

Computer Architecture

Lecture 1: Introduction and Basics

Prof. Onur Mutlu

ETH Zürich

Fall 2023

28 September 2023

Brief Self Introduction



■ Onur Mutlu

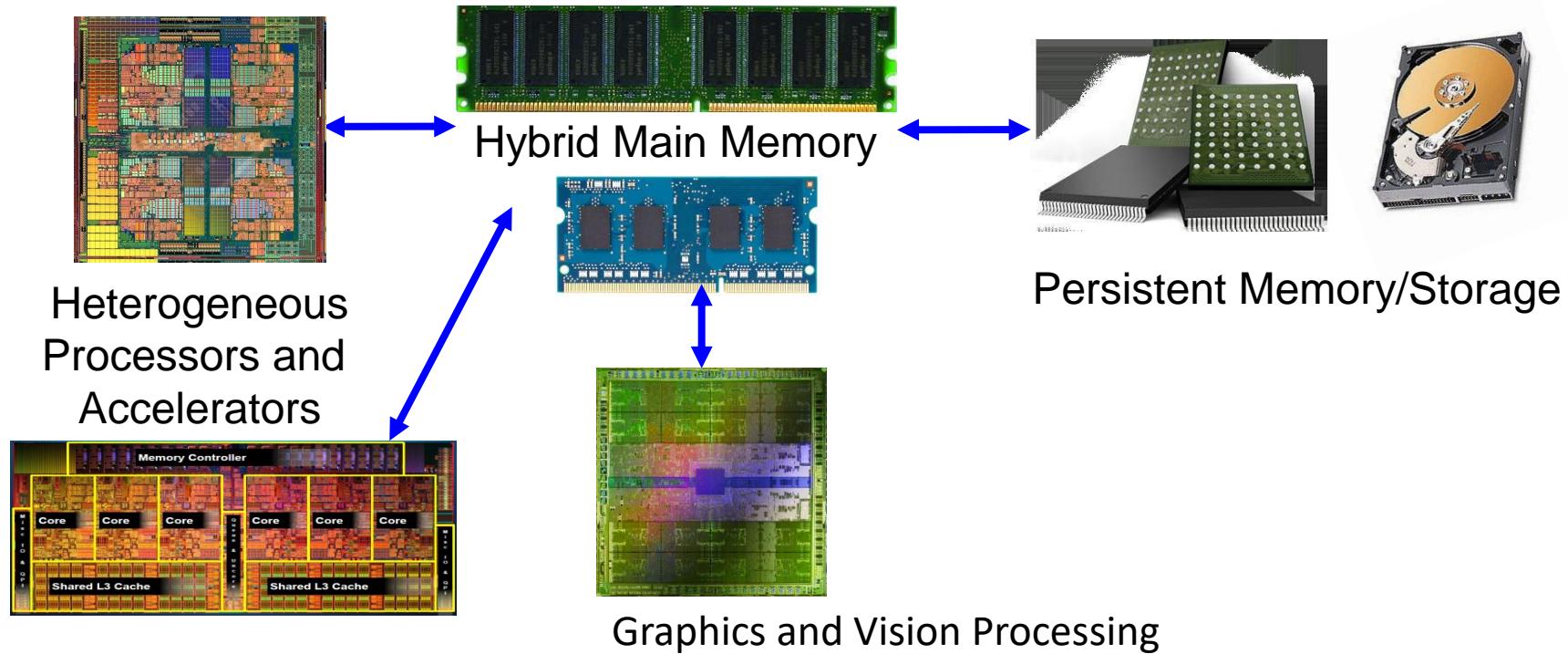
- Full Professor @ ETH Zurich ITET (INFK), since Sept 2015
- Strecker Professor @ Carnegie Mellon University ECE (CS), 2009-2016, 2016-...
- Started the Comp Arch Research Group @ Microsoft Research, 2006-2009
- Worked @ Google, VMware, Microsoft Research, Intel, AMD
- PhD in Computer Engineering from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

■ Research and Teaching in:

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

Current Research Mission

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



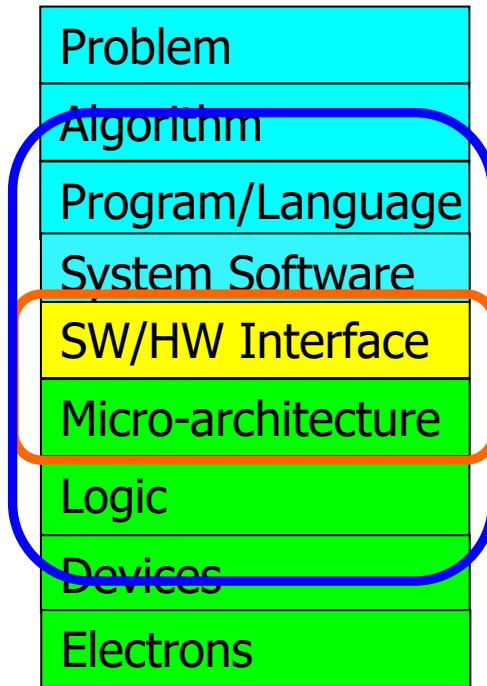
Build fundamentally better architectures

Four Key Current Directions

- Fundamentally Secure/Reliable/Safe Architectures
- Fundamentally Energy-Efficient Architectures
 - Memory-centric (Data-centric) Architectures
- Fundamentally Low-Latency and Predictable Architectures
- Architectures for AI/ML, Genomics, Medicine, Health

The Transformation Hierarchy

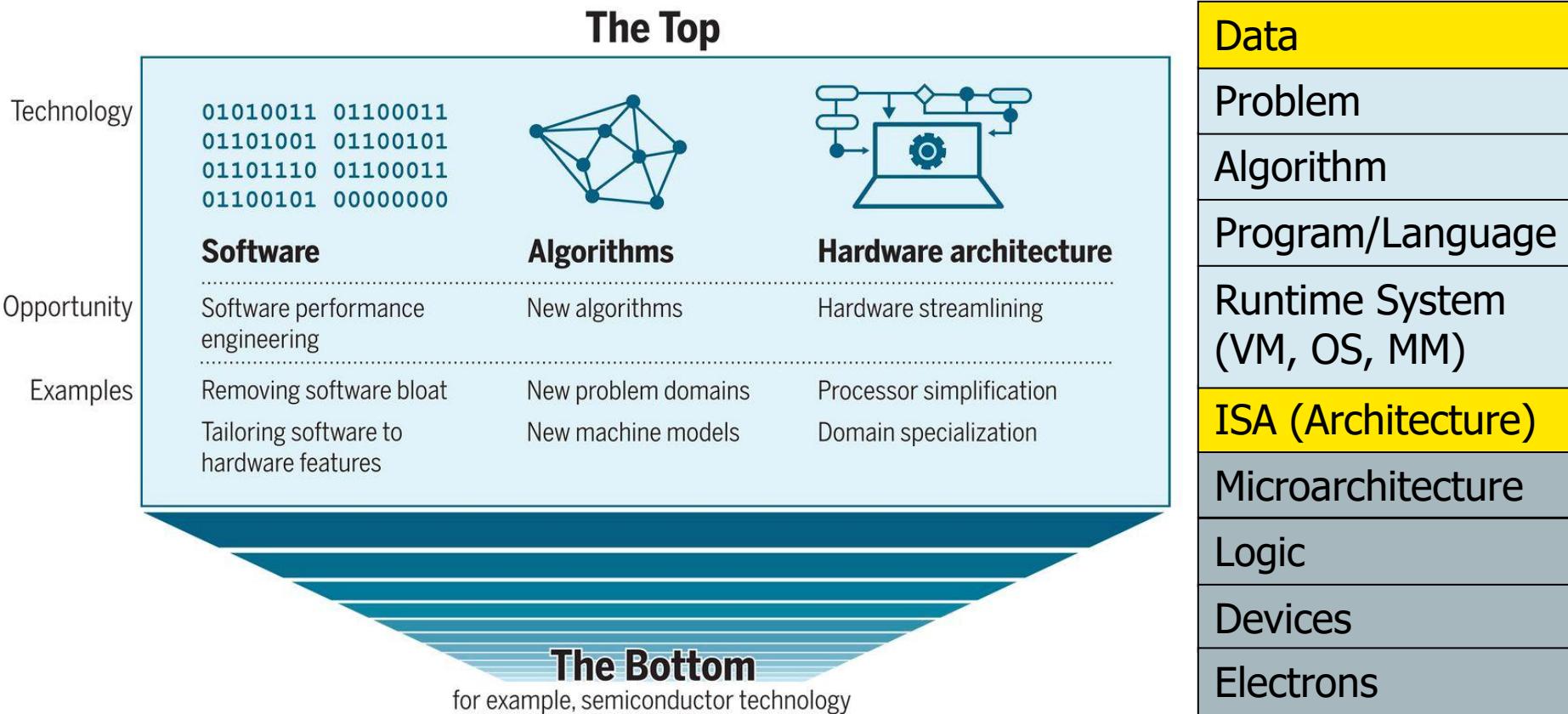
Computer Architecture
(expanded view)



Computer Architecture
(narrow view)

Computing System

Leiserson+, "[There's plenty of room at the Top: What will drive computer performance after Moore's law?](#)", Science, 2020



Richard Feynman, "[There's Plenty of Room at the Bottom: An Invitation to Enter a New Field of Physics](#)", a lecture given at Caltech, 1959.

Software & Hardware Optimizations

Multiplying Two 4096-by-4096 Matrices

```
for i in xrange(4096):  
    for j in xrange(4096):  
        for k in xrange(4096):  
            C[i][j] += A[i][k] *  
B[k][j]
```

The diagram shows the multiplication of two 3x3 matrices. The first matrix has columns 1, 2, 3 highlighted in yellow. The second matrix has rows 7, 8 highlighted in yellow. The result of the multiplication is shown in a third matrix, with the value 58 highlighted in yellow. Arrows point from the highlighted elements of the first matrix to the corresponding row of the second matrix, and from the second matrix to the result matrix.

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 7 & 8 \\ 9 & 10 \\ 11 & 12 \end{bmatrix} = \begin{bmatrix} 58 \end{bmatrix}$$

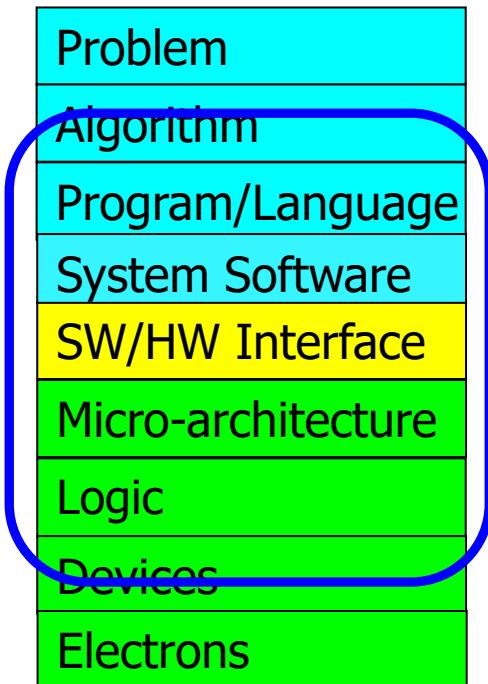
Implementation	Running time (s)	Absolute speedup
Python	25,552.48	1x
Java	2,372.68	11x
C	542.67	47x
Parallel loops	69.80	366x
Parallel divide and conquer	3.80	6,727x
plus vectorization	1.10	23,224x
plus AVX intrinsics	0.41	62,806x

Leiserson+, "[There's plenty of room at the Top: What will drive computer performance after Moore's law?](#)", Science, 2020

Axiom

To achieve the highest **energy efficiency** and **performance**:

we must take the expanded view
of computer architecture

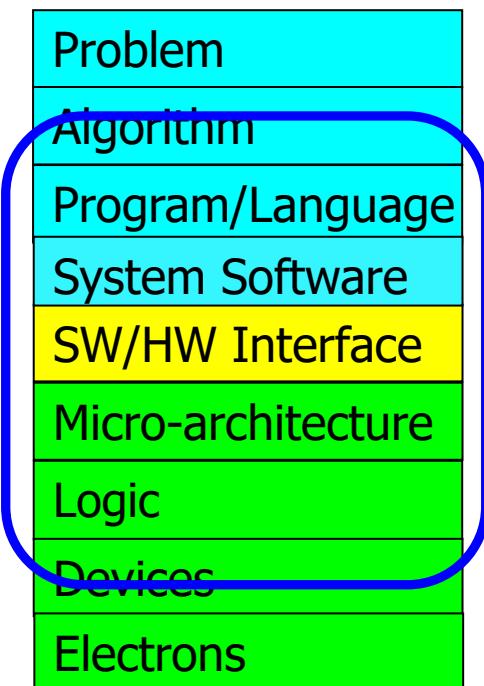


**Co-design across the hierarchy:
Algorithms to devices**

**Specialize as much as possible
within the design goals**

Current Research Mission & Major Topics

Build fundamentally better architectures



- Data-centric arch. for low energy & high perf.
 - Proc. in Mem/DRAM, NVM, unified mem/storage
- Low-latency & predictable architectures
 - Low-latency, low-energy yet low-cost memory
 - QoS-aware and predictable memory systems
- Fundamentally secure/reliable/safe arch.
 - Tolerating all bit flips; patchable HW; secure mem
- Architectures for ML/AI/Genomics/Graph/Med
 - Algorithm/arch./logic co-design; full heterogeneity
- Data-driven and data-aware architectures
 - ML/AI-driven architectural controllers and design
 - Expressive memory and expressive systems

**Broad research
spanning apps, systems, logic
with architecture at the center**

SAFARI Research Group



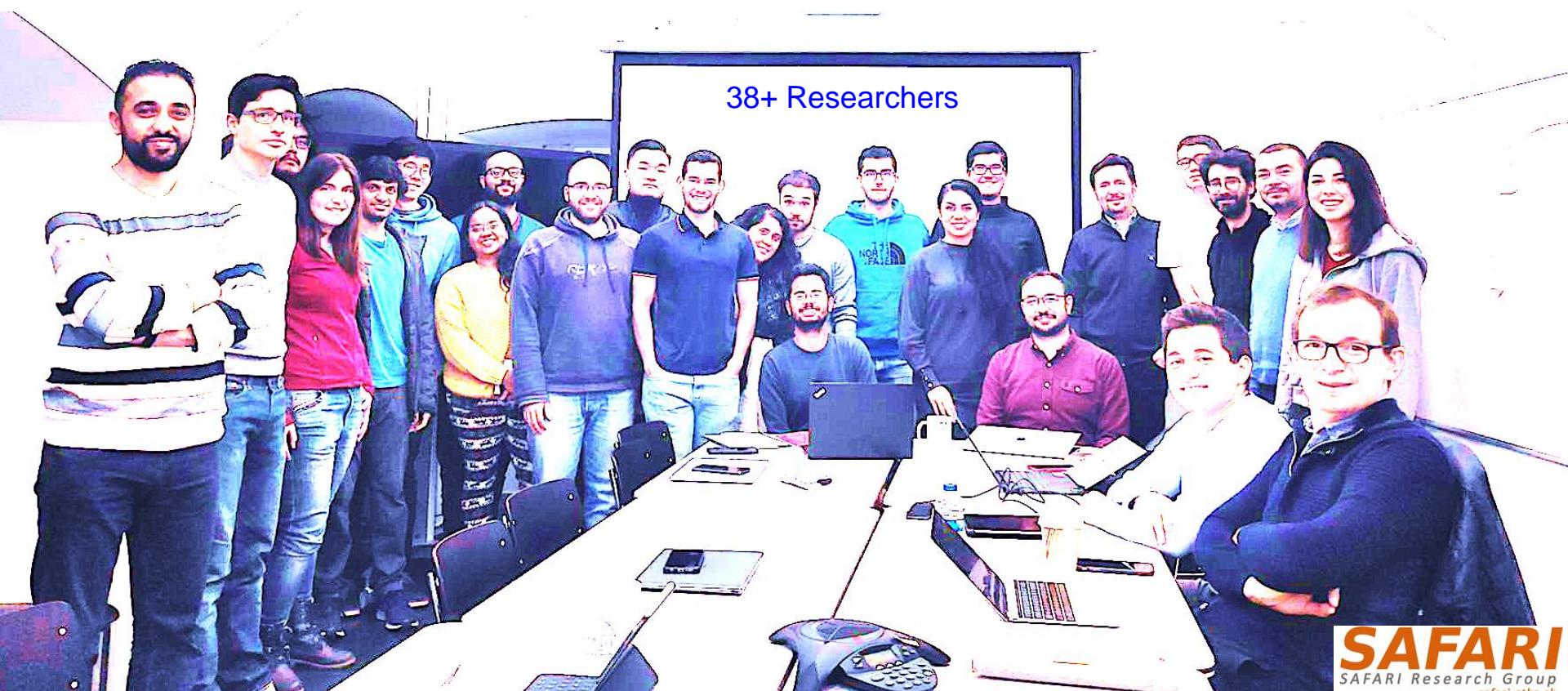
Think BIG, Aim HIGH!

<https://safari.ethz.ch>

Onur Mutlu's SAFARI Research Group

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

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



SAFARI
SAFARI Research Group
safari.ethz.ch

Think BIG, Aim HIGH!

