



A memory access consumes 6400X
the energy of a simple integer addition

Increasingly Complex Systems

Past systems



Microprocessor



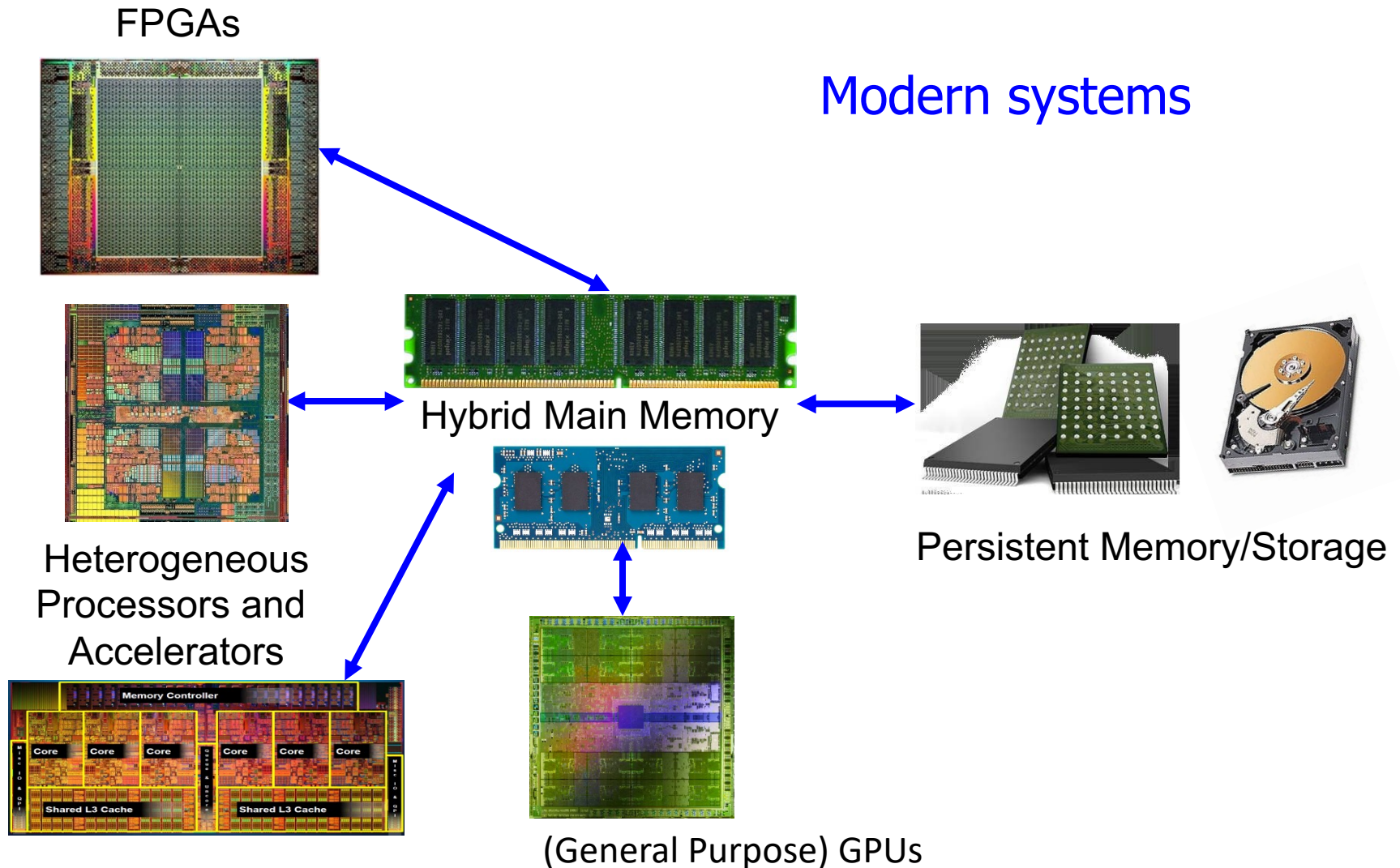
Main Memory



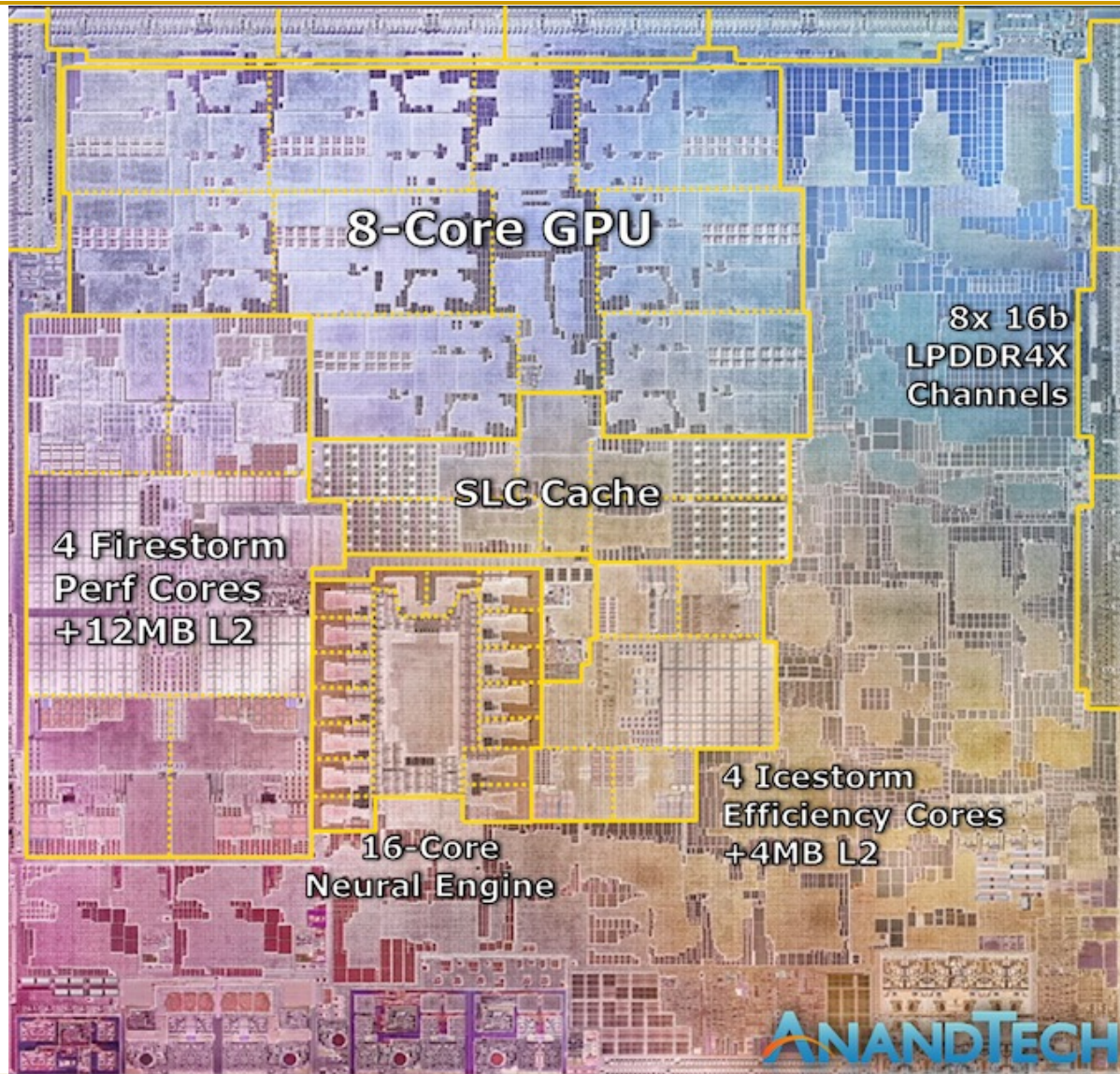
Storage (SSD/HDD)



Increasingly Complex Systems



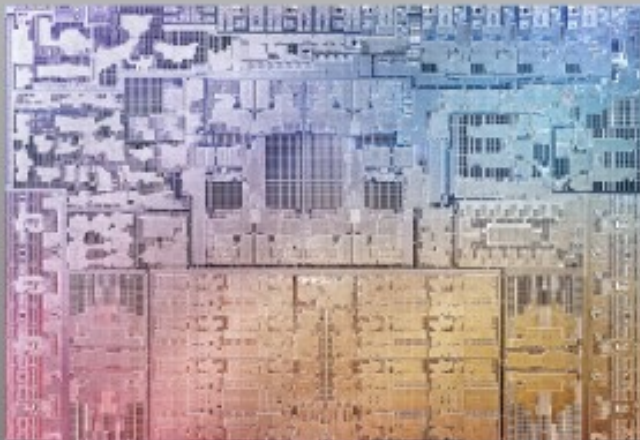
Increasingly Complex Systems on Chip



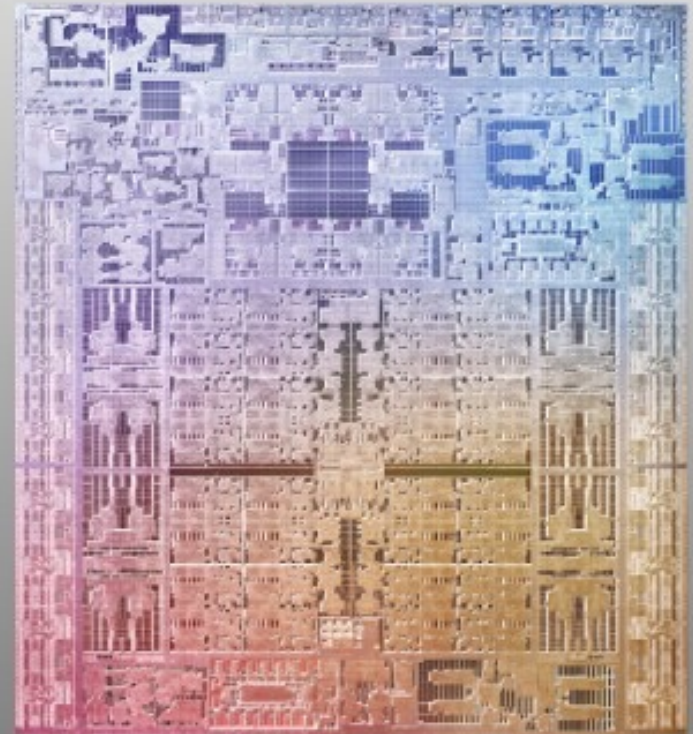
Bigger and More Powerful Systems (2021)