SAFARI

<https://safari.ethz.ch>

SAFARI Newsletter January 2021 Edition

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



Newsletter
January 2021

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



Dear SAFARI friends,

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

SAFARI Newsletter December 2021 Edition

- <https://safari.ethz.ch/safari-newsletter-december-2021/>

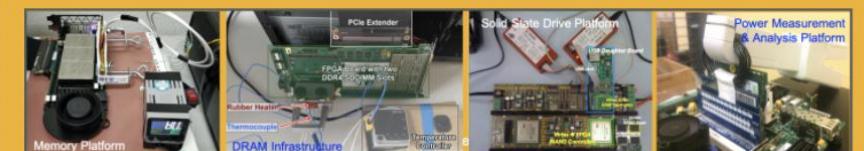
SAFARI
SAFARI Research Group

Think Big, Aim High

f t in y

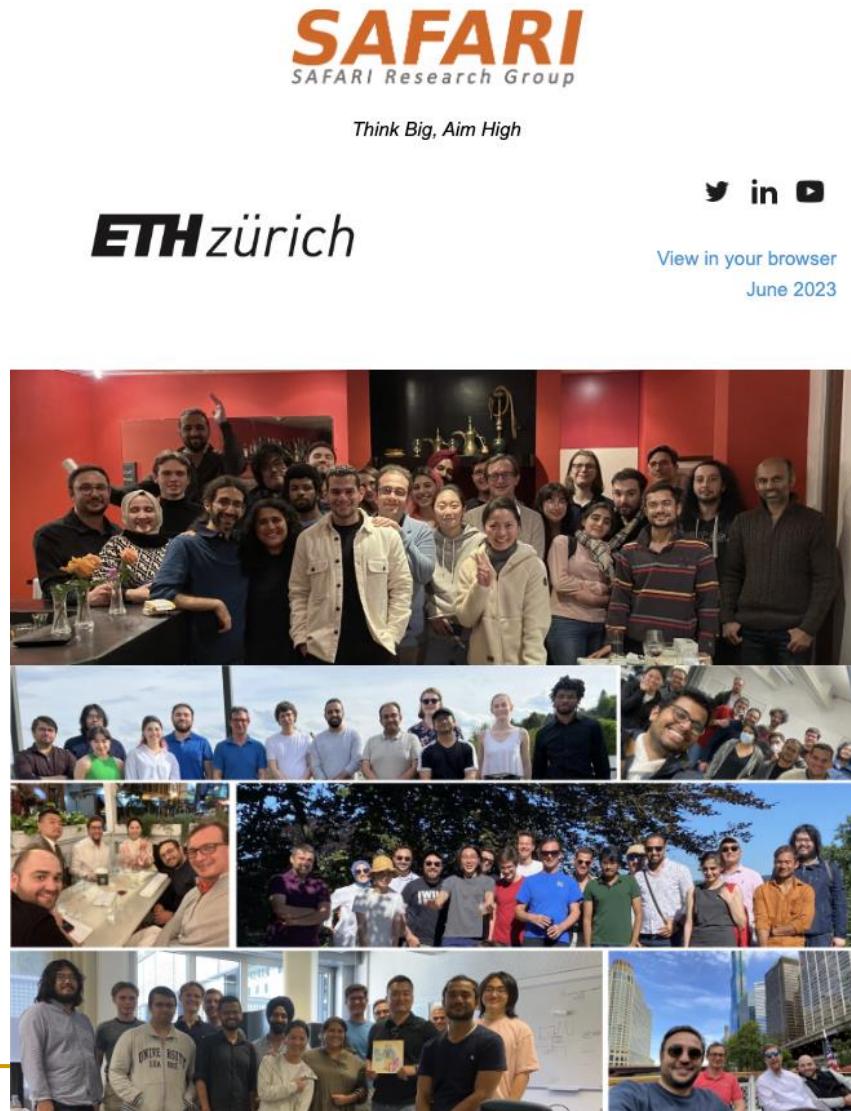
ETH zürich

View in your browser
December 2021



SAFARI Research Group: June 2023 Edition

- <https://safari.ethz.ch/safari-newsletter-june-2023/>



SAFARI PhD and Post-Doc Alumni

- <https://safari.ethz.ch/safari-alumni/>
 - Hasan Hassan (Rivos), **S&P 2020 Best Paper Award, 2020 Pwnie Award, IEEE Micro Top Picks HM 2020**
 - Christina Giannoula (Univ. of Toronto)
 - Minesh Patel (ETH Zurich), **MICRO 2020 and DSN 2020 Best Paper Awards; ISCA Hall of Fame 2021**
 - Damla Senol Cali (Bionano Genomics), **SRC TECHCON 2019 Best Student Presentation Award**
 - Nastaran Hajinazar (Intel)
 - Gagandeep Singh (AMD/Xilinx), **FPL 2020 Best Paper Award Finalist**
 - Amirali Boroumand (Stanford Univ → Google), **SRC TECHCON 2018 Best Presentation, RECOMB-Seq 2018 Best Poster**
 - Jeremie Kim (Apple), **EDAA Outstanding Dissertation Award 2020; IEEE Micro Top Picks 2019; ISCA/MICRO HoF 2021**
 - Nandita Vijaykumar (Univ. of Toronto, Assistant Professor), **ISCA Hall of Fame 2021**
 - Kevin Hsieh (Microsoft Research, Senior Researcher)
 - Justin Meza (Facebook), **HiPEAC 2015 Best Student Presentation Award; ICCD 2012 Best Paper Award**
 - Mohammed Alser (ETH Zurich), **IEEE Turkey Best PhD Thesis Award 2018**
 - Yixin Luo (Google), **HPCA 2015 Best Paper Session**
 - Kevin Chang (Facebook), **SRC TECHCON 2016 Best Student Presentation Award**
 - Rachata Ausavarungnirun (KMUNTB, Assistant Professor), **NOCS 2015 and NOCS 2012 Best Paper Award Finalist**
 - Gennady Pekhimenko (Univ. of Toronto, Assistant Professor), **ISCA Hall of Fame 2021; ASPLOS 2015 SRC Winner**
 - Vivek Seshadri (Microsoft Research)
 - Donghyuk Lee (NVIDIA Research, Senior Researcher), **HPCA Hall of Fame 2018**
 - Yoongu Kim (Software Robotics → Google), **TCAD'19 Top Pick Award; IEEE Micro Top Picks'10; HPCA'10 Best Paper Session**
 - Lavanya Subramanian (Intel Labs → Facebook)
 - Samira Khan (Univ. of Virginia, Assistant Professor), **HPCA 2014 Best Paper Session**
 - Saugata Ghose (Univ. of Illinois, Assistant Professor), **DFRWS-EU 2017 Best Paper Award**
 - Jawad Haj-Yahya (Huawei Research Zurich, Principal Researcher)
-

SAFARI Research Group: Introduction and Research

- Onur Mutlu,

"SAFARI Research Group: Introduction & Research"

*Talk at ETH Future Computing Laboratory Welcome Workshop (EFCL),
Virtual, 6 July 2021.*

[Slides (pptx) (pdf)]

The image shows a YouTube thumbnail for a video titled "SAFARI Research Group: Introduction & Research". The thumbnail features a white background with a gold border. Inside, the title is displayed in green and red text. Below the title, the speaker's name, email, and seminar details are listed. At the bottom, there are logos for SAFARI, ETH Zürich, Carnegie Mellon, and Zoom, along with standard YouTube controls like play, volume, and search. The video has 0:03 / 1:47:54 duration and Intro selected. The channel information at the bottom includes the speaker's profile picture, name, subscribers, and a subscribe button. The video has 719 views and was streamed 1 month ago.

SAFARI Research Group
Introduction & Research

Onur Mutlu
omutlu@gmail.com
<https://people.inf.ethz.ch/omutlu>
23 March 2023
Computer Architecture Seminar

SAFARI ETH zürich Carnegie Mellon zoom

Seminar in Computer Architecture - Lecture 5: Potpourri of Research Topics (Spring 2023)

Onur Mutlu Lectures 32.6K subscribers Subscribed

17 719 views Streamed 1 month ago Livestream - Seminar in Computer Architecture - ETH Zürich (Spring 2023)

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

A Talk on Impactful Research & Teaching

The screenshot shows a YouTube video player. At the top, there's a navigation bar with a play button, volume control, and a progress bar showing 0:27 / 50:31. The main title 'Applying to Grad School & Doing Impactful Research' is displayed in a large serif font within a gold-bordered box. Below the title, the speaker's name 'Onur Mutlu' and email 'omutlu@gmail.com' are shown, along with a direct link 'https://people.inf.ethz.ch/omutlu'. The date '13 June 2020' and the event 'Undergraduate Architecture Mentoring Workshop @ ISCA 2021' are also mentioned. The video is framed by a black border on the left and right sides. At the bottom, there are logos for 'SAFARI', 'ETH zürich', and 'Carnegie Mellon'. The YouTube interface includes a like button (74), a dislike button, share, save, and more options buttons. The channel information shows 'Onur Mutlu Lectures' with 17.2K subscribers. A description at the bottom states 'Panel talk at Undergraduate Architecture Mentoring Workshop at ISCA 2021 (<https://sites.google.com/wisc.edu/uar...>)'.

Principle: Teaching and Research

...

Teaching drives Research

Research drives Teaching

...

Popular uploads

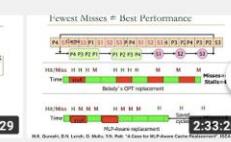
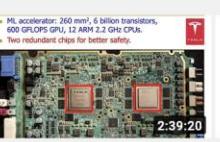
▶ PLAY ALL

[How Computers Work
\(from the ground up\)](#)

1:33:25

[Digital Design & Computer Architecture: Lecture 1:...](#)

49K views • 1 year ago

[Computer Architecture - Lecture 1: Introduction and Basics](#)

36K views • 3 years ago

[Computer Architecture - Lecture 1: Introduction and Basics](#)

31K views • 1 year ago

[Computer Architecture - Lecture 1: Introduction and Basics](#)

30K views • 8 months ago

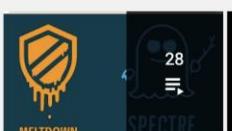
[Design of Digital Circuits - Lecture 1: Introduction and Basics](#)

22K views • 2 years ago

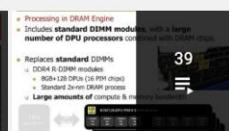
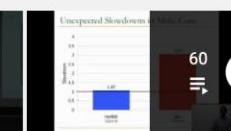
[Computer Architecture - Lecture 2: Fundamentals,...](#)

17K views • 3 years ago

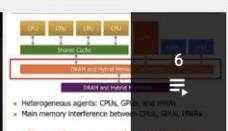
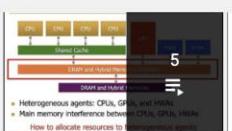
First Course in Computer Architecture & Digital Design 2021-2013

[Livestream - Digital Design and Computer Architecture - ETH...](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)[Digital Design & Computer Architecture - ETH Zürich...](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)[Design of Digital Circuits - ETH Zürich - Spring 2019](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)[Design of Digital Circuits - ETH Zürich - Spring 2018](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)[Digital Circuits and Computer Architecture - ETH Zurich - ...](#)Carnegie Mellon Computer Architec...
[VIEW FULL PLAYLIST](#)[Spring 2015 -- Computer Architecture Lectures ----](#)Carnegie Mellon Computer Architec...
[VIEW FULL PLAYLIST](#)

Advanced Computer Architecture Courses 2020-2012

[Computer Architecture - ETH Zürich - Fall 2020](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)[Computer Architecture - ETH Zürich - Fall 2019](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)[Computer Architecture - ETH Zürich - Fall 2018](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)[Computer Architecture - ETH Zürich - Fall 2017](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)[Fall 2015 - 740 Computer Architecture](#)Carnegie Mellon Computer Architec...
[VIEW FULL PLAYLIST](#)[Fall 2013 - 740 Computer Architecture - Carnegie Mellon](#)Carnegie Mellon Computer Architec...
[VIEW FULL PLAYLIST](#)

Special Courses on Memory Systems

[Computer Architecture Lecture 14b: Flash Memory and Solid-State Drives](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)[Champéry Winter School 2020 - Perugia NIPS Summer School Memory Systems and Memory... 2019](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)[SAMOS Tutorial 2019 - Memory Systems](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)[TU Wien 2019 - Memory Systems and Memory-Centric...](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)[ACACES 2018 Lectures -- Memory Systems and Memory-Centric...](#)Onur Mutlu Lectures
[VIEW FULL PLAYLIST](#)

Online Courses & Lectures

- **First Computer Architecture & Digital Design Course**
 - Digital Design and Computer Architecture
 - Spring 2023 Livestream Edition:
<https://www.youtube.com/playlist?list=PL5Q2soXY2Zi-EImKxYY1SZuGiAOBKaf>
- **Advanced Computer Architecture Course**
 - Computer Architecture
 - Fall 2022 Edition:
<https://www.youtube.com/playlist?list=PL5Q2soXY2Zi-cAls3cyauNzM7-74Eq31O>

Comp Arch (Fall 2021)

Fall 2021 Edition:

- ❑ <https://safari.ethz.ch/architecture/fall2021/doku.php?id=schedule>

Fall 2020 Edition:

- ❑ <https://safari.ethz.ch/architecture/fall2020/doku.php?id=schedule>

Youtube Livestream (2021):

- ❑ https://www.youtube.com/watch?v=4yfkM_5EFg_o&list=PL5Q2soXY2Zi-Mnk1PxjEIG32HAGILkTOF

Youtube Livestream (2020):

- ❑ <https://www.youtube.com/watch?v=c3mPdZA-Fmc&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN>

Master's level course

- ❑ Taken by Bachelor's/Masters/PhD students
- ❑ Cutting-edge research topics + fundamentals in Computer Architecture
- ❑ 5 Simulator-based Lab Assignments
- ❑ Potential research exploration
- ❑ Many research readings

<https://www.youtube.com/onurmutlulectures>



- Lectures/Schedule
- Lecture Buzzwords
- Readings
- HWs
- Labs
- Exams
- Related Courses
- Tutorials

- Computer Architecture FS20: Course Webpage
- Computer Architecture FS20: Lecture Videos
- Digitaltechnik SS21: Course Webpage
- Digitaltechnik SS21: Lecture Videos
- Moodle
- HotCRP
- Verilog Practice Website (HDLBits)

Lecture Video Playlist on YouTube

Watch on YouTube <https://arxiv.org/pdf/2105.03814.pdf>

Watch on YouTube <https://arxiv.org/pdf/2105.03814.pdf>

Fall 2021 Lectures & Schedule

Week	Date	Livestream	Lecture	Readings	Lab	HW
W1	30.09 Thu.	YouTube Live	L1: Introduction and Basics PDF PPT	Required Mentioned	Lab 1 Out	HW 0 Out
	01.10 Fri.	YouTube Live	L2: Trends, Tradeoffs and Design Fundamentals PDF PPT	Required Mentioned		
W2	07.10 Thu.	YouTube Live	L3a: Memory Systems: Challenges and Opportunities PDF PPT	Described Suggested		HW 1 Out
			L3b: Course Info & Logistics PDF PPT			
			L3c: Memory Performance Attacks PDF PPT	Described Suggested		
	08.10 Fri.	YouTube Live	L4a: Memory Performance Attacks PDF PPT	Described Suggested	Lab 2 Out	
			L4b: Data Retention and Memory Refresh PDF PPT	Described Suggested		
			L4c: RowHammer PDF PPT	Described Suggested		

Comp Arch (Fall 2022)

Fall 2022 Edition:

- <https://safari.ethz.ch/architecture/fall2022/doku.php?id=schedule>

Fall 2021 Edition:

- <https://safari.ethz.ch/architecture/fall2021/doku.php?id=schedule>

Youtube Livestream (2022):

- https://www.youtube.com/watch?v=4yfkM_5EFg_o&list=PL5Q2soXY2Zi-Mnk1PxjEIG32HAGILKTOF

Youtube Livestream (2021):

- https://www.youtube.com/watch?v=4yfkM_5EFg_o&list=PL5Q2soXY2Zi-Mnk1PxjEIG32HAGILKTOF

Master's level course

- Taken by Bachelor's/Masters/PhD students
- Cutting-edge research topics + fundamentals in Computer Architecture
- 5 Simulator-based Lab Assignments
- Potential research exploration
- Many research readings

<https://www.youtube.com/onurmutlulectures>


Computer Architecture - Fall 2022

Trace: • start • schedule

[Home](#)

[Announcements](#)

Materials

- [Lectures/Schedule](#)
- [Lecture Buzzwords](#)
- [Readings](#)
- [HWs](#)
- [Labs](#)
- [Exams](#)
- [Related Courses](#)
- [Tutorials](#)

Resources

- [Computer Architecture FS21: Course Webpage](#)
- [Computer Architecture FS21: Lecture Videos](#)
- [Digitalechnik SS21: Course Webpage](#)
- [Digitalechnik SS21: Lecture Videos](#)
- [Moodle](#)
- [HotCRP](#)
- [Verilog Practice Website \(HDLBits\)](#)

Lecture Video Playlist on YouTube

Livestream Lecture Playlist



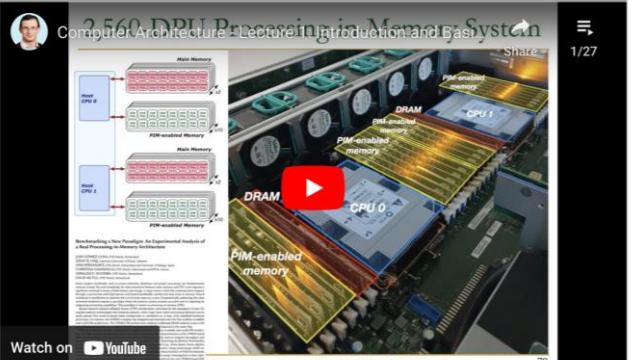
The largest ML accelerator chip (2021)
850,000 cores

Cerebras WSE-2
2.6 Trillion transistors
46,225 mm²

Largest GPU
54.2 Billion transistors
826 mm²
WSE Accelerator
<https://www.cerebras.com/show/14758/hot-chips-31-live-blog-cerebrus-wafer-scale-deep-learning/>

Lecture Playlist from Fall 2021

3.560 DPU Processing-in-Memory System



Watch on [YouTube](#) <https://arxiv.org/pdf/2105.03814.pdf>

Fall 2022 Lectures & Schedule

Week	Date	Livestream	Lecture	Readings	Lab	HW
W1	29.09 Thu.	YouTube Live	L1: Introduction and Basics (PDF) (PPT)	Required Mentioned	Lab 1 Out	HW 0 Out
	30.09 Fri.	YouTube Live	L2a: Memory Systems: Challenges and Opportunities (PDF) (PPT)	Described Suggested		
			L2b: Course Info & Logistics (PDF) (PPT)			
W2	06.10 Thu.	YouTube Live	L3: Processing using Memory (PDF) (PPT)	Described Suggested		HW 1 Out

DDCA (Spring 2022)

Spring 2022 Edition:

- ❑ <https://safari.ethz.ch/digitaltechnik/spring2022/oku.php?id=schedule>

Spring 2021 Edition:

- ❑ <https://safari.ethz.ch/digitaltechnik/spring2021/oku.php?id=schedule>

Youtube Livestream (Spring 2022):

- ❑ <https://www.youtube.com/watch?v=cpXdE3HwvK0&list=PL5Q2soXY2Zi97Ya5DEUpMpO2bbAoaG7c6>

Youtube Livestream (Spring 2021):

- ❑ https://www.youtube.com/watch?v=LbC0EZY8yw4&list=PL5Q2soXY2Zi_uej3aY39YB5pfW4SJ7LIN

Bachelor's course

- ❑ 2nd semester at ETH Zurich
- ❑ Rigorous introduction into "How Computers Work"
- ❑ Digital Design/Logic
- ❑ Computer Architecture
- ❑ 10 FPGA Lab Assignments

<https://www.youtube.com/onurmutlulectures>

Digital Design and Computer Architecture - Spring 2021

Trace: - schedule

Home Announcements Materials Resources

- Lectures/Schedule
- Lecture Buzzwords
- Readings
- Optional HWs
- Labs
- Extra Assignments
- Exams
- Technical Docs