Apple M1



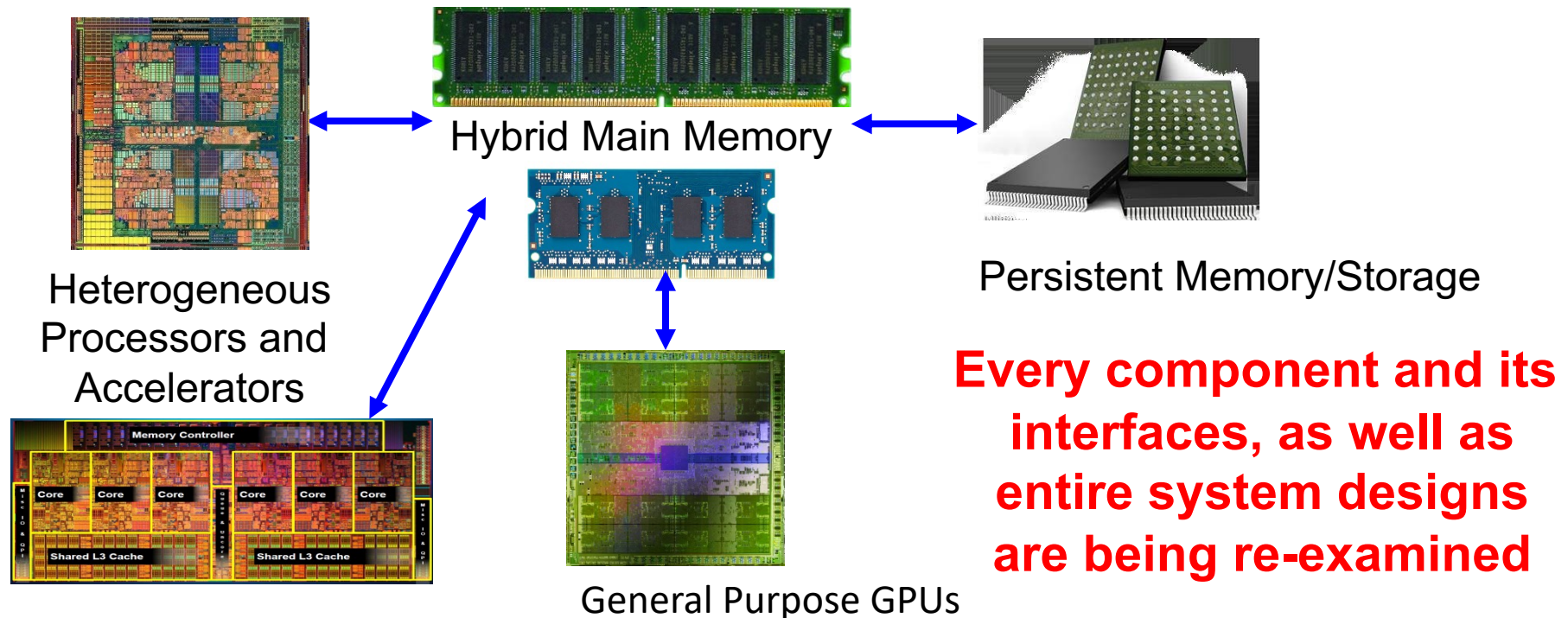
Apple M1 Pro



Apple M1 Max

Computer Architecture Today

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



Computer Architecture Today (II)

- You can revolutionize the way computers are built, if you understand both the hardware and the software (and change each accordingly)
- You can invent new paradigms for computation, communication, and storage
- Recommended book: Thomas Kuhn, “[The Structure of Scientific Revolutions](#)” (1962)
 - Pre-paradigm science: no clear consensus in the field
 - Normal science: dominant theory used to explain/improve things (business as usual); exceptions considered anomalies
 - Revolutionary science: underlying assumptions re-examined

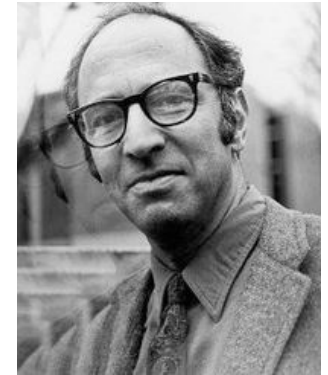
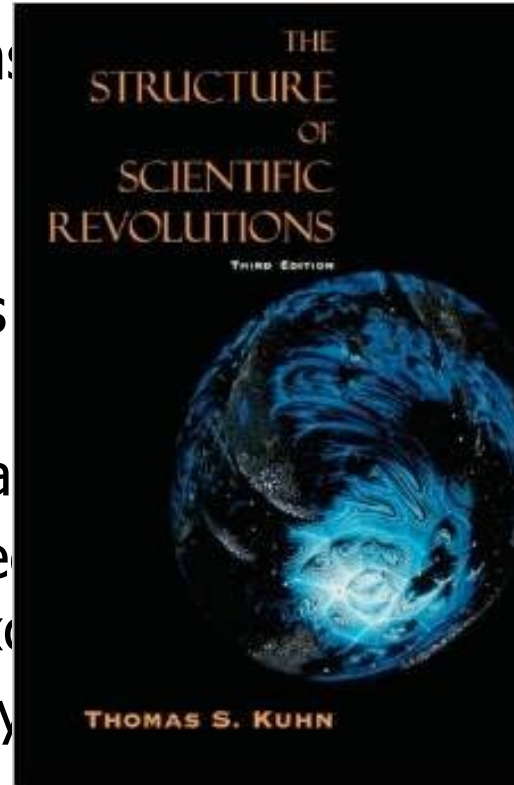
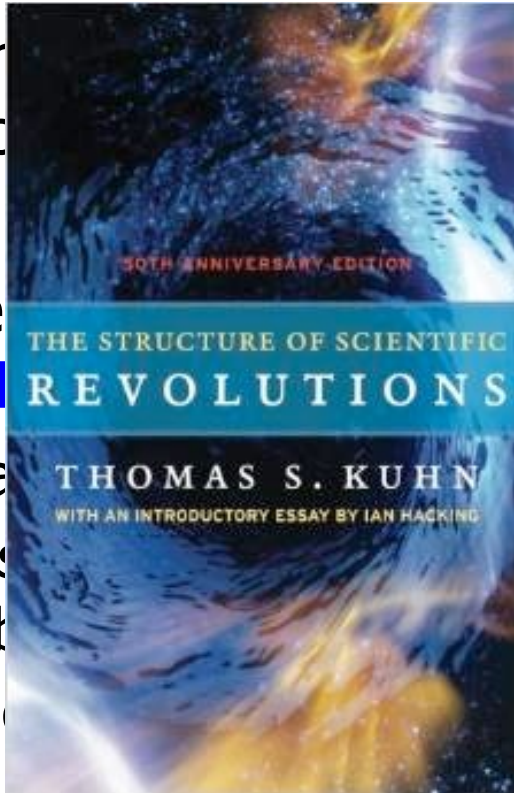
Computer Architecture Today (II)

- You can revolutionize the way computers are built, if you understand both the hardware and the software (and change each accordingly)

- You can improve communication

- Recommended reading: **Scientific Revolutions**

- Pre-para
- Normal s
- things (b
- Revoluti



ure of

eld
improve
anomalies
examined

Takeaways

- It is an exciting time to be understanding and designing computing architectures
- Many challenging and exciting problems in platform design
 - That no one has tackled (or thought about) before
 - That can have huge impact on the world's future
- Driven by huge hunger for data (Big Data), new applications (ML/AI, graph analytics, genomics), ever-greater realism, ...
 - We can easily collect more data than we can analyze/understand
- Driven by significant difficulties in keeping up with that hunger at the technology layer
 - Five walls: Energy, reliability, complexity, security, scalability

Let's Start with Some Puzzles

a.k.a. Computer Architecture resembles Building Architecture

What Is This?



What About This?



What Do the Following
Have in Common?

Gare do Oriente, Lisbon



Milwaukee Art Museum



Athens Olympic Stadium



City of Arts and Sciences, Valencia



Florida Polytechnic University (I)



Oculus, New York City



What do All Those Have in Common
with Bahnhof Stadelhofen?

Answer: All Designed by a Famous Architect

- ETH Alumnus, PhD Civil Engineering
- “The train station has several of the features that became signatures of his work; straight lines and right angles are rare.”



Santiago Calatrava Valls (born 28 July 1951) is a Spanish [architect](#), [structural engineer](#), [sculptor](#) and [painter](#), particularly known for his bridges supported by single leaning pylons, and his railway stations, stadiums, and museums, whose sculptural forms often resemble living organisms.^[1] His best-known works include the [Milwaukee Art Museum](#), the [Turning Torso](#) tower in [Malmo](#), Sweden, the [Margaret Hunt Hill Bridge](#) in [Dallas](#), Texas, and the [Museum of Tomorrow](#) in [Rio de Janeiro](#),

Your First Comp. Architecture Assignment

- Go and find the closest Calatrava building to this classroom
 - For those who like a challenge, find the furthest building that was designed by Calatrava to his classroom 😊
- Appreciate the beauty & out-of-the-box and creative thinking
- Think about tradeoffs in the design
 - Strengths, weaknesses, goals of design
- Derive principles on your own for good design and innovation
- Due date: **Any time during or after this course**
 - Later during the course is better
 - Apply what you have learned in this course
 - Think out-of-the-box

But First, Today's First Assignment

Find The Differences of This and That

This



That



Many Tradeoffs Between Two Designs

- You can list them after you complete the first assignment...

Aside: Evaluation Criteria for the Designs

- Functionality (Does it meet the specification?)
 - Reliability
 - Space requirement
 - Cost
 - Expandability
 - Comfort level of users
 - Happiness level of users
 - Aesthetics
 - Security
 - ...
-
- How to evaluate goodness of design is always a critical question → "Performance" evaluation and metrics

A Key Question

- How was Calavatra able to design especially his key buildings?
 - Can have many guesses
 - (Very) hard work, perseverance, dedication (over decades)
 - Experience
 - Creativity, Out-of-the-box thinking
 - A good understanding of past designs
 - Good judgment and intuition
 - Strong skill combination (math, architecture, art, engineering, ...)
 - Funding (\$\$\$\$), luck, initiative, entrepreneurialism
 - Strong understanding of and commitment to fundamentals
 - Principled design
 - ...
 - You will be exposed to and hopefully develop/enhance many of these skills in this course
-

Principled Design

- “To me, there are **two overriding principles** to be found in nature which are most appropriate for building:
 - one is the **optimal use of material**,
 - the other **the capacity of organisms to change shape, to grow, and to move.**”
 - *Santiago Calatrava*

- “Calatrava's constructions are inspired by natural forms like plants, bird wings, and the human body.”

Gare do Oriente, Lisbon, Revisited



Source: By Martín Gómez Tagle - Lisbon, Portugal, CC BY-SA 3.0, <https://commons.wikimedia.org/w/index.php?curid=13764903>

Source: <http://www.arcspace.com/exhibitions/unsorted/santiago-calatrava/>

A Principled Design

Zoomorphic architecture

From Wikipedia, the free encyclopedia

Zoomorphic architecture is the practice of using animal forms as the inspirational basis and blueprint for architectural design. "While animal forms have always played a role adding some of the deepest layers of meaning in architecture, it is now becoming evident that a new strand of **biomorphism** is emerging where the meaning derives not from any specific representation but from a more general allusion to biological processes."^[1]

Some well-known examples of Zoomorphic architecture can be found in the [TWA Flight Center](#) building in [New York City](#), by [Eero Saarinen](#), or the [Milwaukee Art Museum](#) by [Santiago Calatrava](#), both inspired by the form of a bird's wings.^[3]

What Does This Remind You Of?



The Architect's Answer

Design [\[edit \]](#)

Calatrava said that the Oculus resembles a bird being released from a child's hand. The roof was originally designed to mechanically open to increase light and ventilation to the enclosed space. [Herbert Muschamp](#), architecture critic of *The New York Times*, compared the design to the [Bethesda Terrace and Fountain in Central Park](#), and wrote in 2004:

Strengths and Praise

“ Santiago Calatrava's design for the World Trade Center PATH station should satisfy those who believe that buildings planned for ground zero must aspire to a spiritual dimension. Over the years, many people have discerned a metaphysical element in Mr. Calatrava's work. I hope New Yorkers will detect its presence, too. With deep appreciation, I congratulate the Port Authority for commissioning Mr. Calatrava, the great Spanish architect and engineer, to design a building with the power to shape the future of New York. It is a pleasure to report, for once, that public officials are not overstating the case when they describe a design as breathtaking.^[43]

”

Design Constraints and Criticism

However, Calatrava's original soaring spike design was scaled back because of security issues. The *New York Times* observed in 2005:

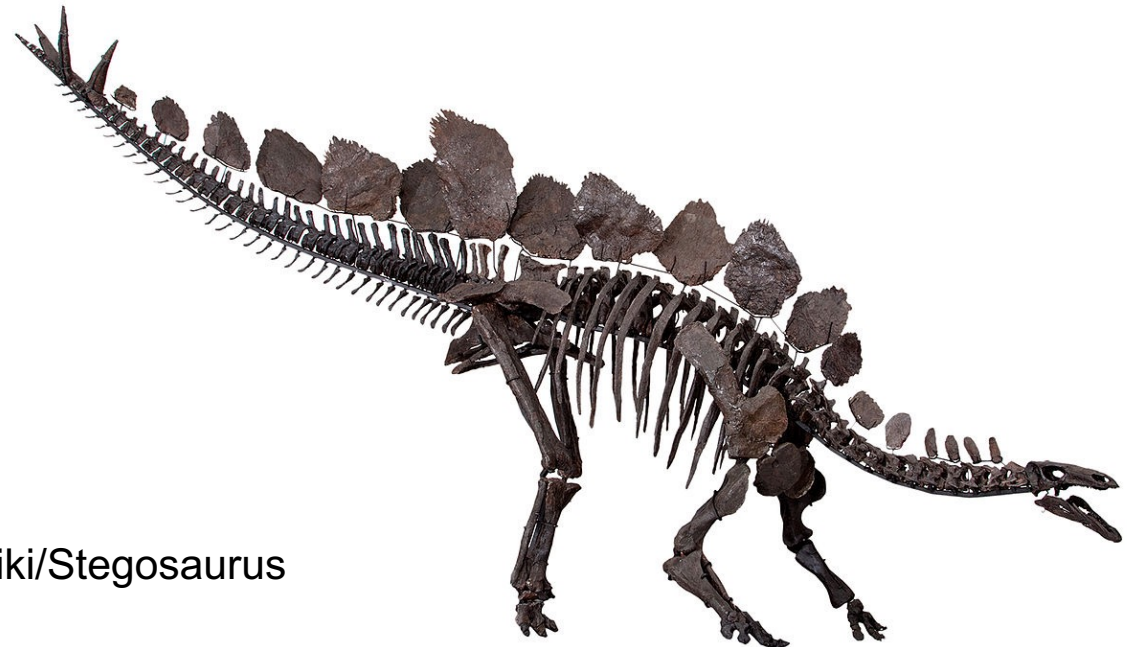
“ In the name of security, Santiago Calatrava's bird has grown a beak. Its ribs have doubled in number and its wings have lost their interstices of glass.... [T]he main transit hall, between Church and Greenwich Streets, will almost certainly lose some of its delicate quality, while gaining structural expressiveness. It may now evoke a slender *stegosaurus* more than it does a bird.^[45] ”

Stegosaurus

From Wikipedia, the free encyclopedia

For the *pachycephalosaurid* of a similar name, see *Stegoceras*.

Stegosaurus (/ˌstɛɡəˈsɔːrəs/^[1]) is a genus of armored dinosaur. Fossils of this genus date to the Late Jurassic period, where they are found in Kimmeridgian to early Tithonian aged strata, between 155 and 150 million years ago, in the western United States and Portugal. Several



Source: <https://en.wikipedia.org/wiki/Stegosaurus>

Susannah Maidment et al. & Natural History Museum, London – Maidment SCR, Brassey C, Barrett PM (2015) The Postcranial Skeleton of an Exceptionally Complete Individual of the Plated Dinosaur *Stegosaurus stenops* (Dinosauria: Thyreophora) from the Upper Jurassic Morrison Formation of Wyoming, U.S.A. PLoS ONE 10(10): e0138352. doi:10.1371/journal.pone.0138352

Design Constraints: Noone is Immune

However, Calatrava's original soaring spike design was scaled back because of security issues. The *New York Times* observed in 2005:

“ In the name of security, Santiago Calatrava's bird has grown a beak. Its ribs have doubled in number and its wings have lost their interstices of glass.... [T]he main transit hall, between Church and Greenwich Streets, will almost certainly lose some of its delicate quality, while gaining structural expressiveness. It may now evoke a slender *stegosaurus* more than it does a bird.^[45] ”

The design was further modified in 2008 to eliminate the opening and closing roof mechanism because of budget and space constraints.^[46]

The Transportation Hub has been dubbed "the world's most expensive transportation hub" for its massive cost for reconstruction—\$3.74 billion dollars.^{[48][58]} By contrast, the proposed two-mile PATH extension

The Lecture Was Slightly Different When I Was at CMU

What Is This?



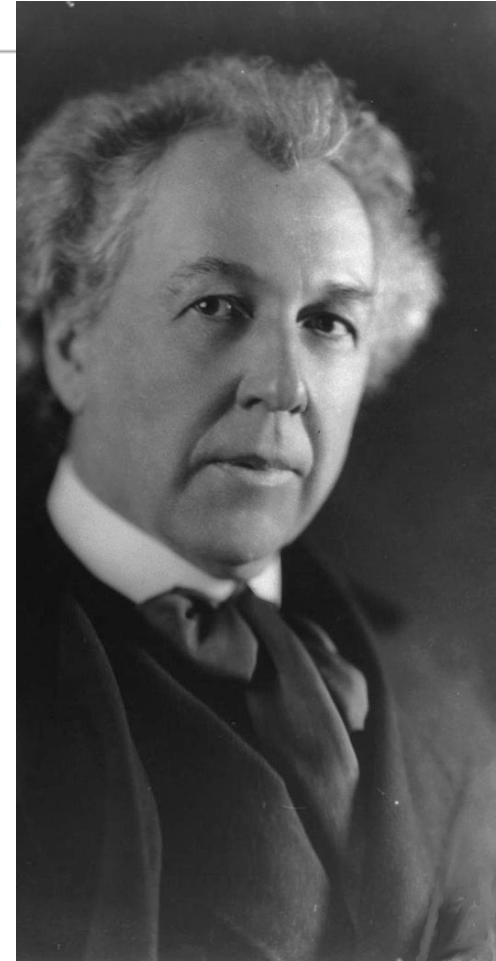
Answer: Masterpiece of A Famous Architect

Fallingwater

From Wikipedia, the free encyclopedia

Fallingwater or **Kaufmann Residence** is a house designed by architect [Frank Lloyd Wright](#) in 1935 in rural [southwestern Pennsylvania](#), 43 miles (69 km) southeast of [Pittsburgh](#).^[4] The home was built partly over a waterfall on [Bear Run](#) in the Mill Run section of [Stewart Township, Fayette County, Pennsylvania](#), in the [Laurel Highlands](#) of the [Allegheny Mountains](#).

Time cited it after its completion as Wright's "most beautiful job";^[5] it is listed among *Smithsonian's* Life List of 28 places "to visit before you die."^[6] It was designated a [National Historic Landmark](#) in 1966.^[3] In 1991, members of the [American Institute of Architects](#) named the house the "best all-time work of American architecture" and in 2007, it was ranked twenty-ninth on the [list of America's Favorite Architecture](#) according to the [AIA](#).



Find The Differences of This and That

This



This



That



A Key Question

- How was Wright able to design his masterpiece?
- Can have many guesses
 - (Very) hard work, perseverance, dedication (over decades)
 - Experience
 - Creativity, Out-of-the-box thinking
 - A good understanding of past designs
 - Good judgment and intuition
 - Strong skill combination (math, architecture, art, engineering, ...)
 - Funding (\$\$\$\$), luck, initiative, entrepreneurialism
 - Strong understanding of and commitment to fundamentals
 - Principled design
 - ...
- You will be exposed to and hopefully develop/enhance many of these skills in this course

A Quote from The Architect Himself

- “architecture [...] based upon **principle**, and not upon **precedent**”



A Principled Design

Organic architecture

From Wikipedia, the free encyclopedia

Organic architecture is a [philosophy](#) of [architecture](#) which promotes harmony between human habitation and the natural world through design approaches so sympathetic and well integrated with its site, that buildings, furnishings, and surroundings become part of a unified, interrelated composition.

A well-known example of organic architecture is [Fallingwater](#), the residence Frank Lloyd Wright designed for the Kaufmann family in rural Pennsylvania. Wright had many choices to locate a home on this large site, but chose to place the home directly over the waterfall and creek creating a close, yet noisy dialog with the rushing water and the steep site. The horizontal striations of stone masonry with daring [cantilevers](#) of colored beige concrete blend with native rock outcroppings and the wooded environment.