- Computer Architecture (CMU) SS15: Lecture Videos
- Computer Architecture (CMU) SS15: Course Website
- Digitaltechnik SS18: Lecture Videos
- Digitaltechnik SS18: Course Website
- Digitaltechnik SS19: Lecture Videos
- Digitaltechnik SS19: Course Website
- Digitaltechnik SS20: Lecture Videos
- Digitaltechnik SS20: Course Website
- Moodle

Lecture Video Playlist on YouTube

Livestream Lecture Playlist

Onur Mutlu, Digital Design and Computer Architecture Today Watch later Share /28

Computing landscape is very different from 10-20 years ago

Applications and technology both demand novel architectures

Heterogeneous Processors and Accelerators

General Purpose GPUs

Persistent Memory/Storage

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

Watch on YouTube 66

Recorded Lecture Playlist

Digital Design & Computer Architecture: Lect... ANSWER Watch later Share /38

How Computers Work (from the ground up)

Watch on YouTube 6

Spring 2021 Lectures/Schedule

Week	Date	Livestream	Lecture	Readings	Lab	HW
W1	25.02 Thu.	YouTube Live	L1: Introduction and Basics PDF PPT	Required Suggested Mentioned		
	26.02 Fri.	YouTube Live	L2a: Tradeoffs, Metrics, Mindset PDF PPT	Required		
			L2b: Mysteries in Computer Architecture PDF PPT	Required Mentioned		
W2	04.03 Thu.	YouTube Live	L3a: Mysteries in Computer Architecture II PDF PPT	Required Suggested Mentioned		

Seminar in Comp Arch (Fall 2022)

Fall 2022 Edition:

- ❑ https://safari.ethz.ch/architecture_seminar/fall2022/doku.php?id=schedule

Fall 2021 Edition:

- ❑ https://safari.ethz.ch/architecture_seminar/fall2021/doku.php?id=schedule

Youtube Livestream (2022):

- ❑ https://www.youtube.com/watch?v=4TcP297mdsI&list=PL5Q2soXY2Zi_7UBNmC9B8Yr5JSwTG9yH4

Youtube Livestream (2021):

- ❑ https://www.youtube.com/playlist?list=PL5Q2soXY2Zi_JMBjHT5wHv-A2FSI54b2v

Critical analysis & presentation course

- ❑ Taken by Bachelor's/Masters/PhD students
- ❑ Cutting-edge research topics + talks + fundamentals in Computer Architecture
- ❑ 20+ research papers, presentations, analyses, discussions, brainstorming

Seminar in Computer Architecture - Fall 2022

Trace: start schedule

Home Materials Past Course Materials Resources Computer Architecture Digital Design and Computer Architecture

Lecture Video Playlist on YouTube

Lecture Playlist

Seminar in Computer Architecture, Lecture 1: Introduction and Basics

Watch on YouTube

Fall 2022 Lectures/Schedule

Week	Date	Livestream	Lecture	Readings	Assignments
W1	22.09 Thu.	YouTube Live	L1a: Course Logistics (PDF) (PPT)	Suggested	
			L1b: Introduction and Basics (PDF) (PPT)	Suggested	
W2	13.10 Thu.	YouTube Live	L2: Intelligent Genomic Analyses (PDF) (PPT)	Suggested	
W3	20.10 Thu.	YouTube Live	L3: Memory-Centric Computing (PDF) (PPT)	Suggested	
W4	27.10 Thu.	YouTube Live	L4: The Story of RowHammer (PDF) (PPT)	Suggested	
W5	03.11 Thu.	YouTube Live	S1.1: Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks, ASPLOS 2018 (PDF) (PPT)	Suggested	
			S1.2: Google Neural Network Models for Edge Devices: Analyzing and Mitigating Machine Learning Inference Bottlenecks, PACT 2021 (PDF) (PPT)	Suggested	
W6	10.11 Thu.	YouTube Live	S2.1: ComputeDRAM: In-Memory Compute Using Off-the-Shelf DRAMs, MICRO 2019 (PDF) (PPT)	Suggested	
			S2.2: A Case for Intelligent RAM, MICRO 1997 (PDF) (PPT)	Suggested	
W7	17.11 Thu.	YouTube Live	S2.3: Multicore Processors, ISCA 1995 (PDF) (PPT)	Suggested	

RowHammer & DRAM Exploration (Fall 2022)

Fall 2022 Edition:

- ❑ https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=softmc

Spring 2022 Edition:

- ❑ https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=softmc

Youtube Livestream (Spring 2022):

- ❑ https://www.youtube.com/watch?v=r5QxuoJWttg&list=PL5Q2soXY2Zi_1trfCckr6PTN8WR72icUO

Bachelor's course

- ❑ Elective at ETH Zurich
- ❑ Introduction to DRAM organization & operation
- ❑ Tutorial on using FPGA-based infrastructure
- ❑ Verilog & C++
- ❑ Potential research exploration

<https://www.youtube.com/onurmutlulectures>

Lecture Video Playlist on YouTube

[Lecture Playlist](#)

P&S SoftMC

Understanding and Improving Modern DRAM Performance, Reliability, and Security with Hands-On Experiments

Hasan Hassan
Prof. Onur Mutlu
ETH Zürich

Watch on [YouTube](#)

2022 Meetings/Schedule (Tentative)

Week	Date	Livestream	Meeting	Learning Materials	Assignments
W0	23.02 Wed.	YouTube Video	P&S SoftMC Tutorial	SoftMC Tutorial Slides (PDF) (PPT)	
W1	08.03 Tue.	YouTube Video	M1: Logistics & Intro to DRAM and SoftMC (PDF) (PPT)	Required Materials Recommended Materials	HW0
W2	15.03 Tue.	YouTube Video	M2: Revisiting RowHammer (PDF) (PPT)	(Paper PDF)	
W3	22.03 Tue.	YouTube Video	M3: Uncovering in-DRAM TRR & TRRespass (PDF) (PPT)		
W4	29.03 Tue.	YouTube Video	M4: Deeper Look Into RowHammer's Sensitivities (PDF) (PPT)		
W5	05.04 Tue.	YouTube Video	M5: QUAC-TRNG (PDF) (PPT)		
W6	12.04 Tue.	YouTube Video	M6: PiDRAM (PDF) (PPT)		

Exploration of Emerging Memory Systems (Fall 2022)

■ Fall 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=ramulator

■ Spring 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=ramulator

■ Youtube Livestream (Spring 2022):

- <https://www.youtube.com/watch?v=aMIIXRQd3s&list=PL5Q2soXY2ZiTlmLGwZ8hBo2925ZApqV>

■ Bachelor's course

- Elective at ETH Zurich
- Introduction to memory system simulation
- Tutorial on using Ramulator
- C++
- Potential research exploration

Lecture Video Playlist on YouTube

[Lecture Playlist](#)

Ramulator Course: Meeting 1: Logistics & Int... Watch Later Share 1/7

P&S Ramulator

Designing and Evaluating Memory Systems and Modern Software Workloads with Ramulator

Hasan Hassan
Prof. Onur Mutlu
ETH Zürich

Watch on YouTube

2022 Meetings/Schedule (Tentative)

Week	Date	Livestream	Meeting	Learning Materials	Assignments
W1	09.03 Wed.	Video	M1: Logistics & Intro to Simulating Memory Systems Using Ramulator PDF (PDF) (PPT)		HW0
W2	16.03 Fri.	Video	M2: Tutorial on Using Ramulator PDF (PDF) (PPT)		
W3	25.02 Fri.	Video	M3: BlockHammer PDF (PDF) (PPT)		
W4	01.04 Fri.	Video	M4: CLR-DRAM PDF (PDF) (PPT)		
W5	08.04 Fri.	Video	M5: SIMDRAIM PDF (PDF) (PPT)		
W6	29.04 Fri.	Video	M6: DAMOV PDF (PDF) (PPT)		
W7	06.05 Fri.	Video	M7: Syncron PDF (PDF) (PPT)		

<https://www.youtube.com/onurmutlulectures>

PIM Course (Fall 2022)

Fall 2022 Edition:

- ❑ https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=processing_in_memory

Spring 2022 Edition:

- ❑ https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=processing_in_memory

Youtube Livestream (Fall 2022):

- ❑ <https://www.youtube.com/watch?v=QLL0wQ9I4Dw&list=PL5Q2soXY2Zi8KzG2CQYRNQOVD0GOBrnKy>

Youtube Livestream (Spring 2022):

- ❑ <https://www.youtube.com/watch?v=9e4ChnwdoVo&list=PL5Q2soXY2Zi-841fUYYUK9EsXKhQKRPyX>

Project course

- ❑ Taken by Bachelor's/Master's students
- ❑ Processing-in-Memory lectures
- ❑ Hands-on research exploration
- ❑ Many research readings

<https://www.youtube.com/onurmutlulectures>

The screenshot shows a presentation slide titled "PIM Review and Open Problem" under the sub-section "Processing in Memory Course, Meeting 1 - Ex". The slide features a header with the authors' names: Onur Mutlu^{a,b}, Saugata Ghose^{b,c}, Juan Gómez-Luna^a, Rachata Ausavarungnirun^d. It includes logos for the SAFARI Research Group, Carnegie Mellon University, University of Illinois Urbana-Champaign, and King Mongkut's University of Technology North Bangkok. The main content discusses "A Modern Primer on Processing in Memory" and includes a link to the arXiv preprint: <https://arxiv.org/pdf/1903.03988.pdf>. A "Watch on YouTube" button is also present.

Spring 2022 Meetings/Schedule

Week	Date	Livestream	Meeting	Learning Materials	Assignments
W1	10.03 Thu.	Live	M1: P&S PIM Course Presentation (PDF) (PPT)	Required Materials Recommended Materials	HW 0 Out
W2	15.03 Tue. 17.03 Thu.		Hands-on Project Proposals M2: Real-world PIM: UPMEM PIM (PDF) (PPT)		
W3	24.03 Thu.	Live	M3: Real-world PIM: Microbenchmarking of UPMEM PIM (PDF) (PPT)		
W4	31.03 Thu.	Live	M4: Real-world PIM: Samsung HBM-PIM (PDF) (PPT)		
W5	07.04 Thu.	Live	M5: How to Evaluate Data Movement Bottlenecks (PDF) (PPT)		
W6	14.04 Thu.	Live	M6: Real-world PIM: SK Hynix AIM (PDF) (PPT)		
W7	21.04 Thu.	Premiere	M7: Programming PIM Architectures (PDF) (PPT)		
W8	28.04 Thu.	Premiere	M8: Benchmarking and Workload Suitability on PIM (PDF) (PPT)		
W9	05.05 Thu.	Premiere	M9: Real-world PIM: Samsung AxDIMM (PDF) (PPT)		
W10	12.05 Thu.	Premiere	M10: Real-world PIM: Alibaba HB-PNM (PDF) (PPT)		
W11	19.05 Thu.	Live	M11: SpMV on a Real PIM Architecture (PDF) (PPT)		
W12	26.05 Thu.	Live	M12: End-to-End Framework for Processing-using-Memory (PDF) (PPT)		
W13	02.06 Thu.	Live	M13: Bit-Serial SIMD Processing using DRAM (PDF) (PPT)		
W14	09.06 Thu.	Live	M14: Analyzing and Mitigating ML Inference Bottlenecks (PDF) (PPT)		
W15	15.06 Thu.	Live	M15: In-Memory HTAP Databases with HW/SW Co-design (PDF) (PPT)		
W16	23.06 Thu.	Live	M16: In-Storage Processing for Genome Analysis (PDF) (PPT)		
W17	18.07 Mon.	Premiere	M17: How to Enable the Adoption of PIM? (PDF) (PPT)		
W18	09.08 Tue.	Premiere	SS1: ISVLSI 2022 Special Session on PIM (PDF & PPT)		

Genomics Course (Fall 2022)

Fall 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/fall2022/do_doku.php?id=bioinformatics

Spring 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/spring2022/do_doku.php?id=bioinformatics

Youtube Livestream (Fall 2022):

- https://www.youtube.com/watch?v=nA41964-9r8&list=PL5Q2soXY2Zi8tFIQvdxOdizD_EhVAMVQV

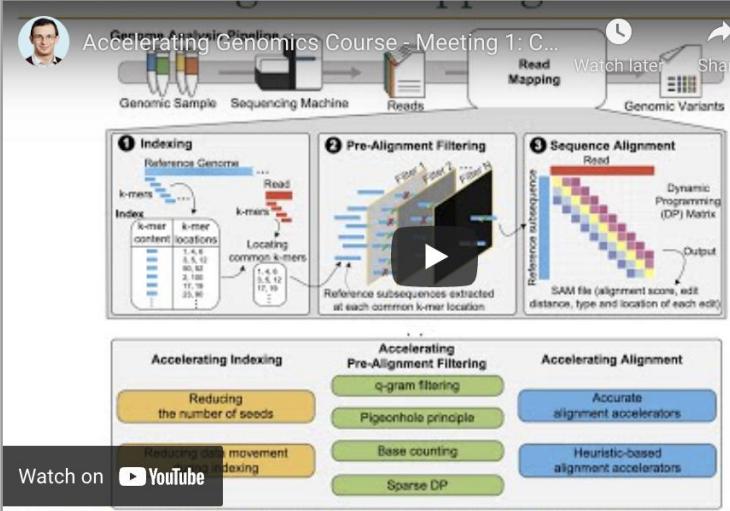
Youtube Livestream (Spring 2022):

- https://www.youtube.com/watch?v=DEL_5A_Y3TI&list=PL5Q2soXY2Zi8NrPDgOR1yRU_Cxxjw-u18

Project course

- Taken by Bachelor's/Master's students
- Genomics lectures
- Hands-on research exploration
- Many research readings

<https://www.youtube.com/onurmutlulectures>



Spring 2022 Meetings/Schedule

Week	Date	Livestream	Meeting	Learning Materials
W1	11.3 Fri.	YouTube Live	M1: P&S Accelerating Genomics Course Introduction & Project Proposals PDF PPT	Required Materials Recommended Materials
W2	18.3 Fri.	YouTube Live	M2: Introduction to Sequencing PDF PPT	
W3	25.3 Fri.	YouTube Premiere	M3: Read Mapping PDF PPT	
W4	01.04 Fri.	YouTube Premiere	M4: GateKeeper PDF PPT	
W5	08.04 Fri.	YouTube Premiere	M5: MAGNET & Shouji PDF PPT	
W6	15.4 Fri.	YouTube Premiere	M6: SneakySnake PDF PPT	
W7	29.4 Fri.	YouTube Premiere	M7: GenStore PDF PPT	
W8	06.05 Fri.	YouTube Premiere	M8: GRIM-Filter PDF PPT	
W9	13.05 Fri.	YouTube Premiere	M9: Genome Assembly PDF PPT	
W10	20.05 Fri.	YouTube Live	M10: Genomic Data Sharing Under Differential Privacy PDF PPT	
W11	10.06 Fri.	YouTube Premiere	M11: Accelerating Genome Sequence Analysis PDF PPT	

Hetero. Systems (Spring'22)

■ Spring 2022 Edition:

- ❑ https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=heterogeneous_systems

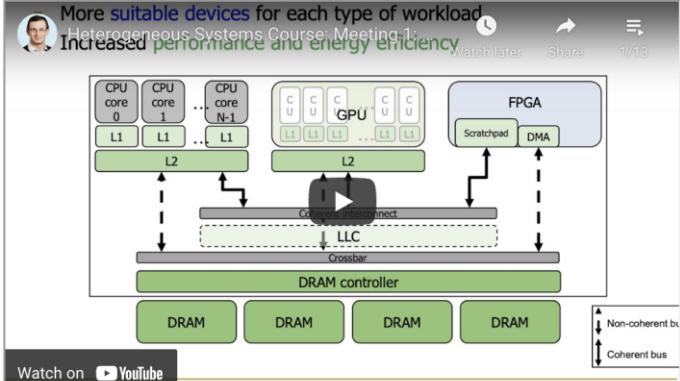
■ Youtube Livestream:

- ❑ <https://www.youtube.com/watch?v=oFO5fTrgFIY&list=PL5Q2soXY2Zi9XrgXR38IMFTjmY6h7Gzm>

■ Project course

- ❑ Taken by Bachelor's/Master's students
- ❑ GPU and Parallelism lectures
- ❑ Hands-on research exploration
- ❑ Many research readings

<https://www.youtube.com/onurmutlulectures>



Watch on [YouTube](#)
Balogh+, "Collaborative Computing for Heterogeneous Integrated Systems," ICPE 2017.