A Key Question

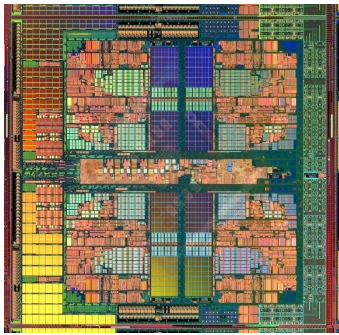
- How was Wright able to design his masterpiece?
- Can have many guesses
 - (Very) hard work, perseverance, dedication (over decades)
 - Experience
 - Creativity, Out-of-the-box thinking
 - A good understanding of past designs
 - Good judgment and intuition
 - Strong skill combination (math, architecture, art, engineering, ...)
 - Funding (\$\$\$\$), luck, initiative, entrepreneurialism
 - Strong understanding of and commitment to fundamentals
 - Principled design
 - ...
- You will be exposed to and hopefully develop/enhance many of these skills in this course

Takeaways

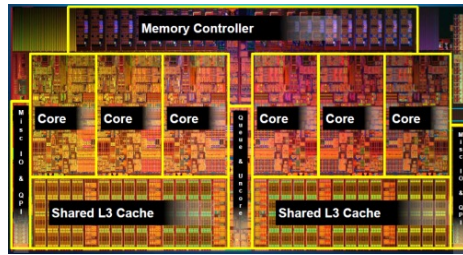
- It all starts from the **basic building blocks and design principles**
- And, **knowledge of how to use, apply, enhance them**
- **Underlying technology might change** (e.g., steel vs. wood)
 - but **methods** of taking advantage of technology **bear resemblance**
 - **methods** used for design **depend on the principles** employed

The Same Applies to Processor Chips

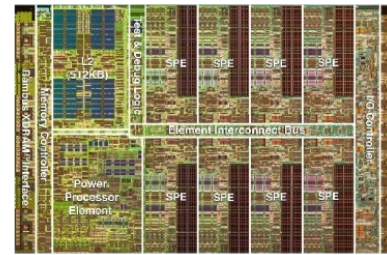
- There are **basic building blocks** and **design principles**



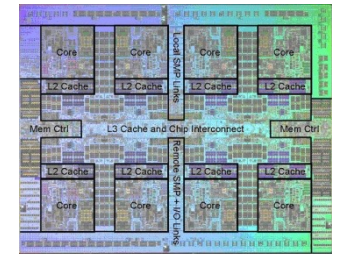
AMD Barcelona
4 cores



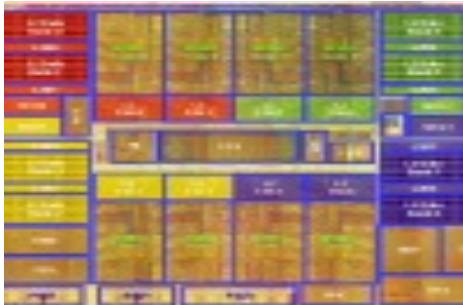
Intel Core i7
8 cores



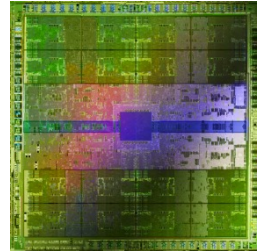
IBM Cell BE
8+1 cores



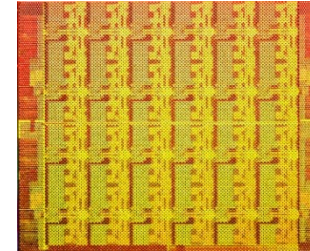
IBM POWER7
8 cores



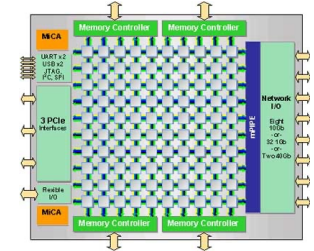
Sun Niagara II
8 cores



Nvidia Fermi
448 "cores"



Intel SCC
48 cores, networked



Tiler TILE Gx
100 cores, networked

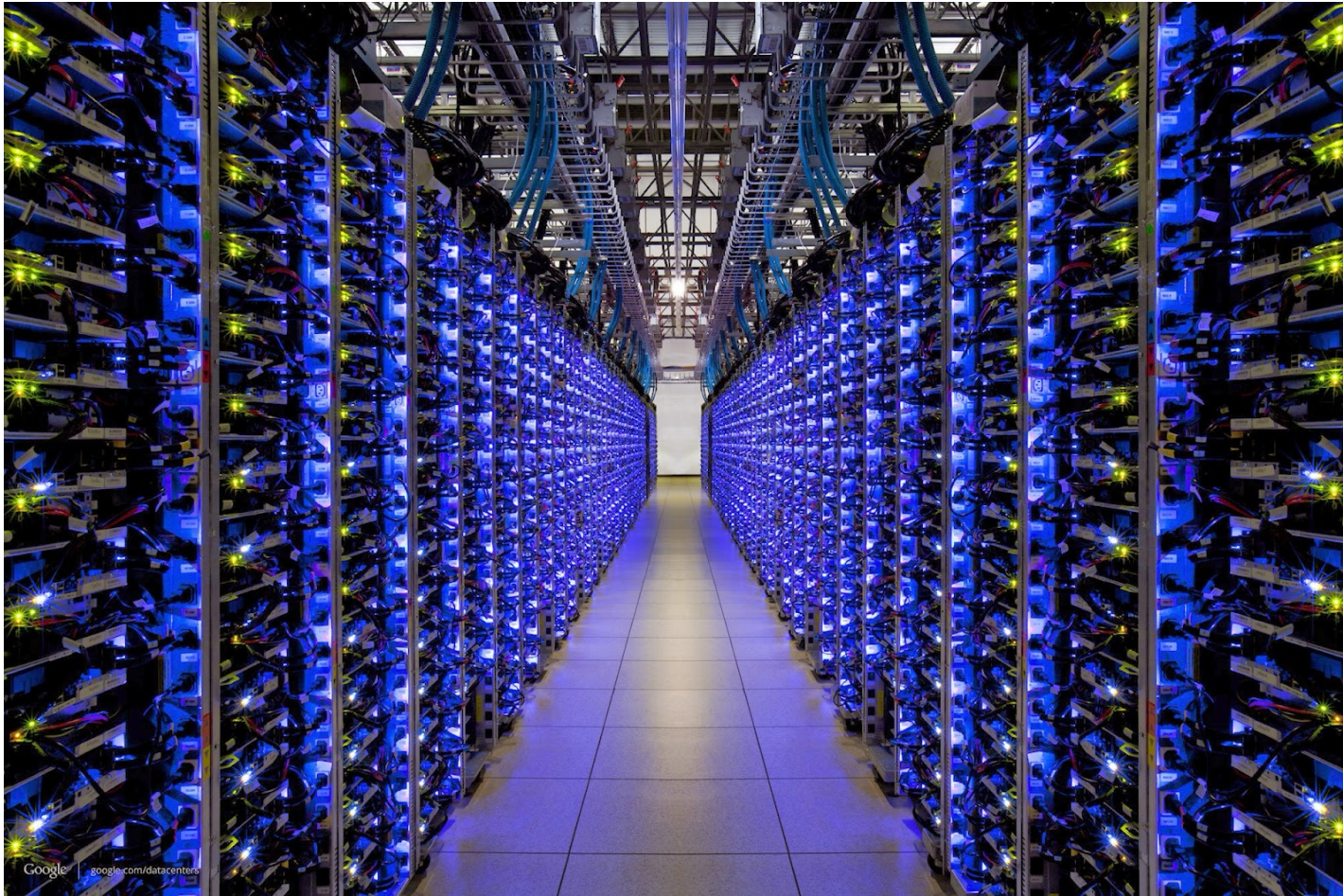
The Same Applies to Computing Systems

- There are **basic building blocks** and **design principles**



The Same Applies to Computing Systems

- There are **basic building blocks** and **design principles**



Different Platforms, Different Goals



Different Platforms, Different Goals



Different Platforms, Different Goals



Different Platforms, Different Goals

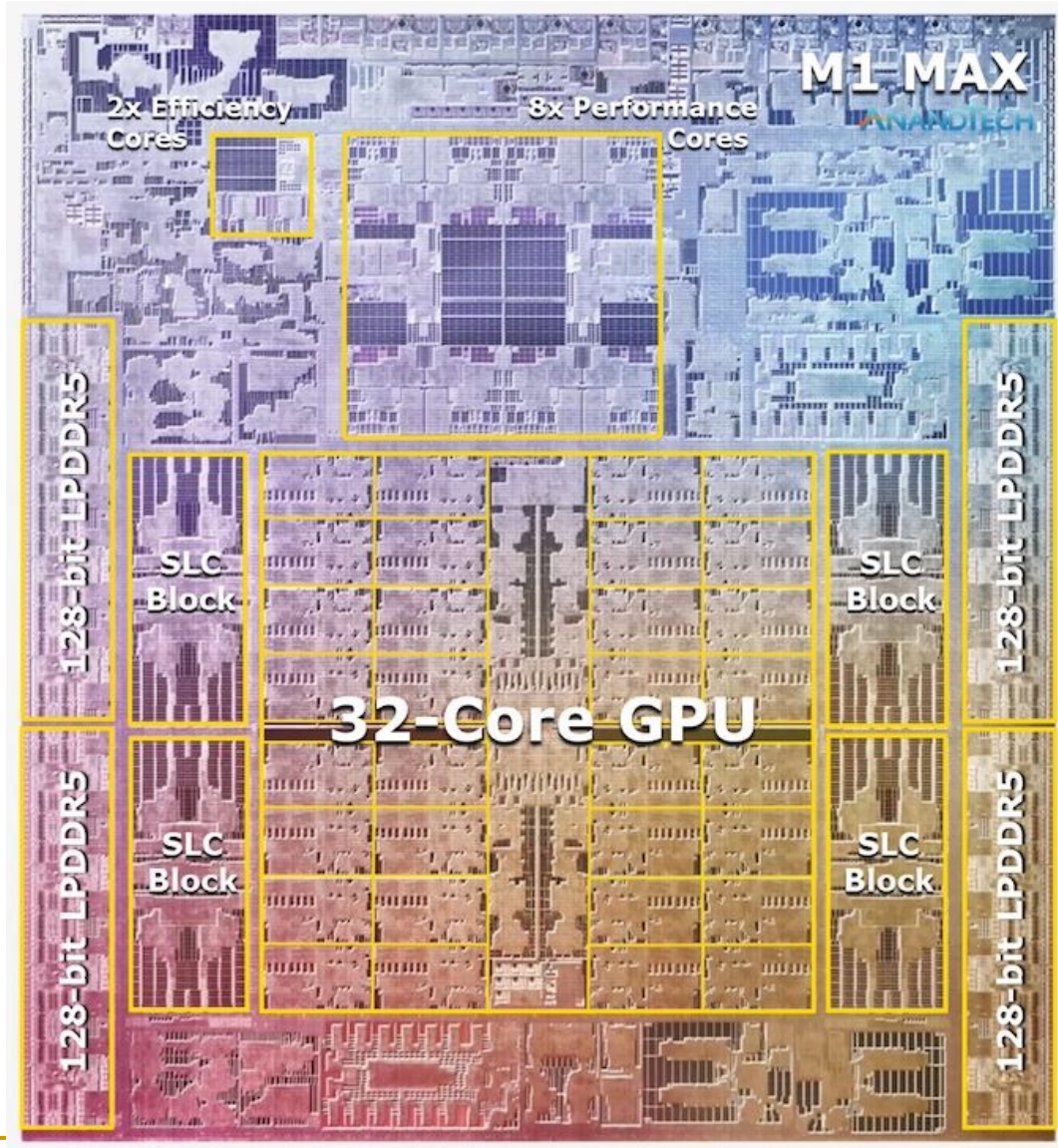


Jack Dongara

Different Platforms, Different Goals



Apple M1 Max System on Chip (2021)



Google Tensor Processing Unit (~2016)

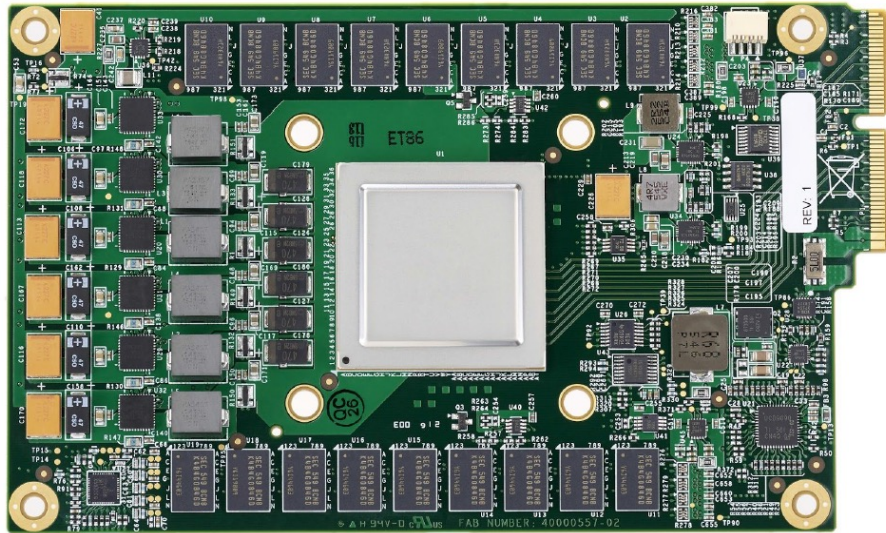


Figure 3. TPU Printed Circuit Board. It can be inserted in the slot for an SATA disk in a server, but the card uses PCIe Gen3 x16.

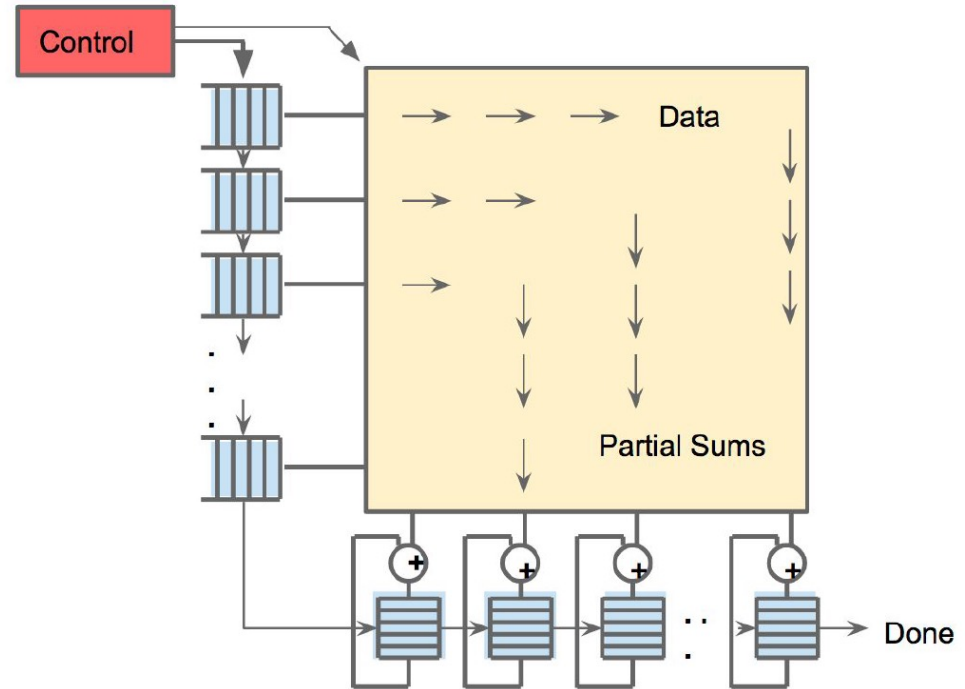
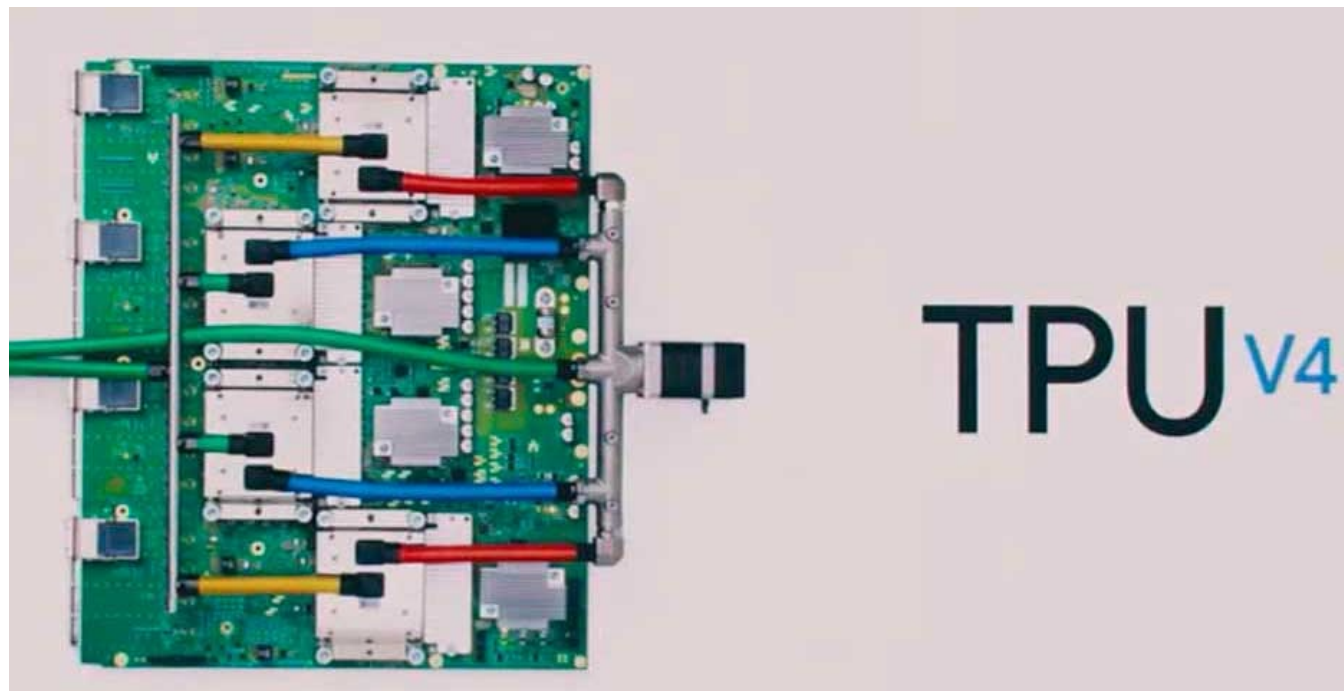


Figure 4. Systolic data flow of the Matrix Multiply Unit. Software has the illusion that each 256B input is read at once, and they instantly update one location of each of 256 accumulator RAMs.

Jouppi et al., “In-Datcenter Performance Analysis of a Tensor Processing Unit”, ISCA 2017.

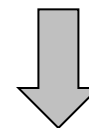
Google TPU Generation IV (2021)