52

Spring 2022 Meetings/Schedule

Week	Date	Livestream	Meeting	Learning Materials	Assignments
W1	15.03 Tue.	YouTube Premiere	M1: P&S Course Presentation (PDF) (PPT)	Required Materials Recommended Materials	HW 0 Out
W2	22.03 Tue.	YouTube Premiere	M2: SIMD Processing and GPUs (PDF) (PPT)		
W3	29.03 Tue.	YouTube Premiere	M3: GPU Software Hierarchy (PDF) (PPT)		
W4	05.04 Tue.	YouTube Premiere	M4: GPU Memory Hierarchy (PDF) (PPT)		
W5	12.04 Tue.	YouTube Premiere	M5: GPU Performance Considerations (PDF) (PPT)		
W6	19.04 Tue.	YouTube Premiere	M6: Parallel Patterns: Reduction (PDF) (PPT)		
W7	26.04 Tue.	YouTube Premiere	M7: Parallel Patterns: Histogram (PDF) (PPT)		
W8	03.05 Tue.	YouTube Premiere	M8: Parallel Patterns: Convolution (PDF) (PPT)		
W9	10.05 Tue.	YouTube Premiere	M9: Parallel Patterns: Prefix Sum (Scan) (PDF) (PPT)		
W10	17.05 Tue.	YouTube Premiere	M10: Parallel Patterns: Sparse Matrices (PDF) (PPT)		
W11	24.05 Tue.	YouTube Premiere	M11: Parallel Patterns: Graph Search (PDF) (PPT)		
W12	01.06 Wed.	YouTube Premiere	M12: Parallel Patterns: Merge Sort (PDF) (PPT)		
W13	07.06 Tue.	YouTube Premiere	M13: Dynamic Parallelism (PDF) (PPT)		
W14	15.06 Wed.	YouTube Premiere	M14: Collaborative Computing (PDF) (PPT)		
W15	24.06 Fri.	YouTube Premiere	M15: GPU Acceleration of Genome Sequence Alignment (PDF) (PPT)		
W16	14.07 Thu.	YouTube Premiere	M16: Accelerating Agent-based Simulations (PDF) (ODP)		

HW/SW Co-Design (Spring 2022)

■ Spring 2022 Edition:

- ❑ https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=hw_sw_co_design

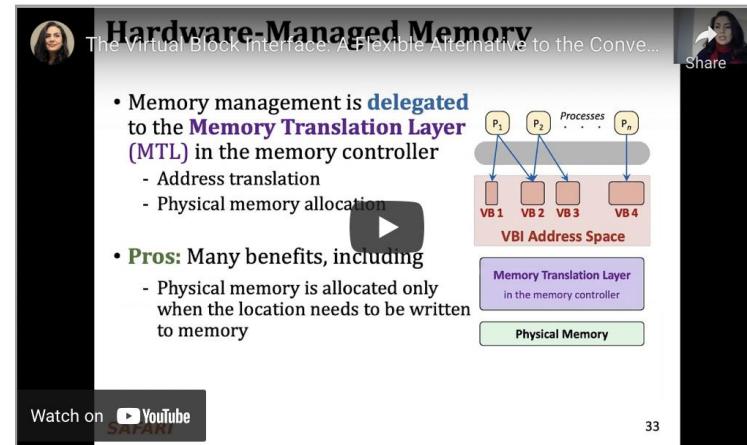
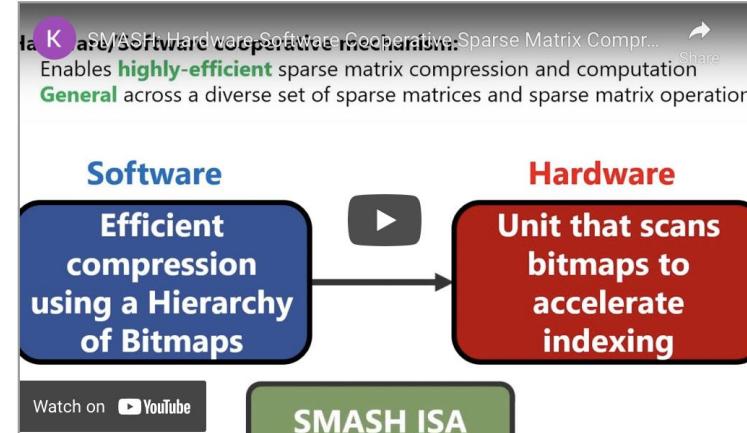
■ Youtube Livestream:

- ❑ <https://youtube.com/playlist?list=PL5Q2soXY2Zi8nH7un3ghD2nutKWWdk-NK>

■ Project course

- ❑ Taken by Bachelor's/Master's students
- ❑ HW/SW co-design lectures
- ❑ Hands-on research exploration
- ❑ Many research readings

<https://www.youtube.com/onurmutlulectures>



2022 Meetings/Schedule (Tentative)

Week	Date	Livestream	Meeting	Materials	Assignments
W0	16.03	YouTube Live	Intro to HW/SW Co-Design (PPTX) (PDF)	Required	HW 0 Out
W1	23.03		Project selection	Required	
W2	30.03	YouTube Live	Virtual Memory (I) (PPTX) (PDF)		
W3	13.04	YouTube Live	Virtual Memory (II) (PPTX) (PDF)		

SSD Course (Spring 2023)

■ Spring 2023 Edition:

- https://safari.ethz.ch/projects_and_seminars/spring2023/doku.php?id=modern_ssds

■ Fall 2022 Edition:

- https://safari.ethz.ch/projects_and_seminars/fall2022/do_ku.php?id=modern_ssds

■ Youtube Livestream (Spring 2023):

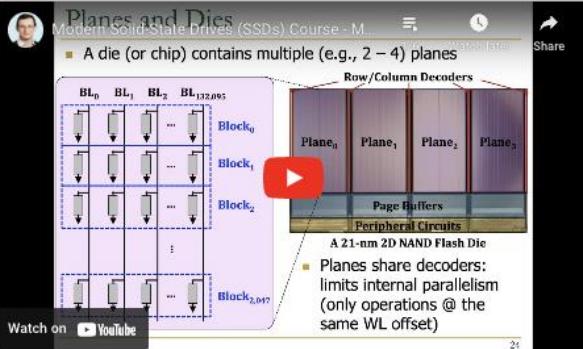
- https://www.youtube.com/watch?v=4VTwOMmsnJY&list=PL5Q2soXY2Zi_8qOM5Icpp8hB2SHtm4z57&pp=iAQB

■ Youtube Livestream (Fall 2022):

- <https://www.youtube.com/watch?v=hqLrd-Uj0aU&list=PL5Q2soXY2Zi9BJhenUq4JI5bwhAMpAp13&p=p=iAQB>

■ Project course

- Taken by Bachelor's/Master's students
- SSD Basics and Advanced Topics
- Hands-on research exploration
- Many research readings



Fall 2022 Meetings/Schedule

Week	Date	Livestream	Meeting	Learning Materials	Assignments
W1	06.10		M1: P&S Course Presentation PDF PPT	Required Recommended	
W2	12.10	YouTube Live	M2: Basics of NAND Flash-Based SSDs PDF PPT	Required Recommended	
W3	19.10	YouTube Live	M3: NAND Flash Read/Write Operations PDF PPT	Required Recommended	
W4	26.10	YouTube Live	M4: Processing inside NAND Flash PDF PPT	Required Recommended	
W5	02.11	YouTube Live	M5: Advanced NAND Flash Commands & Mapping PDF PPT	Required Recommended	
W6	09.11	YouTube Live	M6: Processing inside Storage PDF PPT	Required Recommended	
W7	23.11	YouTube Live	M7: Address Mapping & Garbage Collection PDF PPT	Required Recommended	
W8	30.11	YouTube Live	M8: Introduction to MQSim PDF PPT	Required Recommended	
W9	14.12	YouTube Live	M9: Fine-Grained Mapping and Multi-Plane Operation-Aware Block Management PDF PPT	Required Recommended	
W10	04.01.2023	YouTube Premiere	M10a: NAND Flash Basics PDF PPT	Required Recommended	
			M10b: Reducing Solid-State Drive Read Latency by Optimizing Read-Retry PDF PPT Paper	Required Recommended	
			M10c: Evanescos: Architectural Support for Efficient Data Sanitization in Modern Flash-Based Storage Systems PDF PPT Paper	Required Recommended	
			M10d: DeepSketch: A New Machine Learning-Based Reference Search Technique for Post-Deduplication Delta Compression PDF PPT Paper	Required Recommended	
W11	11.01	YouTube Live	M11: FLIN: Enabling Fairness and Enhancing Performance in Modern NVMe Solid State Drives PDF PPT	Required	
W12	25.01	YouTube Premiere	M12: Flash Memory and Solid-State Drives PDF PPT	Recommended	

Hands-On Projects & Seminars Courses

- https://safari.ethz.ch/projects_and_seminars/doku.php



SAFARI Project & Seminars Courses (Spring 2022)

Log In

Search

Recent Changes Media Manager Sitemap

Trace: • [heterogeneous_systems](#) • [ramulator](#) • [bioinformatics](#) • [processing_in_memory](#) • [start](#)

start

Home

Courses

- SoftMC
- Ramulator
- Accelerating Genomics
- Mobile Genomics
- Processing-in-Memory
- Heterogeneous Systems
- Modern SSDs
- Hardware/Software Co-design

SAFARI Projects & Seminars Courses (Spring 2022)

Welcome to the wiki for Project and Seminar courses SAFARI offers.

Courses we offer:

- Understanding and Improving Modern DRAM Performance, Reliability, and Security with Hands-On Experiments
- Designing and Evaluating Memory Systems and Modern Software Workloads with Ramulator
- Accelerating Genome Analysis with FPGAs, GPUs, and New Execution Paradigms
- Genome Sequencing on Mobile Devices
- Exploring the Processing-in-Memory Paradigm for Future Computing Systems
- Hands-on Acceleration on Heterogeneous Computing Systems
- Understanding and Designing Modern NAND Flash-Based Solid-State Drives (SSDs)
- Intelligent Architectures using Hardware/Software Cooperative Techniques

Special Research Sessions & Courses

- Special Session at ISVLSI 2022: 9 cutting-edge talks

The image shows a screenshot of a YouTube video player. The main content is a presentation slide with the following text:

In-Memory Processing
ISVLSI 2022 Special Session

IEEE Computer Society Annual Symposium on VLSI

ISVLSI 2022 logo

Adonis room
Ailathon resort, Paphos, Cyprus
July 4th, 2022

Video controls at the bottom include play/pause, volume, and progress bar showing 0:04 / 3:36:35. The video title is "Dr. Juan Gómez-Luna, "Introduction to the ISVLSI 2022 Special Session on Processing-in-Memory" >".

Below the video player, the YouTube channel information is shown:

ISVLSI 2022 Special Session on Processing-in-Memory

1,286 views • Premiered Aug 9, 2022

Onur Mutlu Lectures
26.9K subscribers

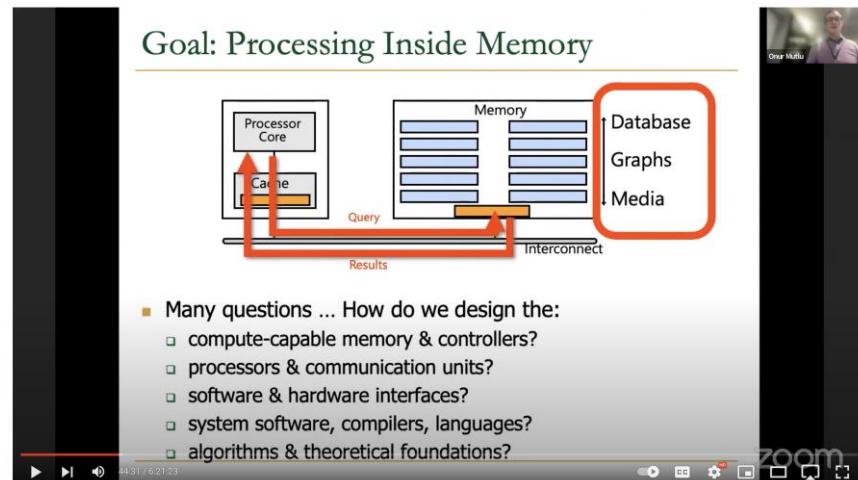
Interaction buttons: like (61), dislike, share, download, clip, save, more.

Analytics and edit video buttons are also visible.

Real PIM Tutorial (HPCA 2023)

■ February 26th: Lectures + Hands-on labs + Invited

The screenshot shows the homepage of the HPCA 2023 Real-World PIM Tutorial. It features a header with the title and a search bar. Below the header is a navigation menu with links to 'Recent Changes', 'Media Manager', and 'Sitemap'. A sidebar on the left contains a logo and a link to 'Trace: start'. The main content area has a section titled 'Real-world Processing-in-Memory Architectures' with a 'Tutorial Description' and a detailed 'Table of Contents' listing various sections like 'Real-world Processing-in-Memory Architectures', 'Tutorial Description', 'Organizers', 'Agenda (February 26, 2023)', 'Lectures (tentative)', 'Hands-on Labs (tentative)', and 'Learning Materials'. There are also sections on '2,560-DPU Processing-in-Memory System' with a diagram and a link to a paper, and a section on 'PIM can provide large improvements in both performance and energy consumption'. The footer includes a link to the arXiv paper.



The image shows a YouTube video player for the 'HPCA 2023 Tutorial: Real-World Processing-in-Memory Architectures' by Onur Mutlu Lectures. The video has 1.8K views and was streamed 1 month ago. The video player interface includes a play button, progress bar, and various sharing options.

Time	Speaker	Title	Materials
8:00am-8:40am	Prof. Onur Mutlu	Memory-Centric Computing	PDF PPT
8:40am-10:00am	Dr. Juan Gómez Luna	Processing-Near-Memory: Real PNM Architectures Programming General-purpose PIM	PDF PPT
10:20am-11:00am	Dr. Dimin Niu	A 3D Logic-to-DRAM Hybrid Bonding Process-Near-Memory Chip for Recommendation System	
11:00am-11:40am	Dr. Christina Giannoula	SparseP: Towards Efficient Sparse Matrix Vector Multiplication on Real Processing-In-Memory Architectures	PDF PPT
1:30pm-2:10pm	Dr. Juan Gómez Luna	Processing-Using-Memory: Exploiting the Analog Operational Properties of Memory Components	PDF PPT
2:10pm-2:50pm	Dr. Manuel Le Gallo	Deep Learning Inference Using Computational Phase-Change Memory	
2:50pm-3:30pm	Dr. Juan Gómez Luna	PIM Adoption Issues: How to Enable PIM Adoption?	PDF PPT
3:40pm-5:40pm	Dr. Juan Gómez Luna	Hands-on Lab: Programming and Understanding a Real Processing-in-Memory Architecture	Handout PDF PPT

<https://www.youtube.com/live/f5-nT1tbz5w?feature=share>

<https://events.safari.ethz.ch/real-pim-tutorial/doku.php?id=start>

Real PIM Tutorial (ASPLOS 2023)

■ March 26th: Lectures + Hands-on labs + Invited lectures

The screenshot shows the ASPLOS 2023 Real-World PIM Tutorial website. At the top, there's a logo of a pencil writing on a notepad with the word "EDW". Below it, the title "ASPLOS 2023 Real-World PIM Tutorial" is displayed. A search bar and navigation links for "Recent Changes", "Media Manager", and "Sitemap" are visible. The main content area has a header "Real-world Processing-in-Memory Systems for Modern Workloads". It includes a "Tutorial Description" section with text about PIM, a "Table of Contents" sidebar with items like "Real-world Processing-in-Memory Systems for Modern Workloads", "Tutorial Description", "Organizers", "Agenda (March 26, 2023)", "Lectures (tentative)", "Hands-on Labs (tentative)", "Learning Materials", and "Registration", and a "2,560-DPU Processing-in-Memory System" section with a diagram.

Tutorial Materials

Time	Speaker	Title	Materials
9:00am-10:20am	Prof. Onur Mutlu	Memory-Centric Computing	PDF PPT
10:40am-12:00pm	Dr. Juan Gómez Luna	Processing-Near-Memory: Real PNM Architectures Programming General-purpose PIM	PDF PPT
1:40pm-2:20pm	Prof. Alexandra (Sasha) Fedorova (UBC)	Processing in Memory in the Wild	PDF PPT
2:20pm-3:20pm	Dr. Juan Gómez Luna & Ataberk Olgun	Processing-Using-Memory: Exploiting the Analog Operational Properties of Memory Components	PDF PPT PDF PPT
3:40pm-4:10pm	Dr. Juan Gómez Luna	Adoption issues: How to enable PIM? Accelerating Modern Workloads on a General-purpose PIM System	PDF PPT PDF PPT
4:10pm-4:50pm	Dr. Yongkee Kwon & Eddy (Chanwook) Park (SK Hynix)	System Architecture and Software Stack for GDDR6-AiM	PDF PPT
4:50pm-5:00pm	Dr. Juan Gómez Luna	Hands-on Lab: Programming and Understanding a Real Processing-in-Memory Architecture	Handout PDF PPT

The screenshot shows a video player interface for the "Real-world Processing-in-Memory Systems for Modern Workloads" tutorial. The title "Real-world Processing-in-Memory Systems for Modern Workloads" and subtitle "Accelerating Modern Workloads on a General-purpose PIM System" are at the top. Below the title, it says "Dr. Juan Gómez Luna Professor Onur Mutlu". The video player shows the ETH Zürich logo and the date "Sunday, March 26, 2023". The video has 33 views and was streamed 7 days ago. The URL "https://www.youtube.com/live/oYCaLcT0Km_o?feature=share" is shown below the video player.

https://www.youtube.com/live/oYCaLcT0Km_o?feature=share

<https://events.safari.ethz.ch/asplos-pim-tutorial/doku.php?id=start>

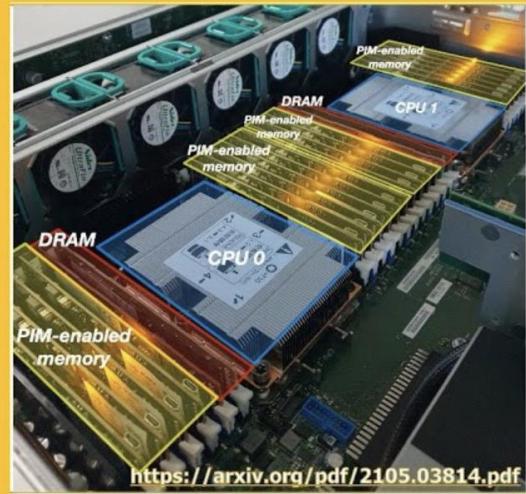
Real PIM Tutorial (ISCA 2023)

ISCA 2023 Real-World PIM Tutorial Sunday, June 18, Orlando, Florida

Organizers: Juan Gómez Luna, Onur Mutlu, Ataberk Olgun
Program: <https://events.safari.ethz.ch/isca-pim-tutorial/>



Overview PIM | PNM | UPMEM PIM |
PNM for neural networks |
PNM for recommender systems |
PNM for ML workloads |
How to enable PIM? | PUM prototypes
Hands-on Labs: Benchmarking |
Accelerating real-world workloads



ISCA 2023 Tutorial: Real-world Processing-in-Memory Systems for Modern Workloads



Onur Mutlu Lectures
33.4K subscribers



Subscribed



8



1



Share



Save

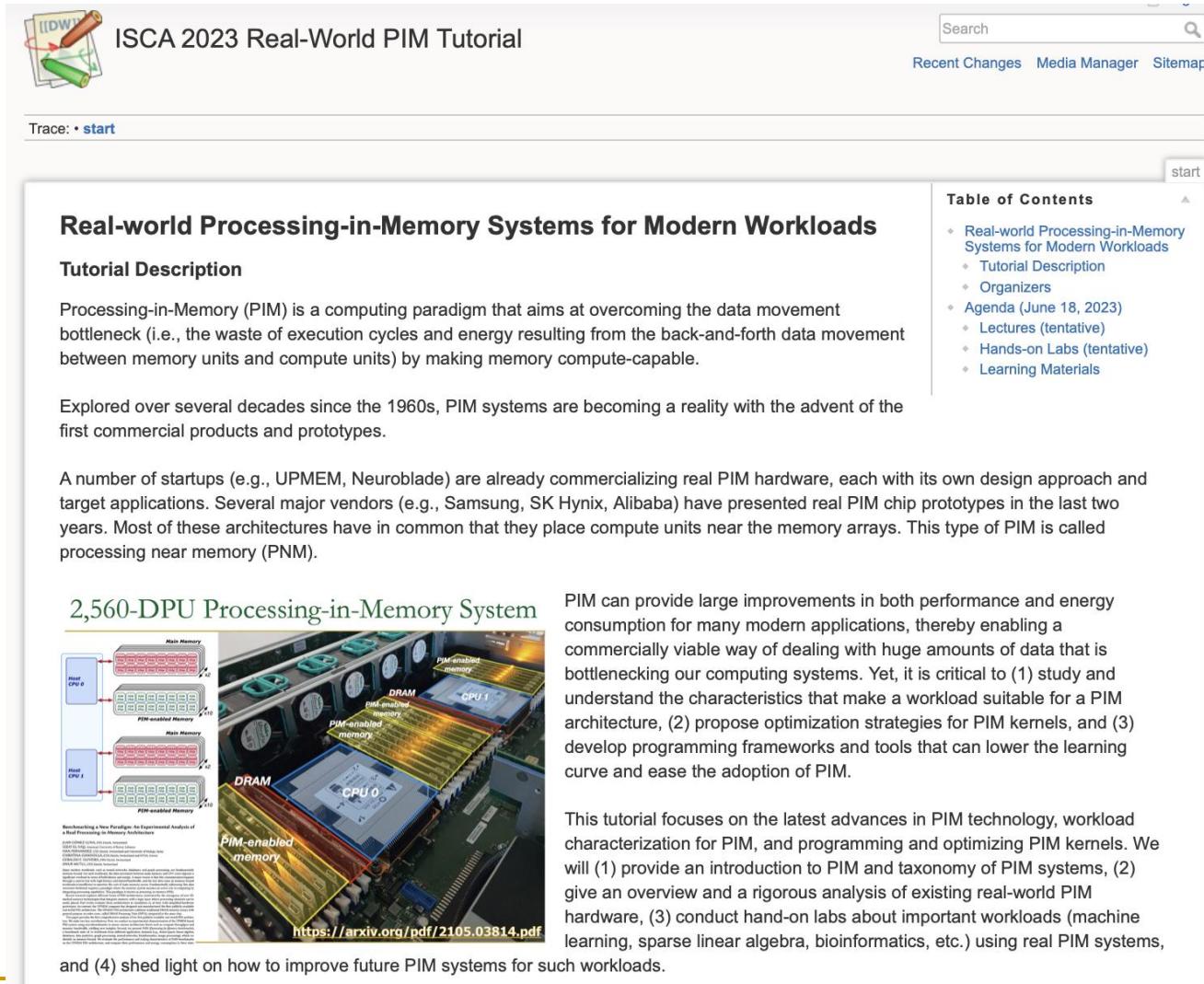


1 waiting Scheduled for Jun 18, 2023 Livestream - Data-Centric Architectures: Fundamentally Improving Performance and Energy (Spring 2023)

ISCA 2023 Tutorial: Real-world Processing-in-Memory Systems for Modern Workloads

Real PIM Tutorial (ISCA 2023)

■ June 18th: Lectures + Hands-on labs + Invited lectures



The screenshot shows the homepage of the ISCA 2023 Real-World PIM Tutorial. The header includes the logo, title, search bar, and navigation links for Recent Changes, Media Manager, and Sitemap. The main content area features a section titled "Real-world Processing-in-Memory Systems for Modern Workloads" with a "Tutorial Description" and a "Table of Contents". Below this is a diagram of a 2,560-DPU Processing-in-Memory System and a photograph of a circuit board with PIM-enabled memory components.

Real-world Processing-in-Memory Systems for Modern Workloads

Tutorial Description

Processing-in-Memory (PIM) is a computing paradigm that aims at overcoming the data movement bottleneck (i.e., the waste of execution cycles and energy resulting from the back-and-forth data movement between memory units and compute units) by making memory compute-capable.

Explored over several decades since the 1960s, PIM systems are becoming a reality with the advent of the first commercial products and prototypes.

A number of startups (e.g., UPMEM, Neuroblade) are already commercializing real PIM hardware, each with its own design approach and target applications. Several major vendors (e.g., Samsung, SK Hynix, Alibaba) have presented real PIM chip prototypes in the last two years. Most of these architectures have in common that they place compute units near the memory arrays. This type of PIM is called processing near memory (PNM).

2,560-DPU Processing-in-Memory System

PIM can provide large improvements in both performance and energy consumption for many modern applications, thereby enabling a commercially viable way of dealing with huge amounts of data that is bottlenecking our computing systems. Yet, it is critical to (1) study and understand the characteristics that make a workload suitable for a PIM architecture, (2) propose optimization strategies for PIM kernels, and (3) develop programming frameworks and tools that can lower the learning curve and ease the adoption of PIM.

This tutorial focuses on the latest advances in PIM technology, workload characterization for PIM, and programming and optimizing PIM kernels. We will (1) provide an introduction to PIM and taxonomy of PIM systems, (2) give an overview and a rigorous analysis of existing real-world PIM hardware, (3) conduct hand-on labs about important workloads (machine learning, sparse linear algebra, bioinformatics, etc.) using real PIM systems, and (4) shed light on how to improve future PIM systems for such workloads.

Real PIM Tutorial (MICRO 2023)

- October 29th: Lectures + Hands-on labs + Invited lectures

 MICRO 2023 Real-World PIM Tutorial

Trace: • [start](#)

[Search](#) [Recent Changes](#) [Media Manager](#) [Sitemap](#)

[Edit](#)

Real-world Processing-in-Memory Systems for Modern Workloads

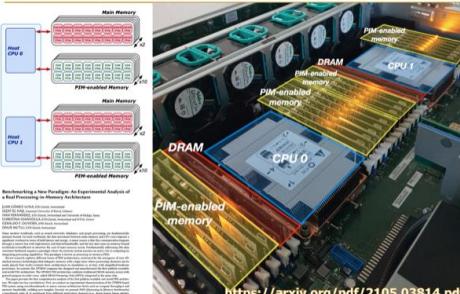
Tutorial Description

Processing-in-Memory (PIM) is a computing paradigm that aims at overcoming the data movement bottleneck (i.e., the waste of execution cycles and energy resulting from the back-and-forth data movement between memory units and compute units) by making memory compute-capable.

Explored over several decades since the 1960s, PIM systems are becoming a reality with the advent of the first commercial products and prototypes.

A number of startups (e.g., UPMEM, Neuroblade) are already commercializing real PIM hardware, each with its own design approach and target applications. Several major vendors (e.g., Samsung, SK Hynix, Alibaba) have presented real PIM chip prototypes in the last two years. Most of these architectures have in common that they place compute units near the memory arrays. This type of PIM is called processing near memory (PNM).

2,560-DPU Processing-in-Memory System



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

PIM can provide large improvements in both performance and energy consumption for many modern applications, thereby enabling a commercially viable way of dealing with huge amounts of data that is bottlenecking our computing systems. Yet, it is critical to (1) study and understand the characteristics that make a workload suitable for a PIM architecture, (2) propose optimization strategies for PIM kernels, and (3) develop programming frameworks and tools that can lower the learning curve and ease the adoption of PIM.

This tutorial focuses on the latest advances in PIM technology, workload characterization for PIM, and programming and optimizing PIM kernels. We will (1) provide an introduction to PIM and taxonomy of PIM systems, (2) give an overview and a rigorous analysis of existing real-world PIM hardware, (3) conduct hand-on labs about important workloads (machine learning, sparse linear algebra, bioinformatics, etc.) using real PIM systems, and (4) shed light on how to improve future PIM systems for such workloads.

Livestream

 [YouTube livestream](#)

[Edit](#)

Table of Contents

- Real-world Processing-in-Memory Systems for Modern Workloads
 - Tutorial Description
 - Livestream
 - Organizers
- Agenda (Tentative, October 29, 2023)
 - Lectures
 - Learning Materials

<https://youtube.com/live/ohU00NSIxOI?feature=share>

Principle: Insight and Ideas

Focus on Insight

Encourage New Ideas

Principle: Learning and Scholarship

Focus on
learning and scholarship

Principle: Environment of Freedom

Create an environment
that values
free exploration,
openness, collaboration,
hard work, creativity

Principle: Learning and Scholarship

The quality of your work
defines your impact

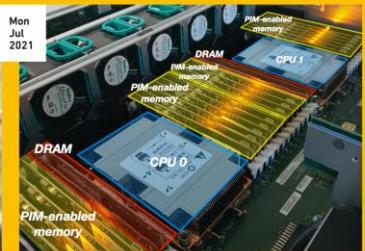
SAFARI Live Seminars

SAFARI Live Seminars in Computer Architecture



Dr. Juan Gómez Luna, ETH Zurich
Understanding a Modern Processing-in-Memory Architecture: Benchmarking and Experimental Characterization

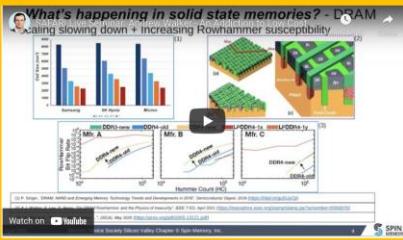
12
Mon
Jul
2021



SAFARI Live Seminars in Computer Architecture

Dr. Andrew Walker, Schiltron Corporation & Nexgen Power Systems
An Addiction to Low Cost Per Memory Bit – How to Recognize it and What to Do About it

19
Mo
Jul
2021



SAFARI Live Seminars in Computer Architecture

Geraldo F. Oliveira, ETH Zurich
DAMOV: A New Methodology and Benchmark Suite for Evaluating Data Movement Bottlenecks

22
Do
Jul
2021



Near-Data Processing (2/2)



Near-DRAM-banks processing for general-purpose computing
0.9 TOPS compute throughput¹



1.2 TFLOPS compute throughput²

The goal of Near-Data Processing (NDP) is to mitigate data movement

SAFARI
Real-time Random Number Generation Using DRAM Cells

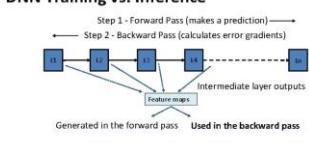
SAFARI Live Seminars in Computer Architecture



Gennady Pekhimenko, University of Toronto
Efficient DNN Training at Scale: from Algorithms to Hardware

5
Do
Aug
2021

DNN Training vs. Inference



SAFARI Live Seminars in Computer Architecture



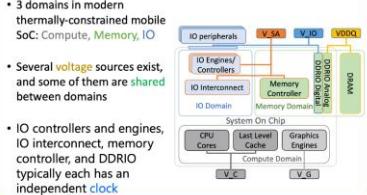
Jawad Haj-Yahya, Huawei Research Center Zurich
Power Management Mechanisms in Modern Microprocessors and Their Security Implications

16
Mo
Aug
2021



Overview of a Modern SoC Architecture

- 3 domains in modern thermally-constrained mobile SoC: Compute, Memory, IO
- Several voltage sources exist, and some of them are shared between domains
- IO controllers and engines, IO interconnect, memory controller, and DDRIO typically each has an independent clock



SAFARI Live Seminars in Computer Architecture

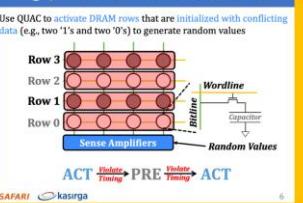


Ataberk Olgun, TOBB & ETH Zurich
QUAC-TRNG: High-Throughput True Random Number Generation Using Quadruple Row Activation in Commodity DRAM Chips

15
Mi
Sep
2021



Using QUAC to Generate Random Values

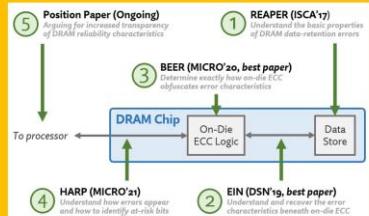


SAFARI Live Seminars in Computer Architecture



Minesh Patel, ETH Zurich
Enabling Effective Error Mitigation in Memory Chips That Use On-Die ECCs

21
Tues
Sep
2021

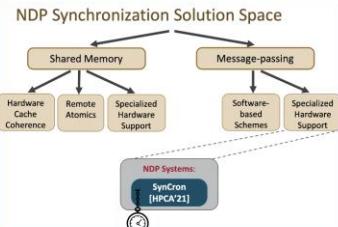


SAFARI Live Seminars in Computer Architecture



Christina Giannoula, National Technical University of Athens
Efficient Synchronization Support for Near-Data-Processing Architectures

27
Mo
Sep
2021



SAFARI Live Seminars in Computer Architecture



Jawad Haj-Yahya, Huawei Research Center Zurich
Security Implications of Power Management Mechanisms in Modern Processors, Current Studies and Future Trends

4
Mo
Okt
2021



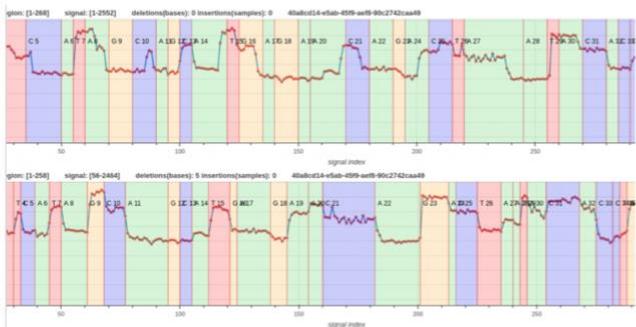
Experimental Methodology

- We experimentally study three modern Intel processors
 - Haswell, Coffee Lake, and Cannon Lake
- We measure voltage and current using a Data Acquisition card (NI-DAQ)
 - Host Computer
 - NI-DAQ
 - Processor & Board Computer
 - CPU Cores VR
 - Sense Resistor

SAFARI Live Seminars (Yesterday)

SAFARI Live Seminars in Computer Architecture

An Ecosystem for Scalable & Computationally Efficient Nanopore Data Processing



SPEAKER
Hasindu
Gamaarachchi
UNSW Sydney

ETH zürich

SAFARI
SAFARI Research Group



UNSW
SYDNEY



Garvan Institute
of Medical Research

Sept 27, 2023 9AM CEST

SAFARI Live Seminar: Hasindu Gamaarachchi, September 27 2023

Posted on September 18, 2023 by ewent

Join us for our upcoming **SAFARI Live Seminar**:

Speaker: Hasindu Gamaarachchi, UNSW Sydney

Date: Wednesday, September 27 2023, 9:00 Zurich time (CEST)

Where: Livestream on YouTube ([Link](#))

Title: An Ecosystem for Scalable & Computationally Efficient Nanopore Data Processing

<https://safari.ethz.ch/safari-seminar-series/>

SAFARI Live Seminars (Upcoming Talks)

SAFARI Live Seminars in Computer Architecture

Reshaping DRAM Scaling by
Enabling System-Memory
Cooperation



SPEAKER
Minesh Patel
SAFARI Research Group
(former PhD student)

ETHzürich **SAFARI**
SAFARI Research Group

Oct 4, 2023 2PM CEST

SAFARI Live Seminar: Minesh Patel, October 4 2023

Posted on September 22, 2023 by ewent

Join us for our upcoming **SAFARI Live Seminar**:

Speaker: **Minesh Patel**, SAFARI Research Group, ETH Zurich (former PhD student)

Date: Wednesday, October 4 2023, 14:00 Zurich time (CEST)

Where: HG E23 & Livestream on YouTube ([Link](#))

Title: Reshaping DRAM Scaling by Enabling System-Memory Cooperation

SAFARI Live Seminars (Upcoming Talks)

SAFARI Live Seminars in Computer Architecture

How does one bit-flip corrupt
an entire deep neural network,
and what to do about it



SPEAKER
Yanjing Li
University of Chicago



Oct 17, 2023 6PM CEST

SAFARI Live Seminar: Yanjing Li, October 17 2023

Posted on September 20, 2023 by ewent

Join us for our upcoming **SAFARI Live Seminar**:

Speaker: [Yanjing Li, University of Chicago](#)

Date: Tuesday, October 17 2023, 18:00 Zurich time (CEST)

Where: Livestream on YouTube ([Link](#))

Title: How does one bit-flip corrupt an entire deep neural network, and what to do about it

Open-Source Artifacts

<https://github.com/CMU-SAFARI>

Open Source Tools: SAFARI GitHub



SAFARI Research Group at ETH Zurich and Carnegie Mellon University

Site for source code and tools distribution from SAFARI Research Group at ETH Zurich and Carnegie Mellon University.

📍 ETH Zurich and Carnegie Mellon U... 🌐 <https://safari.ethz.ch/> 📩 omutlu@gmail.com

Overview

Repositories 71

Projects

Packages

Teams 1

People 44

Settings

Pinned

Customize pins

💻 **ramulator** Public

A Fast and Extensible DRAM Simulator, with built-in support for modeling many different DRAM technologies including DDRx, LPDDRx, GDDRx, WIOx, HBMx, and various academic proposals. Described in the...

● C++ ⭐ 311 🏷 161

💻 **prim-benchmarks** Public

PrIM (Processing-In-Memory benchmarks) is the first benchmark suite for a real-world processing-in-memory (PIM) architecture. PrIM is developed to evaluate, analyze, and characterize the first publ...

● C ⭐ 53 🏷 21

💻 **DAMOV** Public

DAMOV is a benchmark suite and a methodical framework targeting the study of data movement bottlenecks in modern applications. It is intended to study new architectures, such as near-data processin...

● C++ ⭐ 26 🏷 4

💻 **SneakySnake** Public

SneakySnake is the first and the only pre-alignment filtering algorithm that works efficiently and fast on modern CPU, FPGA, and GPU architectures. It greatly (by more than two orders of magnitude...

● VHDL ⭐ 41 🏷 8

💻 **MQSim** Public

MQSim is a fast and accurate simulator modeling the performance of modern multi-queue (MQ) SSDs as well as traditional SATA based SSDs. MQSim faithfully models new high-bandwidth protocol implement...

● C++ ⭐ 146 🏷 93

💻 **rowhammer** Public

Source code for testing the Row Hammer error mechanism in DRAM devices. Described in the ISCA 2014 paper by Kim et al. at http://users.ece.cmu.edu/~omutlu/pub/dram-row-hammer_isca14.pdf.

● C ⭐ 189 🏷 41

<https://github.com/CMU-SAFARI/>

Some Open Source Tools (I)

- Rowhammer – Program to Induce RowHammer Errors
 - <https://github.com/CMU-SAFARI/rowhammer>
- Ramulator – Fast and Extensible DRAM Simulator
 - <https://github.com/CMU-SAFARI/ramulator>
- MemSim – Simple Memory Simulator
 - <https://github.com/CMU-SAFARI/memsim>
- NOCulator – Flexible Network-on-Chip Simulator
 - <https://github.com/CMU-SAFARI/NOCulator>
- SoftMC – FPGA-Based DRAM Testing Infrastructure
 - <https://github.com/CMU-SAFARI/SoftMC>
- Other open-source software from my group
 - <https://github.com/CMU-SAFARI/>

Some Open Source Tools (II)

- MQSim – A Fast Modern SSD Simulator
 - <https://github.com/CMU-SAFARI/MQSim>
- Mosaic – GPU Simulator Supporting Concurrent Applications
 - <https://github.com/CMU-SAFARI/Mosaic>
- IMPICA – Processing in 3D-Stacked Memory Simulator
 - <https://github.com/CMU-SAFARI/IMPICA>
- SMLA – Detailed 3D-Stacked Memory Simulator
 - <https://github.com/CMU-SAFARI/SMLA>
- HWASim – Simulator for Heterogeneous CPU-HWA Systems
 - <https://github.com/CMU-SAFARI/HWASim>
- Other open-source software from my group
 - <https://github.com/CMU-SAFARI/>

More Open Source Tools (III)



SAFARI Research Group at ETH Zurich and Carnegie Mellon University

Site for source code and tools distribution from SAFARI Research Group at ETH Zurich and Carnegie Mellon University.