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

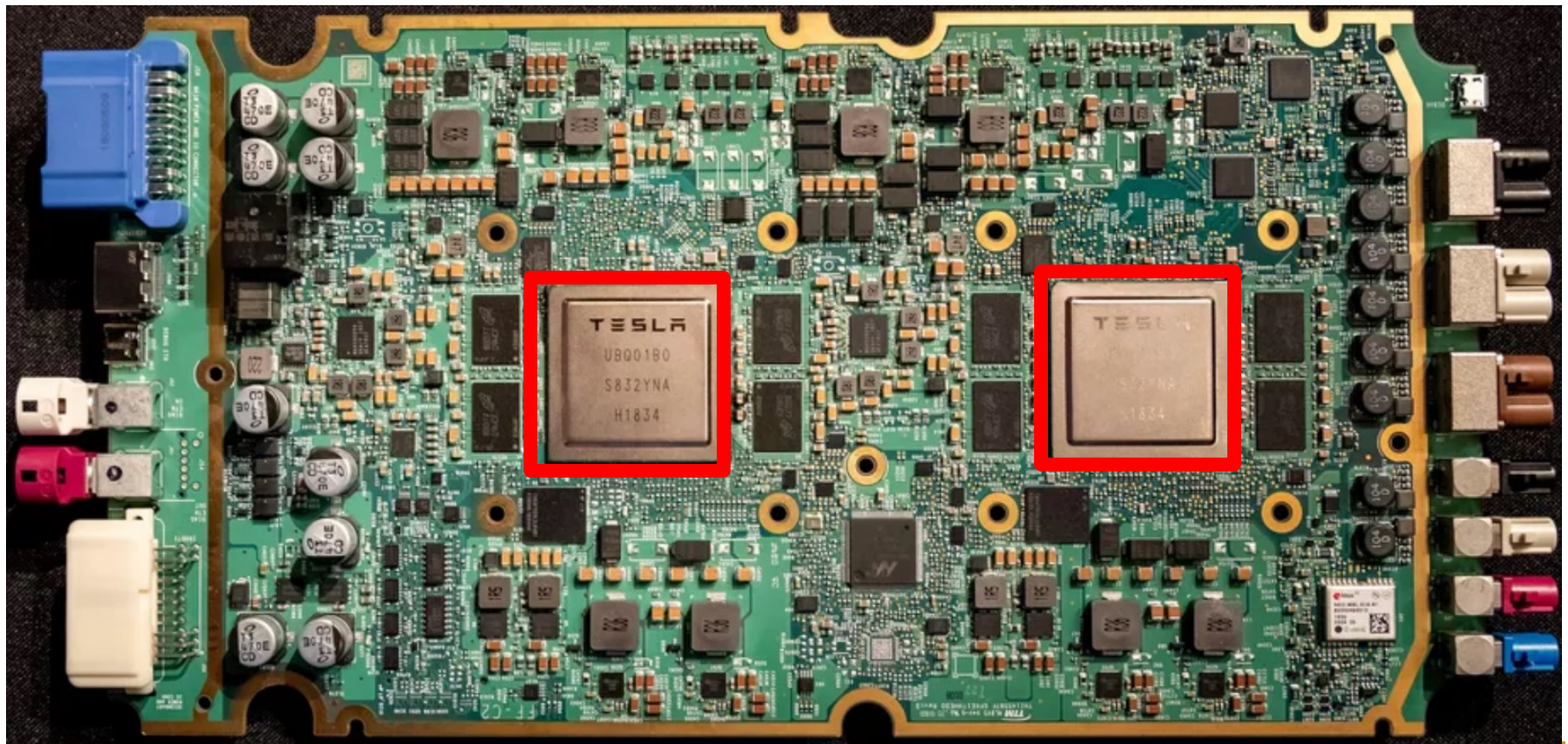


1 ExaFLOPS per board

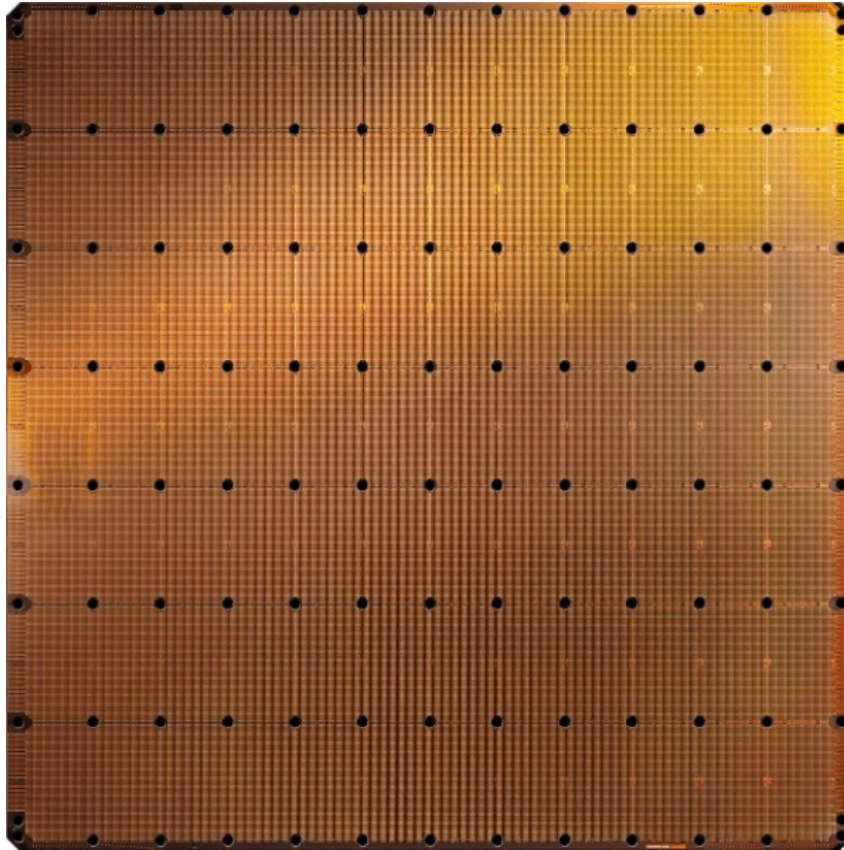
<https://spectrum.ieee.org/tech-talk/computing/hardware/heres-how-googles-tpu-v4-ai-chip-stacked-up-in-training-tests>

TESLA Full Self-Driving Computer (2019)

- ML accelerator: 260 mm², 6 billion transistors, 600 GFLOPS GPU, 12 ARM 2.2 GHz CPUs.
- Two redundant chips for better safety.



Cerebras's Wafer Scale ML Engine-2 (2021)



Cerebras WSE-2
2.6 Trillion transistors
46,225 mm²

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



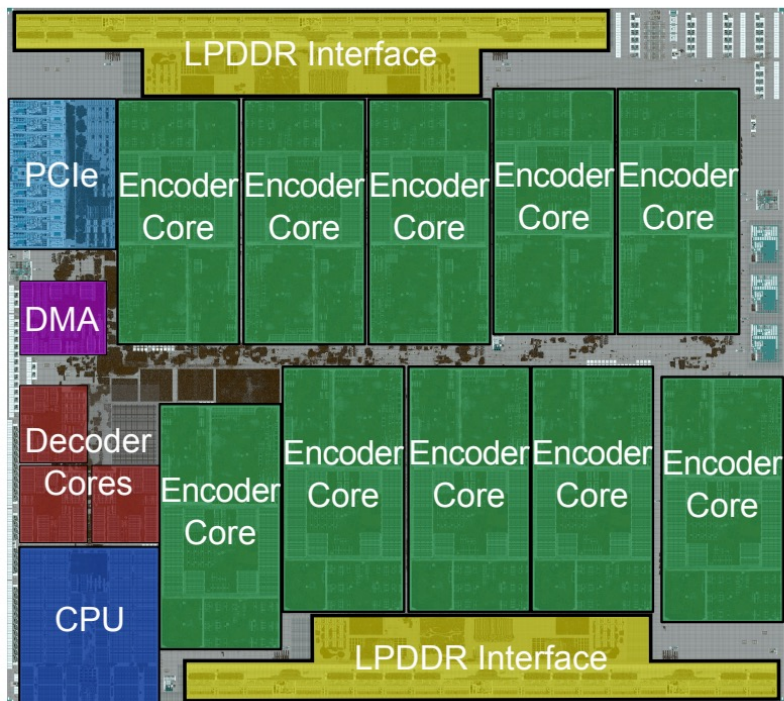
Largest GPU
54.2 Billion transistors
826 mm²
NVIDIA Ampere GA100

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

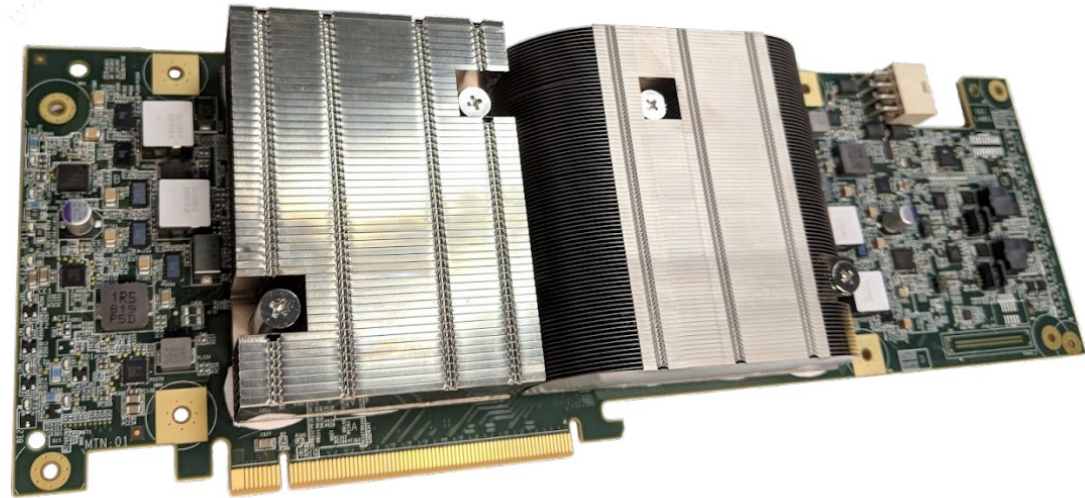
<https://www.cerebras.net/cerebras-wafer-scale-engine-why-we-need-big-chips-for-deep-learning/>

Google's Video Coding Unit (2021)

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



(a) Chip floorplan

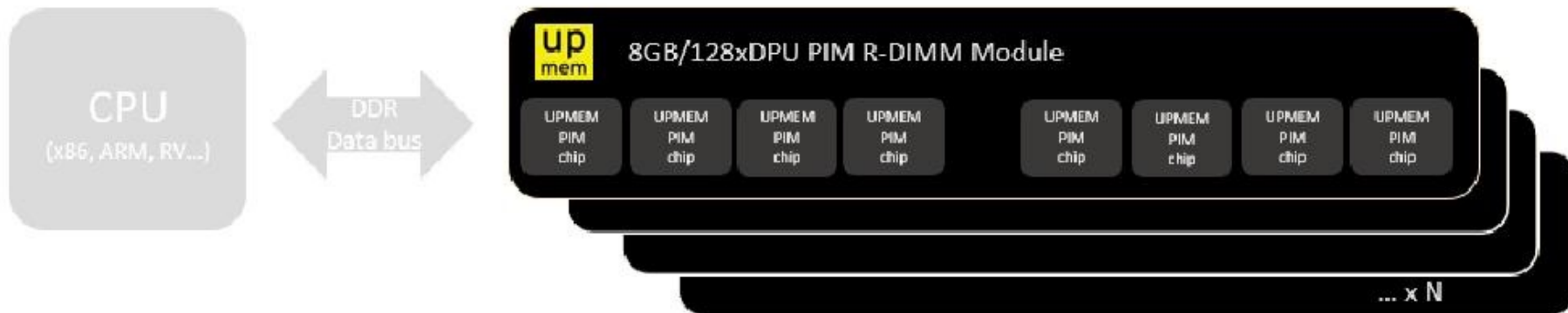


(b) Two chips on a PCBA

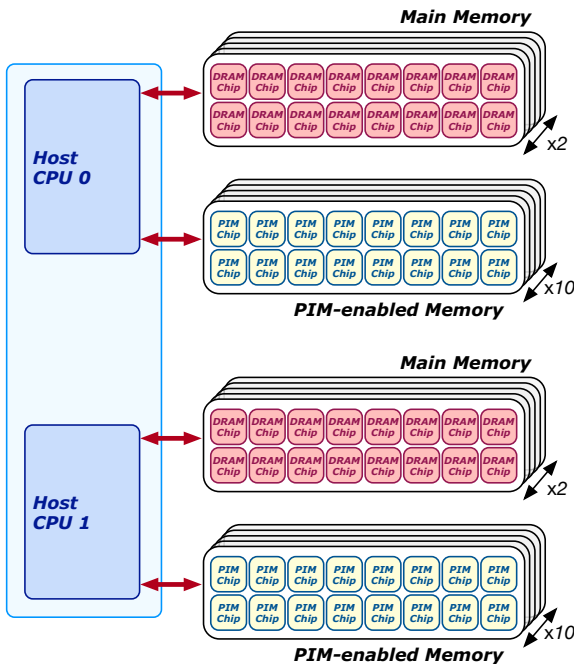
Figure 5: Pictures of the VCU

UPMEM Processing-in-DRAM Engine (2019)

- **Processing in DRAM Engine**
- Includes **standard DIMM modules**, with a **large number of DPU processors** combined with DRAM chips.
- Replaces **standard DIMMs**
 - DDR4 R-DIMM modules
 - 8GB+128 DPUs (16 PIM chips)
 - Standard 2x-nm DRAM process
 - **Large amounts of** compute & memory bandwidth