📍 ETH Zurich and Carnegie Mellon U... 🌐 <https://safari.ethz.ch/> 📩 omutlu@gmail.com

🏠 Overview 🏡 Repositories 71 🏢 Projects 🏢 Packages 🏢 Teams 1 🏢 People 44 🏢 Settings

Pinned

Customize pins

💻 **ramulator** Public

A Fast and Extensible DRAM Simulator, with built-in support for modeling many different DRAM technologies including DDRx, LPDDRx, GDDRx, WIOx, HBMx, and various academic proposals. Described in the...

● C++ ⭐ 311 🏷 161

💻 **prim-benchmarks** Public

PrIM (Processing-In-Memory benchmarks) is the first benchmark suite for a real-world processing-in-memory (PIM) architecture. PrIM is developed to evaluate, analyze, and characterize the first publ...

● C ⭐ 53 🏷 21

💻 **DAMOV** Public

DAMOV is a benchmark suite and a methodical framework targeting the study of data movement bottlenecks in modern applications. It is intended to study new architectures, such as near-data processin...

● C++ ⭐ 26 🏷 4

💻 **SneakySnake** Public

SneakySnake is the first and the only pre-alignment filtering algorithm that works efficiently and fast on modern CPU, FPGA, and GPU architectures. It greatly (by more than two orders of magnitude...

● VHDL ⭐ 41 🏷 8

💻 **MQSim** Public

MQSim is a fast and accurate simulator modeling the performance of modern multi-queue (MQ) SSDs as well as traditional SATA based SSDs. MQSim faithfully models new high-bandwidth protocol implement...

● C++ ⭐ 146 🏷 93

💻 **rowhammer** Public

Source code for testing the Row Hammer error mechanism in DRAM devices. Described in the ISCA 2014 paper by Kim et al. at http://users.ece.cmu.edu/~omutlu/pub/dram-row-hammer_isca14.pdf.

● C ⭐ 189 🏷 41

<https://github.com/CMU-SAFARI/>

Pythia

A Customizable Hardware Prefetching Framework Using Online Reinforcement Learning.

[machine-learning](#) [reinforcement-learning](#) [prefetcher](#) [cache-replacement](#)
[branch-predictor](#) [champsim-simulator](#) [champsim-tracer](#)

● C++ ⚡ 1 ⚪ 0 ⚪ 0 ⚪ 10 Updated yesterday

BurstLink

ψ 0 ⚪ 0 ⚪ 0 ⚪ 10 Updated 21 days ago

MIG-7-PHY-DDR3-Controller

A DDR3 Controller that uses the Xilinx MIG-7 PHY to interface with DDR3 devices.

● Verilog ⚡ 1 ⚪ 1 ⚪ 0 ⚪ 10 Updated on Aug 22

Pythia-HDL

Implementation of Pythia: A Customizable Hardware Prefetching Framework Using Online Reinforcement Learning in Chisel HDL.

[machine-learning](#) [scala](#) [reinforcement-learning](#) [chisel](#) [chisel3](#) [firrtl](#) [hdl](#)

● Scala ⚡ MIT ψ 0 ⚪ 0 ⚪ 0 ⚪ 10 Updated on Jul 31

HARP (Private)

ψ 0 ⚪ 0 ⚪ 0 ⚪ 10 Updated on Jul 31

EINSim

DRAM error-correction code (ECC) simulator incorporating statistical error properties and DRAM design characteristics for inferring pre-correction error characteristics using only the post-correction errors. Described in the 2019 DSN paper by Patel et al.: https://people.inf.ethz.ch/omutlu/pub/understanding-and-modeling-in-DRAM-ECC_dsn19.pdf.

[simulator](#) [reliability](#) [statistical-inference](#) [dram](#) [error-correcting-codes](#)
[map-estimation](#) [error-correction](#)

● C++ ⚡ MIT ψ 0 ⚪ 5 ⚪ 0 ⚪ 10 Updated on Jul 29

DAMOV

DAMOV is a benchmark suite and a methodical framework targeting the study of data movement bottlenecks in modern applications. It is intended to study new architectures, such as near-data processing. Described by Oliveira et al. (preliminary version at <https://arxiv.org/pdf/2105.03725.pdf>)

● C++ ⚡ 1 ⚪ 12 ⚪ 1 ⚪ 10 Updated on Jul 13

MetaSys

Metasys is the first open-source FPGA-based infrastructure with a prototype in a RISC-V core, to enable the rapid implementation and evaluation of a wide range of cross-layer software/hardware cooperative techniques techniques in real hardware. Described in our pre-print: <https://arxiv.org/abs/2105.08123>

ψ 1 ⚪ 0 ⚪ 0 ⚪ 10 Updated on Jul 9

NATSA

NATSA is the first near-data-processing accelerator for time series analysis based on the Matrix Profile (SCRIMP) algorithm. NATSA exploits modern 3D-stacked High Bandwidth Memory (HBM) to enable efficient and fast matrix profile computation near memory. Described in ICCD 2020 by Fernandez et al.

https://people.inf.ethz.ch/omutlu/pub/NATSA_time-...

[accelerator](#) [hbm](#) [time-series-analysis](#) [matrix-profile](#) [near-data-processing](#)
[scrimp](#)

● C++ ⚡ 1 ⚪ 4 ⚪ 0 ⚪ 10 Updated on Jun 28

COVIDHunter

COVIDHunter 🦠: An accurate and flexible COVID-19 outbreak simulation model that forecasts the strength of future mitigation measures and the numbers of cases, hospitalizations, and deaths for a given day, while considering the potential effect of environmental conditions. Described by Alser et al. (preliminary version at <https://arxiv.org/abs/2...>)

[simulation](#) [epidemiology](#) [covid-19](#) [covid-19-data](#) [covid-19-tracker](#)
[reproduction-number](#) [covidhunter](#)

● Swift ⚡ MIT ⚡ 1 ⚪ 5 ⚪ 0 ⚪ 10 Updated on Jun 27

prim-benchmarks

PrIM (Processing-In-Memory benchmarks) is the first benchmark suite for a real-world processing-in-memory (PIM) architecture. PrIM is developed to evaluate, analyze, and characterize the first publicly-available real-world PIM architecture, the UPMEM PIM architecture. Described by Gómez-Luna et al. (preliminary version at <https://arxiv.org/abs/2...>)

● C ⚡ MIT ⚡ 8 ⚪ 18 ⚪ 0 ⚪ 10 Updated on Jun 16

SNP-Selective-Hiding

An optimization-based mechanism 🕵️🔒 to selectively hide the minimum number of overlapping SNPs among the family members 🧑 who participated in the genomic studies (i.e. GWAS). Our goal is to distort the dependencies among the family members in the original database for achieving better privacy without significantly degrading the data utility.

[gwas](#) [genomics](#) [data-privacy](#) [differential-privacy](#) [genomic-data-analysis](#)
[laplace-distribution](#) [genomic-privacy](#)

● MATLAB ⚡ 0 ⚪ 0 ⚪ 0 ⚪ 10 Updated on Jun 16

SneakySnake

SneakySnake🐍 is the first and the only pre-alignment filtering algorithm that works efficiently and fast on modern CPU, FPGA, and GPU architectures. It greatly (by more than two orders of magnitude) expedites sequence alignment calculation for both short and long reads. Described in the Bioinformatics (2020) by Alser et al.

[fpga](#) [gpu](#) [smith-waterman](#) [needleman-wunsch](#) [sequence-alignment](#)
[long-reads](#) [minimap2](#)

● VHDL ⚡ GPL-3.0 ⚡ 6 ⚪ 35 ⚪ 0 ⚪ 11 Updated on May 12

ramulator

A Fast and Extensible DRAM Simulator, with built-in support for modeling many different DRAM technologies including DDRx, LPDDRx, GDDRx, WIOx, HBMx, and various academic proposals. Described in the IEEE CAL 2015 paper by Kim et al. at http://users.ece.cmu.edu/~omutlu/pub/ramulator_dram_simulator-ieee-cal15.pdf

● C++ ⚡ MIT ⚡ 130 ⚪ 250 ⚪ 49 ⚪ 4 Updated on May 11

GenASM

Source code for the software implementations of the GenASM algorithms proposed in our MICRO 2020 paper: Senol Cali et al., "GenASM: A High-Performance, Low-Power Approximate String Matching Acceleration Framework for Genome Sequence Analysis" at <https://people.inf.ethz.ch/omutlu/pub/GenASM-approximate-string-matching-framework-for-genome-analys...>

[approximate-string-matching](#) [read-mapping](#) [hw-sw-co-design](#)
[read-alignment](#) [bitap-algorithm](#) [pre-alignment-filtering](#)
[genome-sequence-analysis](#)

● C ⚡ GPL-3.0 ⚡ 3 ⚪ 20 ⚪ 0 ⚪ 10 Updated on Mar 22

AirLift

AirLift is a tool that updates mapped reads from one reference genome to another. Unlike existing tools, it accounts for regions not shared between the two reference genomes.

Papers, Talks, Videos, Artifacts

- All are openly available at

[**https://people.inf.ethz.ch/omutlu/projects.htm**](https://people.inf.ethz.ch/omutlu/projects.htm)

[**http://scholar.google.com/citations?user=7XyGUGkAAAAJ&hl=en**](http://scholar.google.com/citations?user=7XyGUGkAAAAJ&hl=en)

[**https://www.youtube.com/onurmutlulectures**](https://www.youtube.com/onurmutlulectures)

[**https://github.com/CMU-SAFARI/**](https://github.com/CMU-SAFARI/)

Principle: Environment of Freedom

Create an environment
that values
free exploration,
openness, collaboration,
hard work, creativity

My Suggestions to You

Suggestion to Researchers: Principle: Passion

Follow Your Passion

**(Do not get derailed
by naysayers)**

Principle: Build Infrastructure

**Build Infrastructure to
Enable Your Passion**

Principle: Work Hard

Work Hard to
Enable Your Passion

Suggestion to Researchers: Principle: Resilience

Be Resilient

Principle: Learning and Scholarship

Focus on
learning and scholarship

Principle: Learning and Scholarship

The quality of your work
defines your impact

Principle: Good Mindset, Goals & Focus

You can make a
good impact
on the world

Research & Teaching: Some Overview Talks

<https://www.youtube.com/onurmutlulectures>

■ Future Computing Architectures

- https://www.youtube.com/watch?v=kgiZISOcGFM&list=PL5Q2soXY2Zi8D_5MGV6EnXEJHnV2YFBJI&index=1

■ Enabling In-Memory Computation

- https://www.youtube.com/watch?v=njX_14584Jw&list=PL5Q2soXY2Zi8D_5MGV6EnXEJHnV2YFBJI&index=16

■ Accelerating Genome Analysis

- https://www.youtube.com/watch?v=r7sn41IH-4A&list=PL5Q2soXY2Zi8D_5MGV6EnXEJHnV2YFBJI&index=41

■ Rethinking Memory System Design

- https://www.youtube.com/watch?v=F7xZLNMIY1E&list=PL5Q2soXY2Zi8D_5MGV6EnXEJHnV2YFBJI&index=3

■ Intelligent Architectures for Intelligent Machines

- https://www.youtube.com/watch?v=c6_LgzuNdkw&list=PL5Q2soXY2Zi8D_5MGV6EnXEJHnV2YFBJI&index=25

■ The Story of RowHammer

- https://www.youtube.com/watch?v=sgd7PHQQ1AI&list=PL5Q2soXY2Zi8D_5MGV6EnXEJHnV2YFBJI&index=39

An Interview on Research and Education

- **Computing Research and Education (@ ISCA 2019)**
 - https://www.youtube.com/watch?v=8ffSEKZhmvo&list=PL5Q2soXY2Zi_4oP9LdL3cc8G6NIjD2Ydz

- **Maurice Wilkes Award Speech (10 minutes)**
 - https://www.youtube.com/watch?v=tcQ3zZ3JpuA&list=PL5Q2soXY2Zi8D_5MGV6EnXEJHnV2YFBJI&index=15

More Thoughts and Suggestions

- Onur Mutlu,
"Some Reflections (on DRAM)"
Award Speech for ACM SIGARCH Maurice Wilkes Award, at the ISCA Awards Ceremony, Phoenix, AZ, USA, 25 June 2019.
[[Slides \(pptx\)](#) ([pdf](#))]
[[Video of Award Acceptance Speech \(Youtube; 10 minutes\)](#) ([Youku; 13 minutes](#))]
[[Video of Interview after Award Acceptance \(Youtube; 1 hour 6 minutes\)](#) ([Youku; 1 hour 6 minutes](#))]
[[News Article on "ACM SIGARCH Maurice Wilkes Award goes to Prof. Onur Mutlu"](#)]

- Onur Mutlu,
"How to Build an Impactful Research Group"
57th Design Automation Conference Early Career Workshop (DACE), Virtual, 19 July 2020.
[[Slides \(pptx\)](#) ([pdf](#))]

More Thoughts and Suggestions (II)

- Onur Mutlu,
"Computer Architecture: Why Is It So Important and Exciting Today?"
Invited Lecture at Izmir Institute of Technology (IYTE), Virtual, 16 October 2020.
[Slides (pptx) (pdf)]
[Talk Video (2 hours 12 minutes)]

- Onur Mutlu,
"Applying to Graduate School & Doing Impactful Research"
Invited Panel Talk at the 3rd Undergraduate Mentoring Workshop, held with the 48th International Symposium on Computer Architecture (ISCA), Virtual, 18 June 2021.
[Slides (pptx) (pdf)]
[Talk Video (50 minutes)]

A Talk on Impactful Research & Teaching

You are screen sharing

Applying to Grad School & Doing Impactful Research

Onur Mutlu
omutlu@gmail.com
<https://people.inf.ethz.ch/omutlu>

13 June 2020

Undergraduate Architecture Mentoring Workshop @ ISCA 2021

SAFARI **ETH zürich** **Carnegie Mellon**

0:27 / 50:31

Arch. Mentoring Workshop @ISCA'21 - Applying to Grad School & Doing Impactful Research - Onur Mutlu

1,563 views • Premiered Jun 16, 2021

74 likes 1 dislike ...

Onur Mutlu Lectures
17.2K subscribers

Panel talk at Undergraduate Architecture Mentoring Workshop at ISCA 2021
(<https://sites.google.com/wisc.edu/uar...>)

Required Reading

Richard Hamming

“You and Your Research”

Transcription of the
Bell Communications Research Colloquium Seminar
7 March 1986

<https://safari.ethz.ch/architecture/fall2023/lib/exe/fetch.php?media=youandyourresearch.pdf>

How to Approach This Course

“Formative Experience”

How to Approach This Course

**“High investment,
high return”**

How to Approach This Course

“Recorded lectures
allowed me to go over
the lectures when
necessary”

How to Approach This Course

“YouTube allows me to
watch the lectures on
my TV”

How to Approach This Course

**"The lecturer is very
responsive to
questions and remarks
from students"**

How to Approach This Course

“Perhaps even better
than in-person classes
as questions can be
asked asynchronously”

How to Approach This Course

**“Easy to understand
course format with
homework, labs, and
lectures”**

How to Approach This Course

**“Paper reviews +
assignments + labs,
a really great plan to
learn in a
comprehensive way”**

How to Approach This Course

“the course was
fantastic and I would
do it again at any
time”

How to Approach This Course

Learning experience

Long-term tradeoff
analysis

Critical thinking &
decision making

How to Approach This Course

Concepts & Ideas

Fundamentals

Cutting-edge

Hands-on learning

How to Approach This Course

Your mindset
will determine
what you
get out of the course

Required Reading on Mindset & More

Richard Hamming

“You and Your Research”

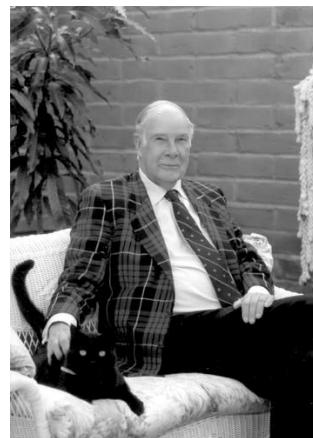
Transcription of the
Bell Communications Research Colloquium Seminar
7 March 1986

<https://safari.ethz.ch/architecture/fall2023/lib/exe/fetch.php?media=youandyourresearch.pdf>

Required Reading on Mindset & More

If you really want to be a first-class scientist you need to know yourself, your weaknesses, your strengths, and your bad faults, like my egotism. How can you convert a fault to an asset? How can you convert a situation where you haven't got enough manpower to move into a direction when that's exactly what you need to do? I say again that I have seen, as I studied the history, the successful scientist changed the viewpoint and what was a defect became an asset.

In summary, I claim that some of the reasons why so many people who have greatness within their grasp don't succeed are: they don't work on important problems, they don't become emotionally involved, they don't try and change what is difficult to some other situation which is easily done but is still important, and they keep giving themselves alibis why they don't. They keep saying that it is a matter of luck. I've told you how easy it is; furthermore I've told you how to reform. Therefore, go forth and become great scientists!



<https://safari.ethz.ch/architecture/fall2023/lib/exe/fetch.php?media=youandyourresearch.pdf>

Why Study Computer Architecture?