Different Platforms, Different Goals



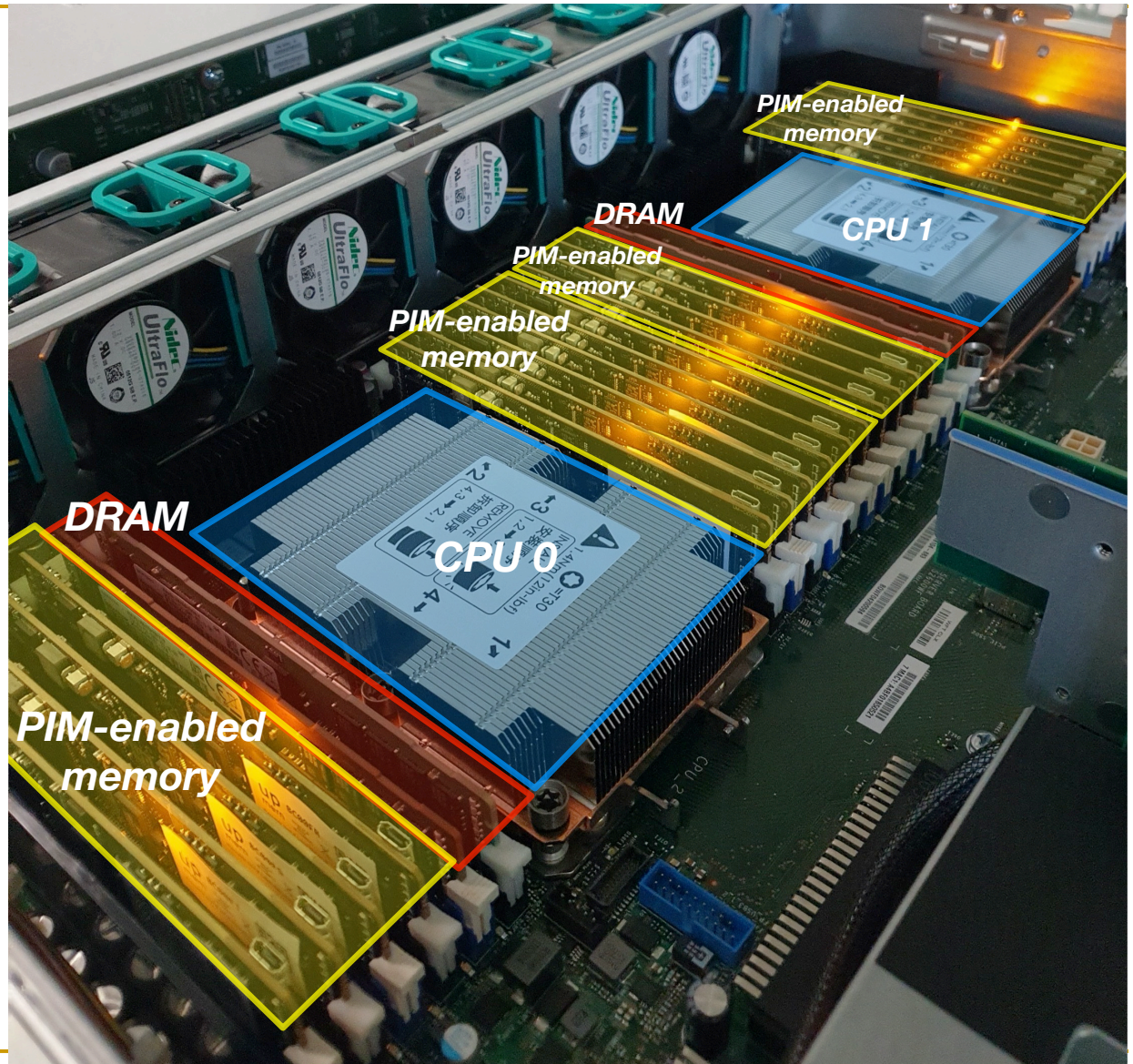
Benchmarking a New Paradigm: An Experimental Analysis of a Real Processing-in-Memory Architecture

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)*.

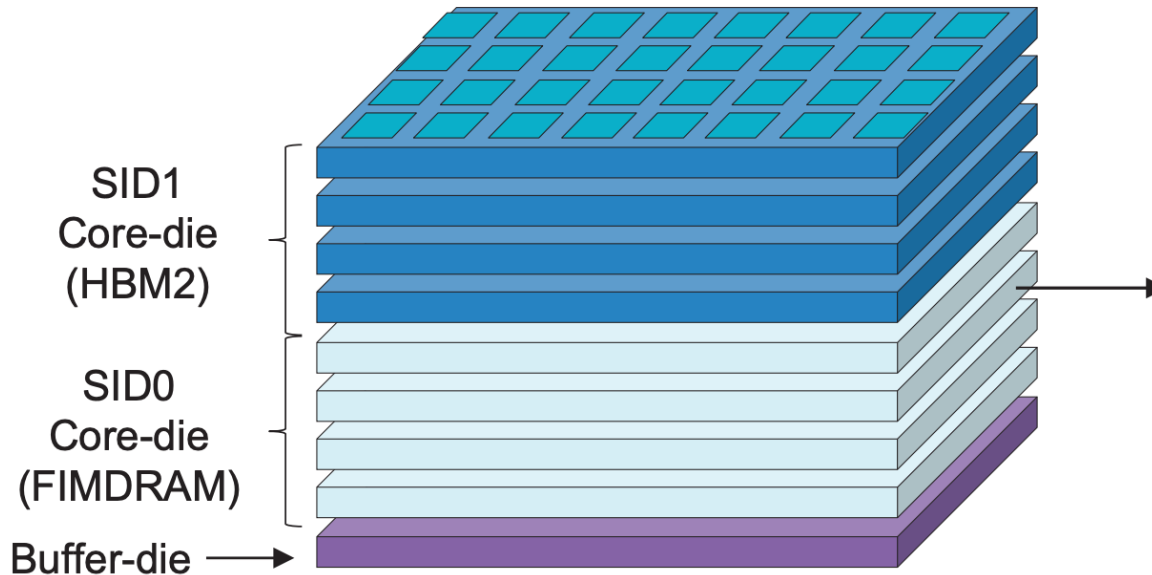
Recent research explores different forms of PIM architectures, motivated by the emergence of new 3D-stacked memory technologies that integrate memory with a logic layer where processing elements can be easily placed. Past works evaluate these architectures in simulation or, at best, with simplified hardware 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 440 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.



Samsung Function-in-Memory DRAM (2021)

■ FIMDRAM based on HBM2



[3D Chip Structure of HBM with FIMDRAM]

Chip Specification

128DQ / 8CH / 16 banks / BL4

32 PCU blocks (1 FIM block/2 banks)

1.2 TFLOPS (4H)

**FP16 ADD /
Multiply (MUL) /
Multiply-Accumulate (MAC) /
Multiply-and- Add (MAD)**

ISSCC 2021 / SESSION 25 / DRAM / 25.4

25.4 A 20nm 6GB Function-In-Memory DRAM, Based on HBM2 with a 1.2TFLOPS Programmable Computing Unit Using Bank-Level Parallelism, for Machine Learning Applications

Young-Cheon Kwon¹, Suk Han Lee¹, Jaehoon Lee¹, Sang-Hyuk Kwon¹, Je Min Ryu¹, Jong-Pil Son¹, Seongil O¹, Hak-Soo Yu¹, Haesuk Lee¹, Soo Young Kim¹, Youngmin Cho¹, Jin Guk Kim¹, Jongyoon Choi¹, Hyun-Sung Shin¹, Jin Kim¹, BengSeng Phuah¹, HyoungMin Kim¹, Myeong Jun Song¹, Ahn Choi¹, Daeho Kim¹, SooYoung Kim¹, Eun-Bong Kim¹, David Wang², Shinhaeng Kang¹, Yuhwan Ro³, Seungwoo Seo³, JoonHo Song³, Jaeyoun Youn¹, Kyomin Sohn¹, Nam Sung Kim¹

¹Samsung Electronics, Hwaseong, Korea

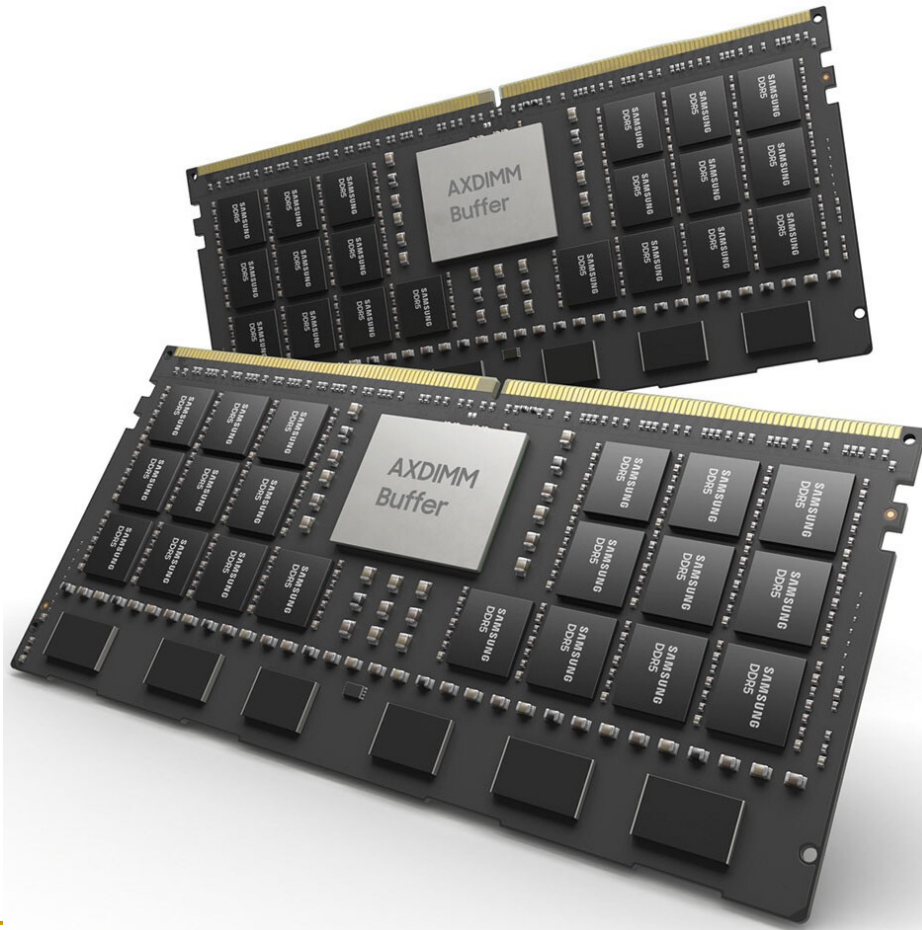
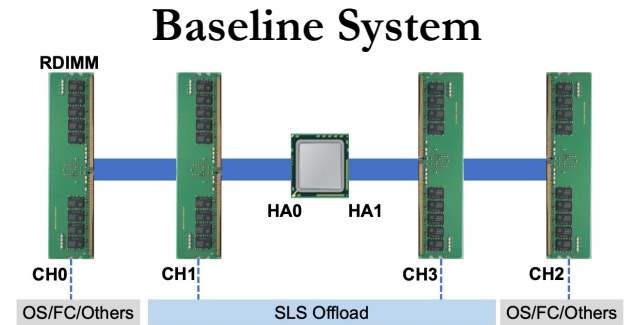
²Samsung Electronics, San Jose, CA

³Samsung Electronics, Suwon, Korea

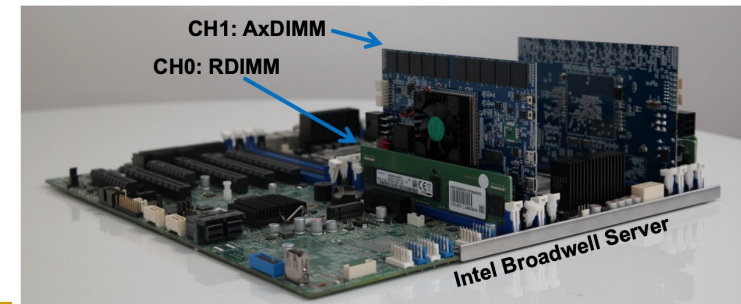
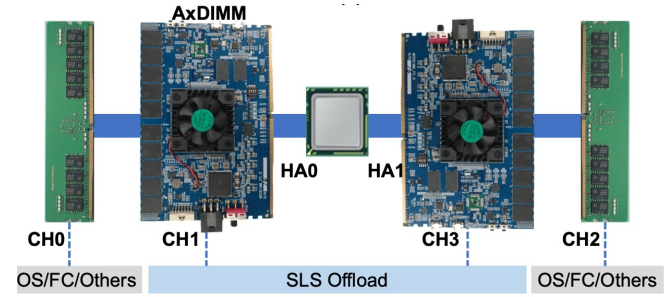
Samsung AxDIMM (2021)