Computer Architecture

- is the **science** and **art** of designing **computing platforms** (hardware, interface, system SW, and programming model)
- to achieve a set of **design goals**
 - E.g., highest performance on earth on workloads X, Y, Z
 - E.g., longest battery life at a form factor that fits in your pocket with cost < \$\$\$ CHF
 - E.g., best average performance across all known workloads at the best performance/cost ratio
 - ...
 - Designing a supercomputer is different from designing a smartphone → But, many fundamental principles are similar

Different Platforms, Different Goals



Different Platforms, Different Goals



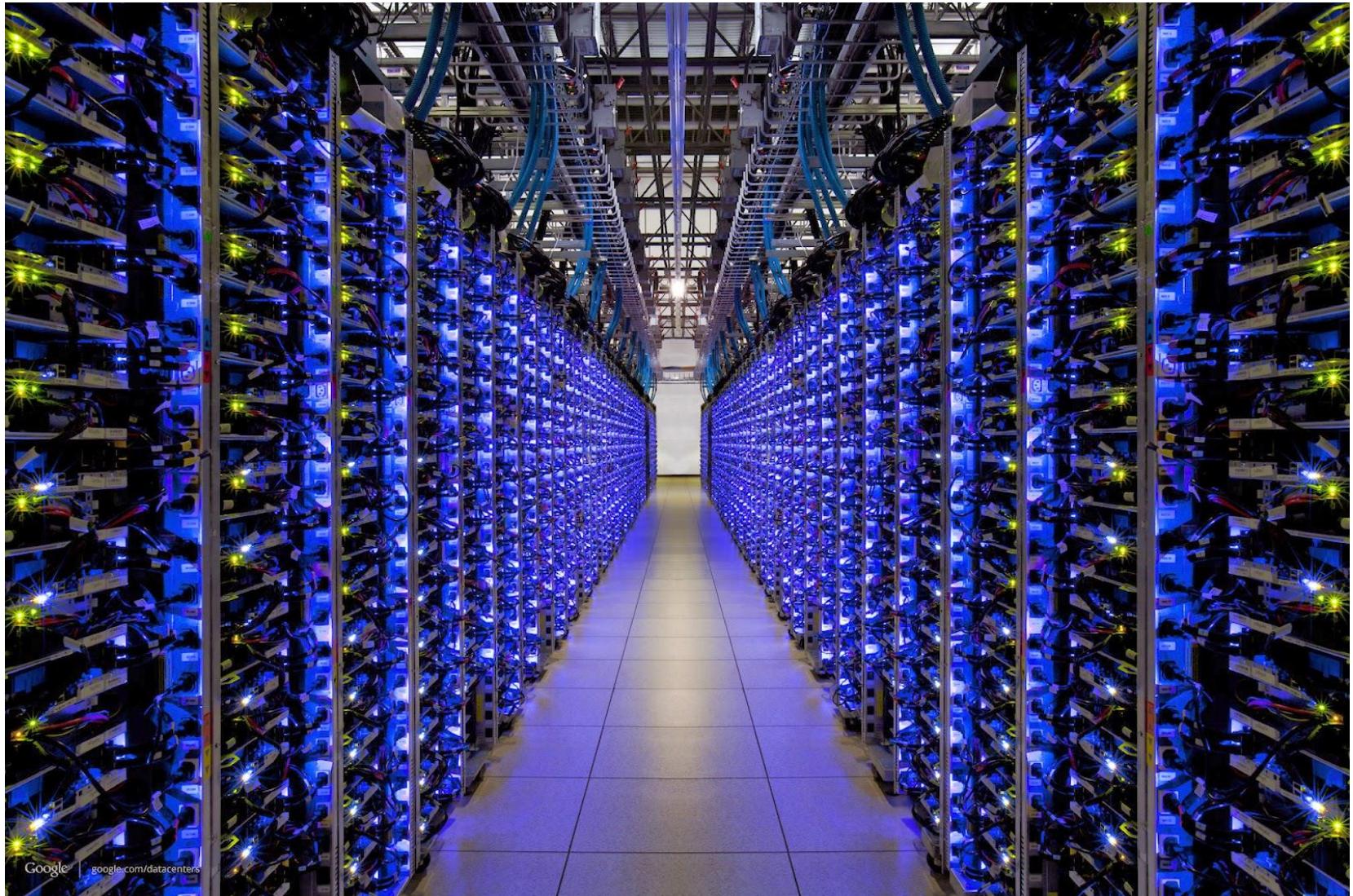
Different Platforms, Different Goals



Different Platforms, Different Goals



Different Platforms, Different Goals



Different Platforms, Different Goals

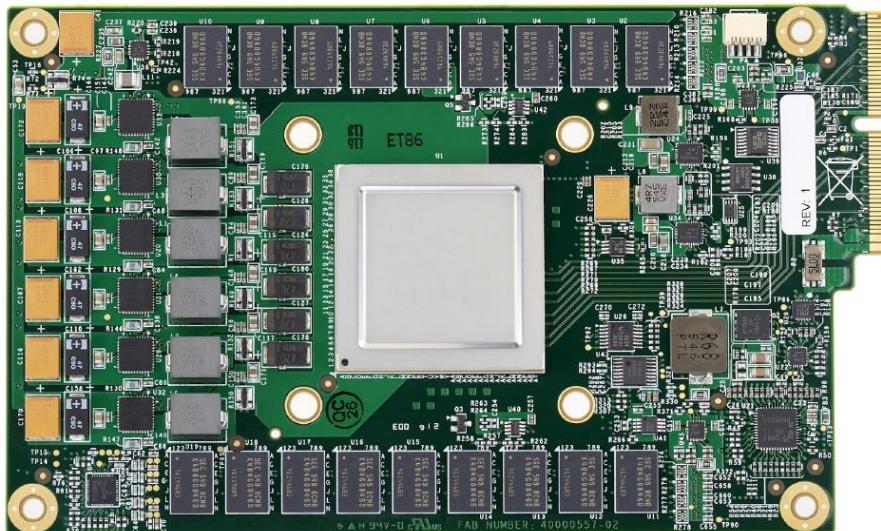


Jack Dongara

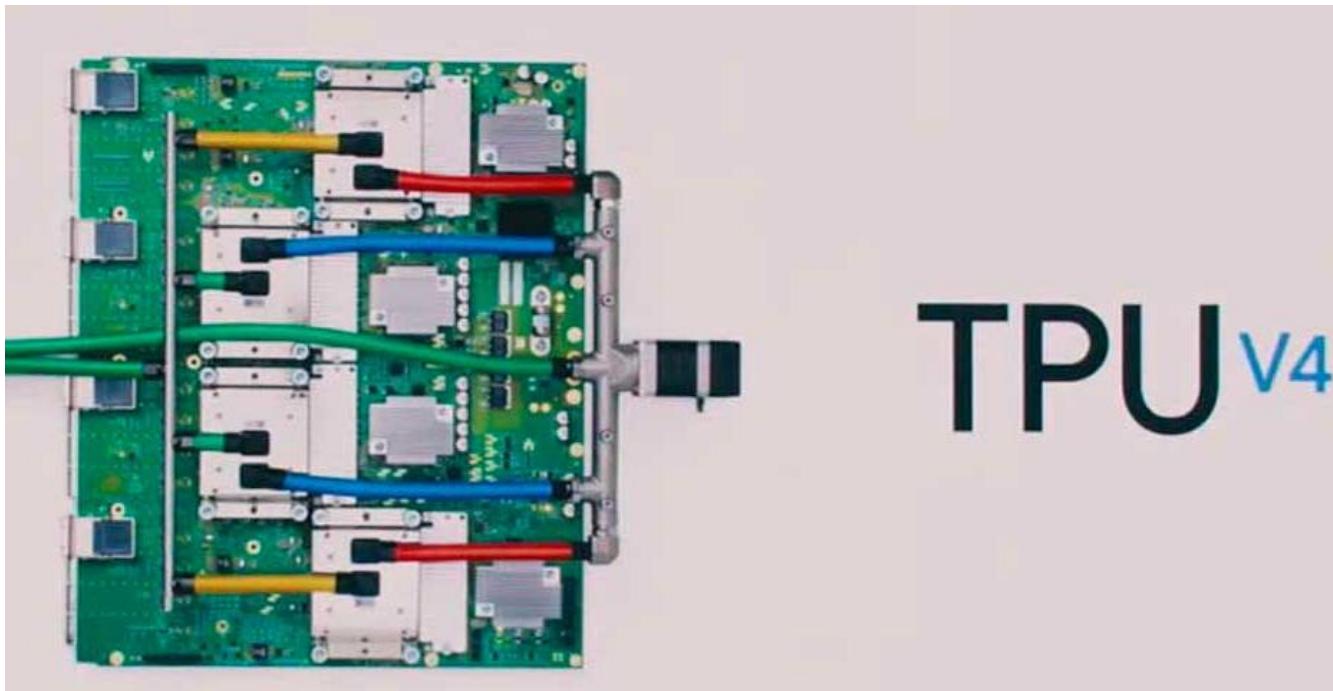
Different Platforms, Different Goals



Different Platforms, Different Goals



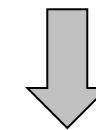
Different Platforms, Different Goals



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

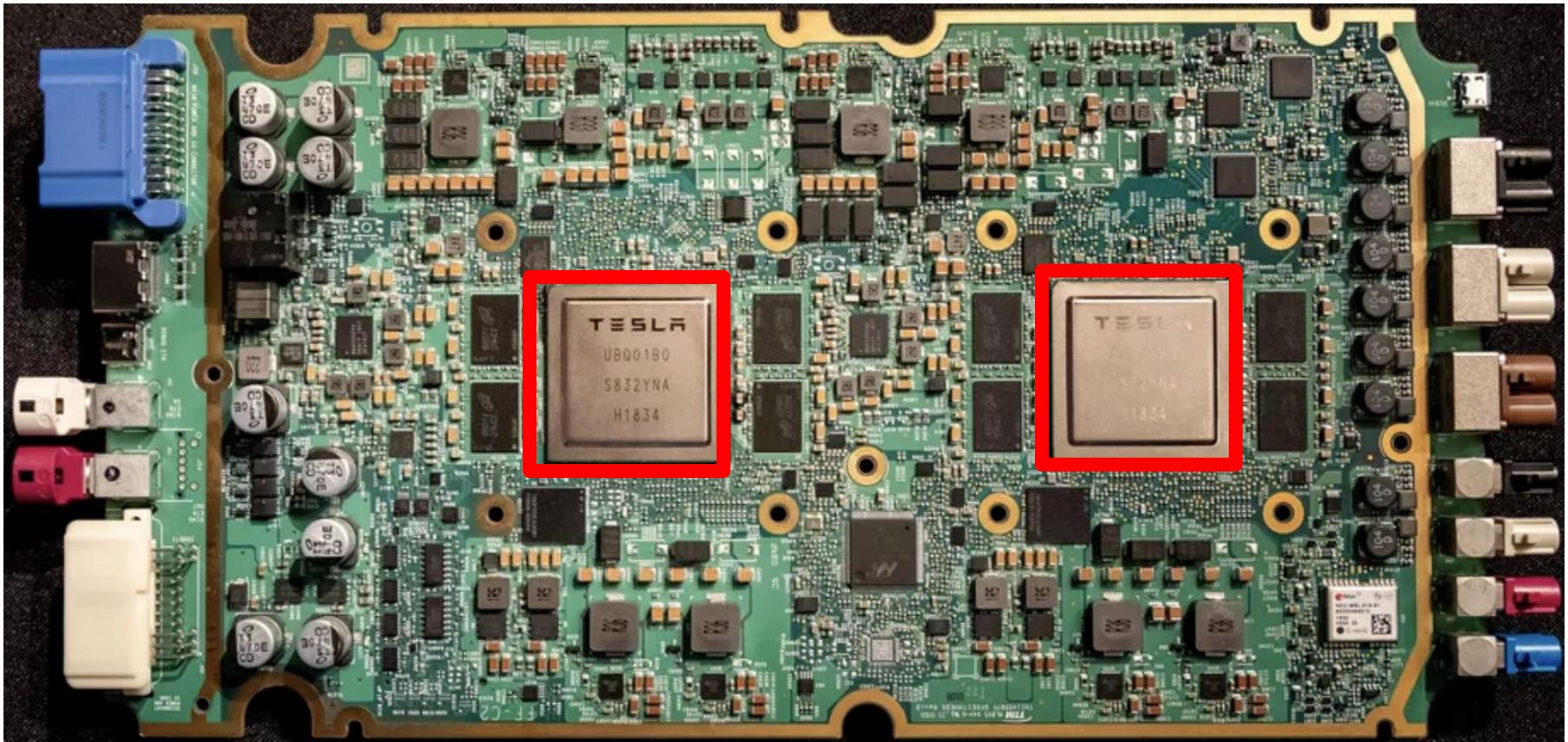


1 ExaFLOPS per board

<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.

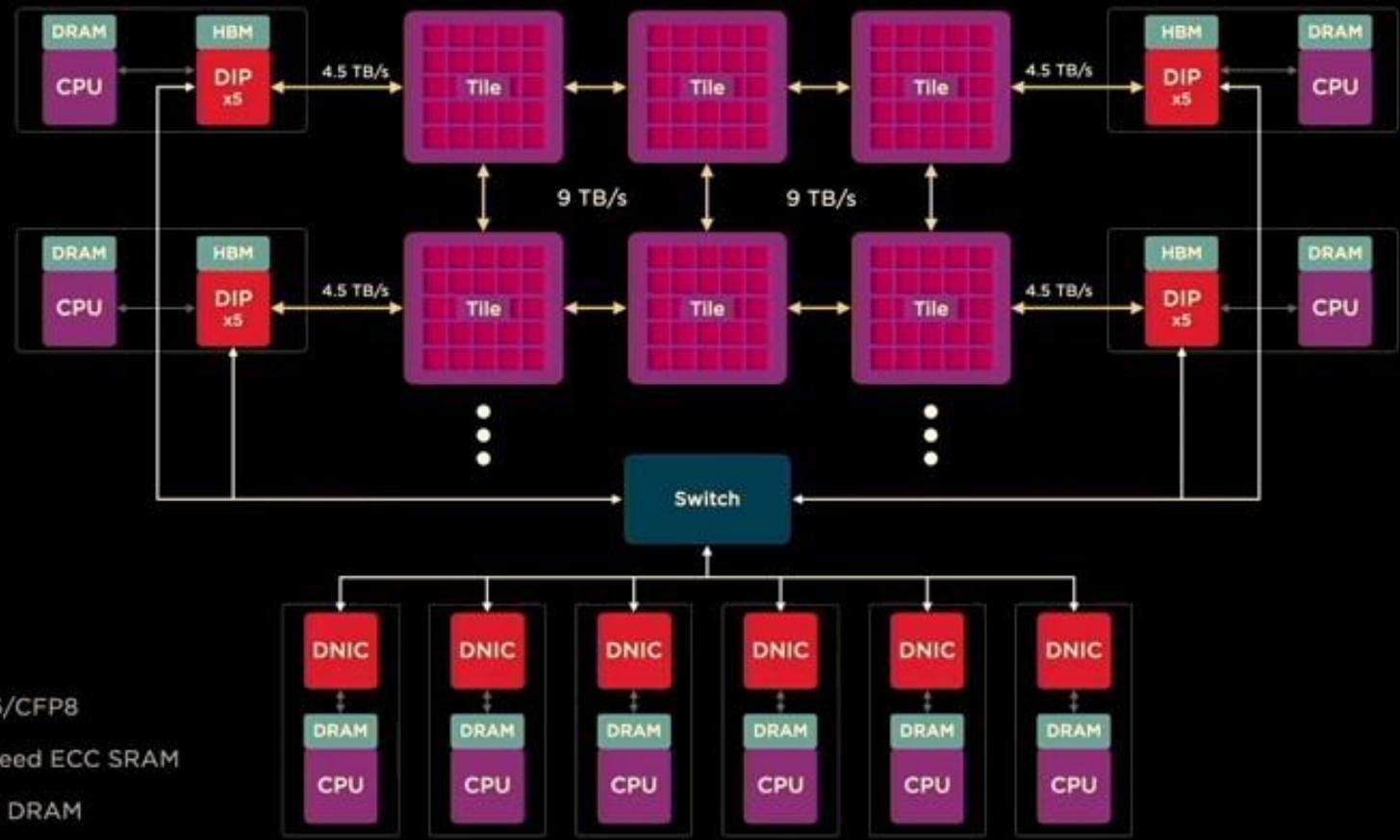


Different Platforms, Different Goals



Tesla Dojo Chip & System

V1 Dojo Training Matrix



Different Platforms, Different Goals



■ Tesla Dojo Chip & System

V1 Dojo Interface Processor

32GB High-Bandwidth Memory

- 800 GB/s Total Memory Bandwidth

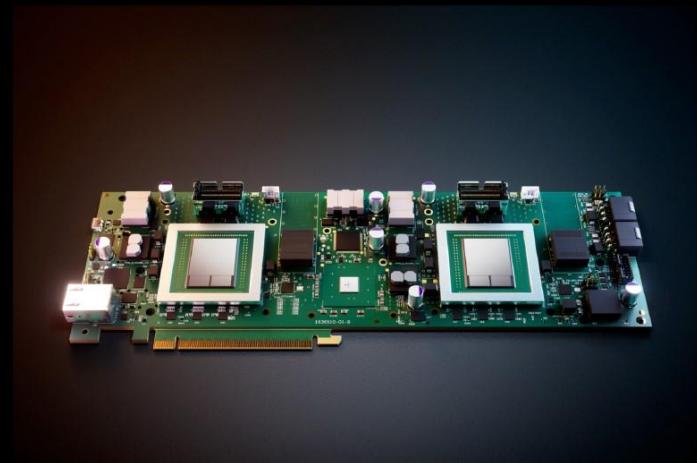
900 GB/s TTP Interface

- Tesla Transport Protocol (TTP) - Full custom protocol
- Provides full DRAM bandwidth to Training Tile

50 GB/s TTP over Ethernet (TTPoE)

- Enables extending communication over standard Ethernet
- Native hardware support

32 GB/s Gen4 PCIe Interface



Different Platforms, Different Goals



■ Tesla Dojo Chip & System

D1 Chip

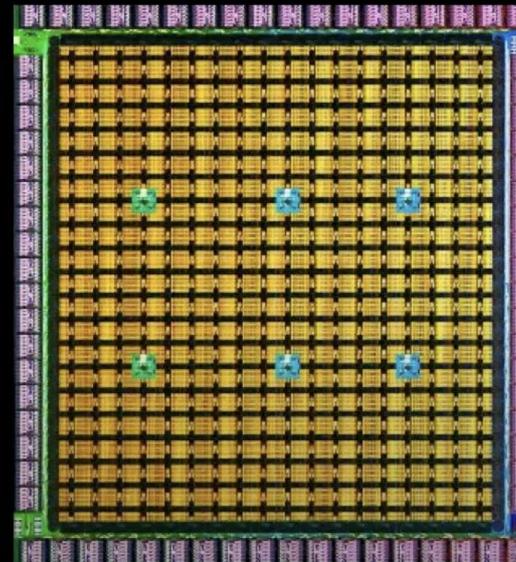
362 TFLOPs BF16/CFP8

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

The graph plots training compute power against time. The Y-axis represents power in units, ranging from 0 to 12,000. The X-axis shows dates from 08-19 to 08-21. A blue step-line shows discrete jumps in power, while a red dotted line shows a continuous exponential-like increase. The data points are approximately:

Date	Blue Step-Line Power (Units)	Red Dotted Line Power (Units)
08-19	~2000	~2000
11-19	~2000	~2000
02-20	~3500	~2500
05-20	~3500	~4000
08-20	~3500	~5500
11-20	~6000	~7000
02-21	~7000	~8500
05-21	~9000	~10000
08-21	~11000	~12000

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

A photograph of a massive server room, likely the Tesla Dojo facility, showing rows upon rows of server racks packed closely together under a complex ceiling of pipes and lights.