- DDRx-PIM

Baseline system

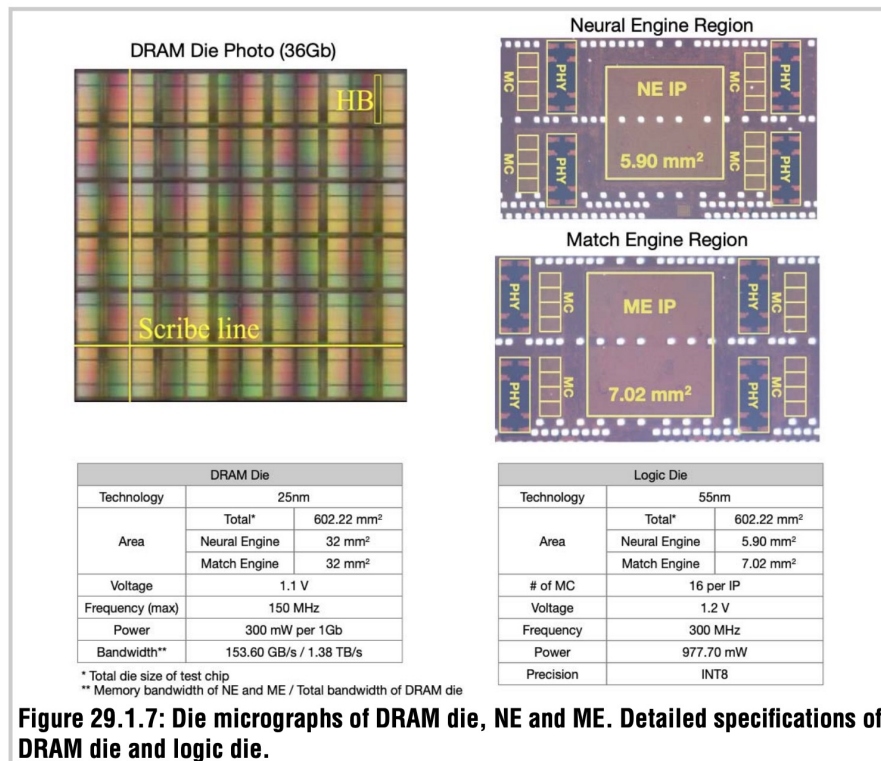
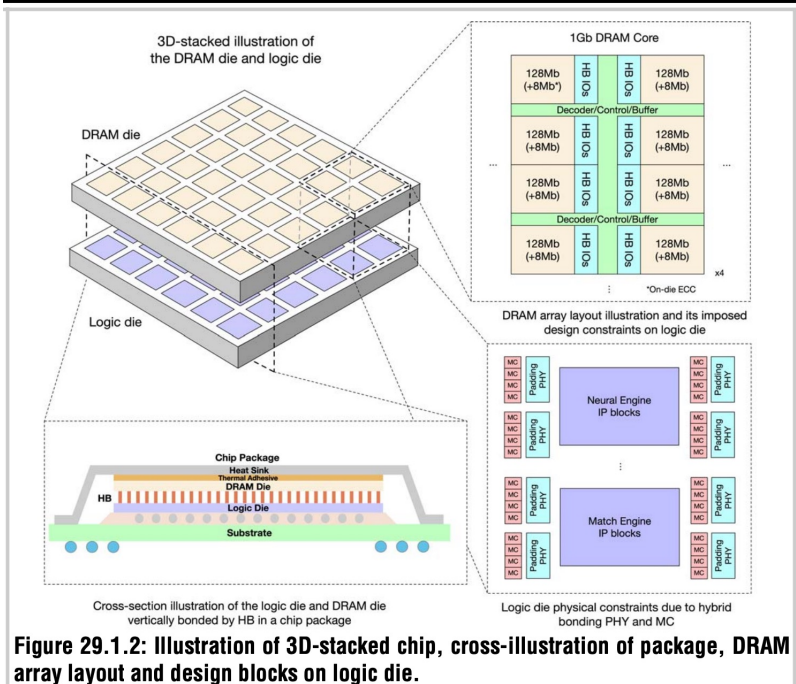


AxDIMM System



AliBaba PIM Recommendation System (2022)

ISSCC 2022 / February 24, 2022 / 8:30 AM



29.1 184QPS/W 64Mb/mm² 3D Logic-to-DRAM Hybrid Bonding with Process-Near-Memory Engine for Recommendation System

Dimin Niu¹, Shuangchen Li¹, Yuhao Wang¹, Wei Han¹, Zhe Zhang², Yijin Guan², Tianchan Guan³, Fei Sun¹, Fei Xue¹, Lide Duan¹, Yuanwei Fang¹, Hongzhong Zheng¹, Xiping Jiang⁴, Song Wang⁴, Fengguo Zuo⁴, Yubing Wang⁴, Bing Yu⁴, Qiwei Ren⁴, Yuan Xie¹

Recall: Takeaways

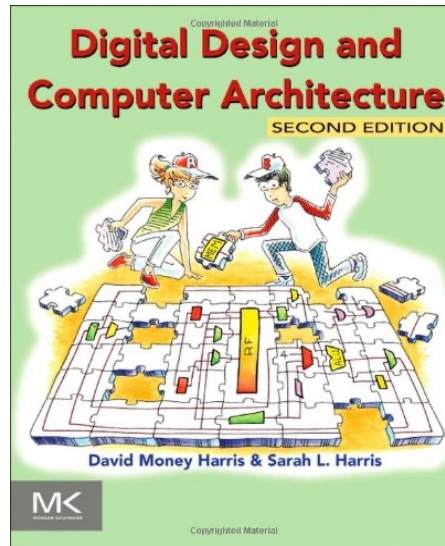
- It all starts from the basic building blocks and design principles
- And, knowledge of how to use, apply, enhance them
- Underlying technology might change (e.g., steel vs. wood)
 - but methods of taking advantage of technology bear resemblance
 - methods used for design depend on the principles employed

Basic Building Blocks

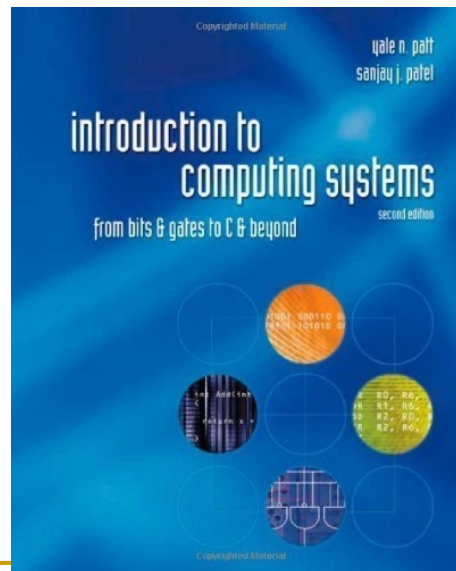
- Electrons
- Transistors
- Logic Gates
- Combinational Logic Circuits
- Sequential Logic Circuits
 - Storage Elements and Memory
- ...
- Cores
- Caches
- Interconnect
- Memories
- ...

Reading Assignments for This Week

- Chapter 1 in Harris & Harris



- Chapters 1-2 in Patt and Patel



- Supplementary Lecture Slides on Binary Numbers

Recall: 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
-

Recall: 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
 - ...

Course Info and Logistics

If You Need Help

- Post your question on Moodle Q&A Forum
 - <https://moodle-app2.let.ethz.ch/course/view.php?id=19395>
 - 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

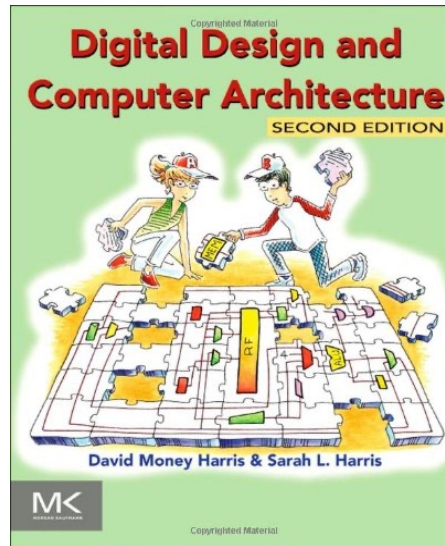
Where to Get Up-to-date Course Info?

- Website:
 - <https://pooyanjamshidi.github.io/csce212/>
 - Lecture slides and (videos)
 - Readings
 - 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

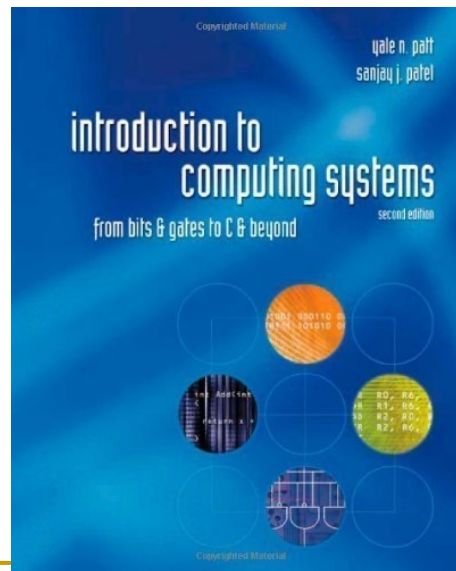
- TA

Reading Assignments for This Week

- Chapter 1 in Harris & Harris



- Chapters 1-2 in Patt and Patel (encouraged)



Reading Assignments for Next Week

- Combinational Logic chapters from both books
 - Harris and Harris, Chapter 2
 - Patt and Patel, Chapter 3
- Check the course website for all future readings
 - Required
 - Recommended
 - Mentioned

Future Lectures and Assignments

- You can also anticipate (and plan for) future lectures and assignments based on Spring 2023 schedule:
 - <https://pooyanjamshidi.github.io/csce212/lectures/>

Takeaways

- It is an exciting time to be understanding and designing computing architectures
- Many challenging and exciting problems in platform design
 - That no one has tackled (or thought about) before
 - That can have huge impact on the world's future
- Driven by huge hunger for data (Big Data), new applications (ML/AI, graph analytics, genomics), ever-greater realism, ...
 - We can easily collect more data than we can analyze/understand
- Driven by significant difficulties in keeping up with that hunger at the technology layer
 - Five walls: Energy, reliability, complexity, security, scalability

Major High-Level Goals of This Course

In 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)
 - **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, tradeoffs are important!

Why These Goals?

- Because you are here for a Computer Science degree
- **Regardless of your future direction**, learning the principles of computer architecture will be useful to
 - design better systems (software + hardware)
 - make better tradeoffs in design
 - understand why computers behave the way they do
 - solve problems better
 - think “in parallel”
 - think critically
 - ...

I presume you all know the number systems?

- Binary Number
- Hexadecimal Numbers
- Bits, Bytes, Words
- least significant bit (lsb), most significant bit (msb)
- Least Significant Byte (LSB), Most Significant Byte (MSB)
- KB, MB, GB, TB
- Binary Addition
- Signed Binary Numbers