08-19 11-19 02-20 05-20 08-20 11-20 02-21 05-21 08-21

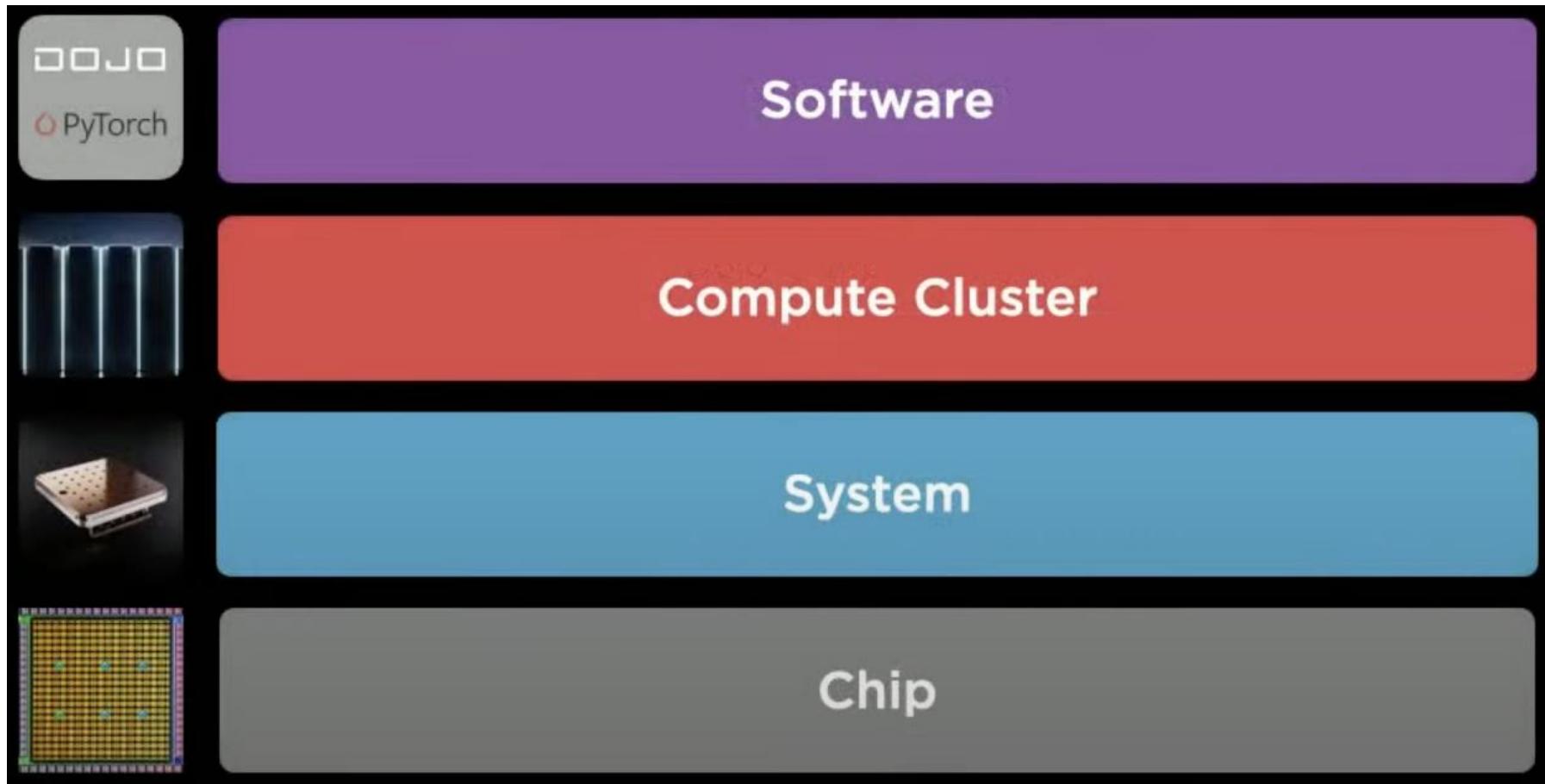
1:45:20 / 3:03:20 · Hardware Integration >

CC T 🔍 S 🔍 🔴 LIVE 🔴

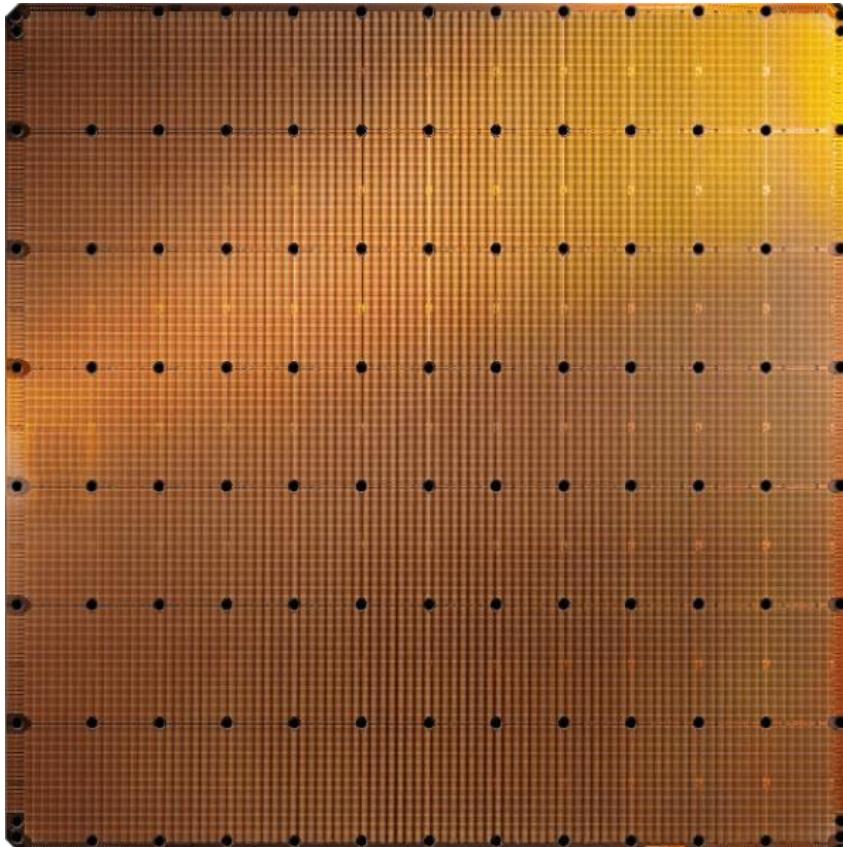
Different Platforms, Different Goals



- Tesla Dojo Chip & System



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

<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/>

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

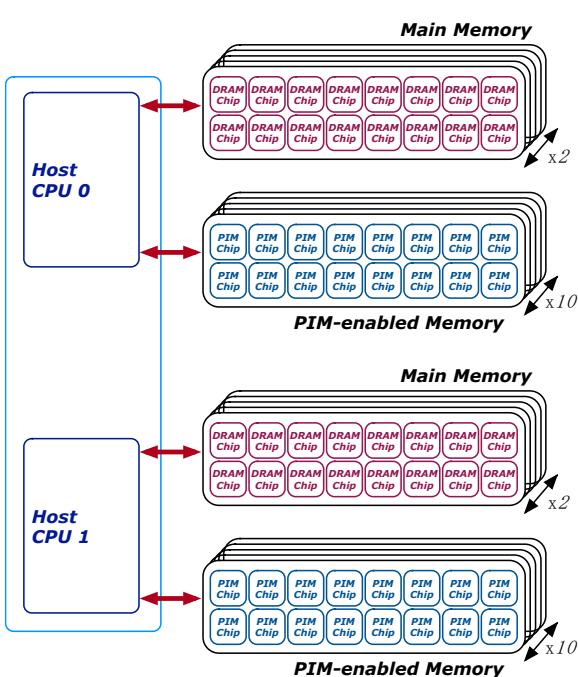
July-Aug. 2021, pp. 39-48, vol. 41

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 HAJI, American University of Beirut, Lebanon

IVAN FERNANDEZ, ETH Zürich, Switzerland and University of Málaga, Spain

CHRISTINA GIANNOULA, ETH Zürich, Switzerland and NTUA, Greece

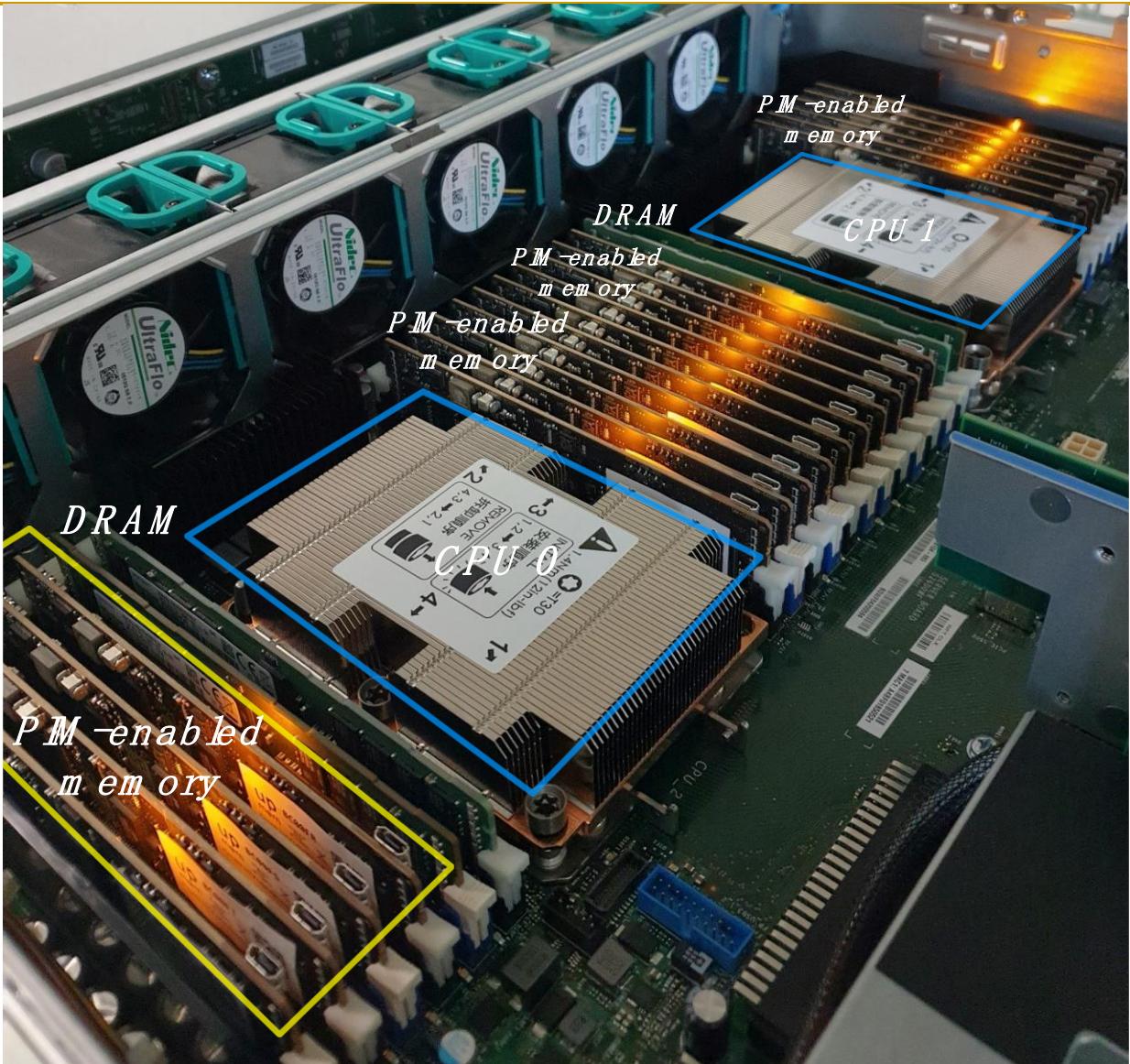
GERALDO F. OLIVEIRA, ETH Zürich, Switzerland

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 computation happens through a slow bus with high latency and limited bandwidth, and the low data reuse in memory-bound workloads is insufficient to amortize the cost of main memory access. Fundamentally addressing this *data movement bottleneck* requires a paradigm where the memory system assumes an active role in computing by integrating processing capabilities. This paradigm is known as *processing-in-memory* (PIM).

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

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

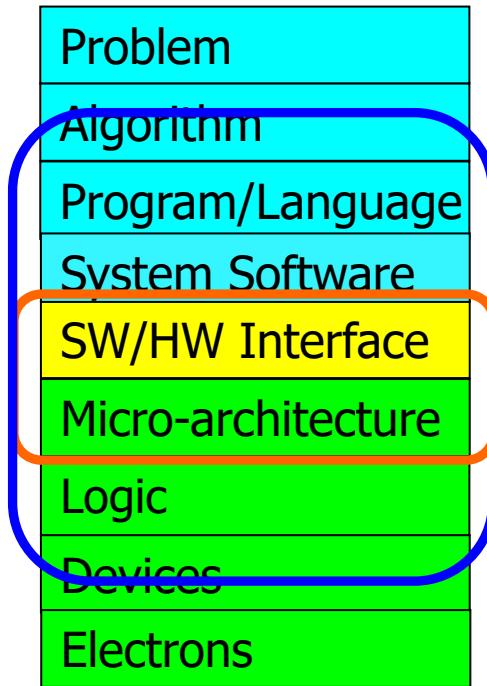


What is Computer Architecture?

- 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.

The Transformation Hierarchy

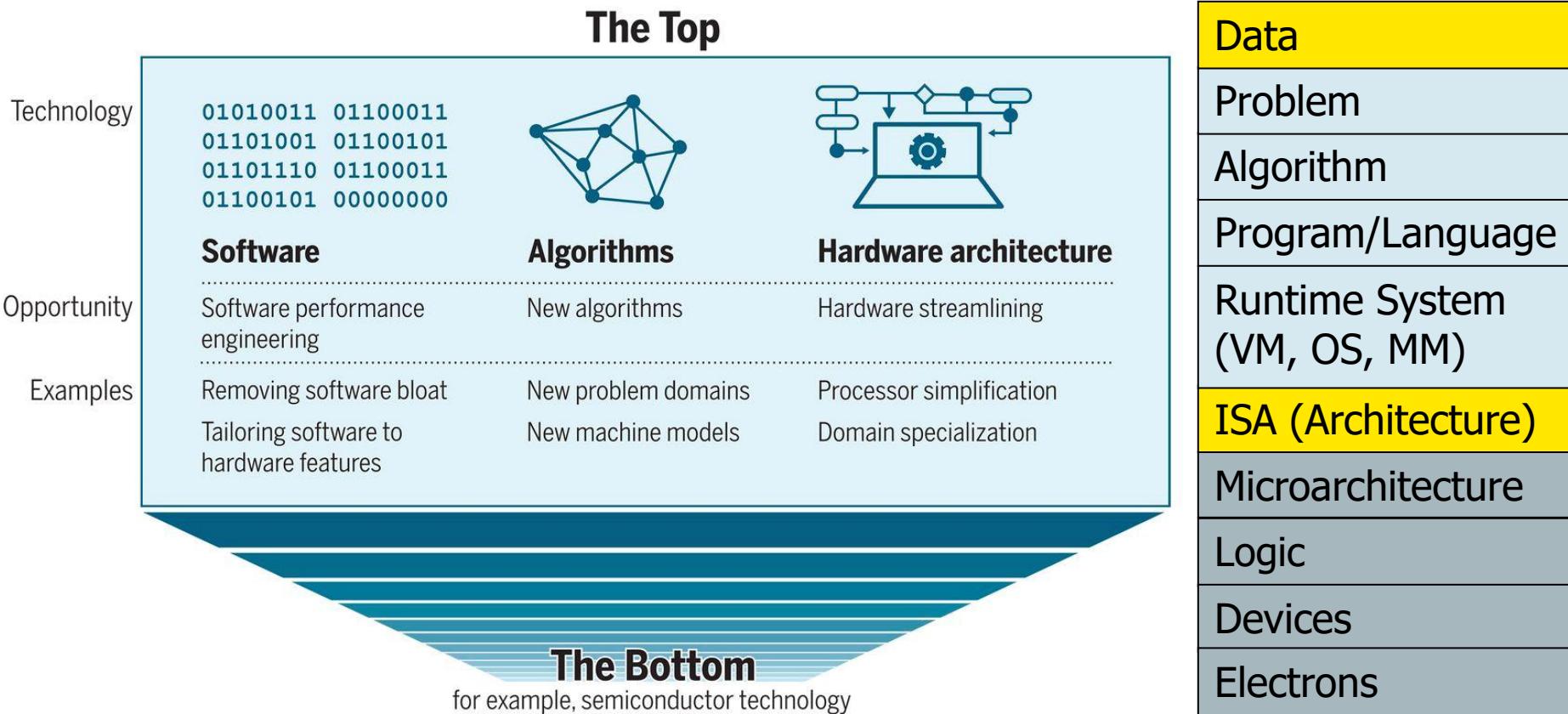
Computer Architecture
(expanded view)



Computer Architecture
(narrow view)

Computing System

Leiserson+, "[There's plenty of room at the Top: What will drive computer performance after Moore's law?](#)", Science, 2020



Richard Feynman, "[There's Plenty of Room at the Bottom: An Invitation to Enter a New Field of Physics](#)", a lecture given at Caltech, 1959.

Software & Hardware Optimizations

Multiplying Two 4096-by-4096 Matrices

```
for i in xrange(4096):  
    for j in xrange(4096):  
        for k in xrange(4096):  
            C[i][j] += A[i][k] *  
B[k][j]
```

The diagram shows the multiplication of two 3x3 matrices. The first matrix has columns 1, 2, 3 highlighted in yellow. The second matrix has rows 7, 8 highlighted in yellow. The result of the multiplication is shown in a third matrix, with the value 58 highlighted in yellow. Arrows point from the highlighted elements of the first matrix to the corresponding row of the second matrix, and from the second matrix to the result matrix.

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 7 & 8 \\ 9 & 10 \\ 11 & 12 \end{bmatrix} = \begin{bmatrix} 58 \end{bmatrix}$$

Implementation	Running time (s)	Absolute speedup
Python	25,552.48	1x
Java	2,372.68	11x
C	542.67	47x
Parallel loops	69.80	366x
Parallel divide and conquer	3.80	6,727x
plus vectorization	1.10	23,224x
plus AVX intrinsics	0.41	62,806x

Leiserson+, "[There's plenty of room at the Top: What will drive computer performance after Moore's law?](#)", Science, 2020

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**

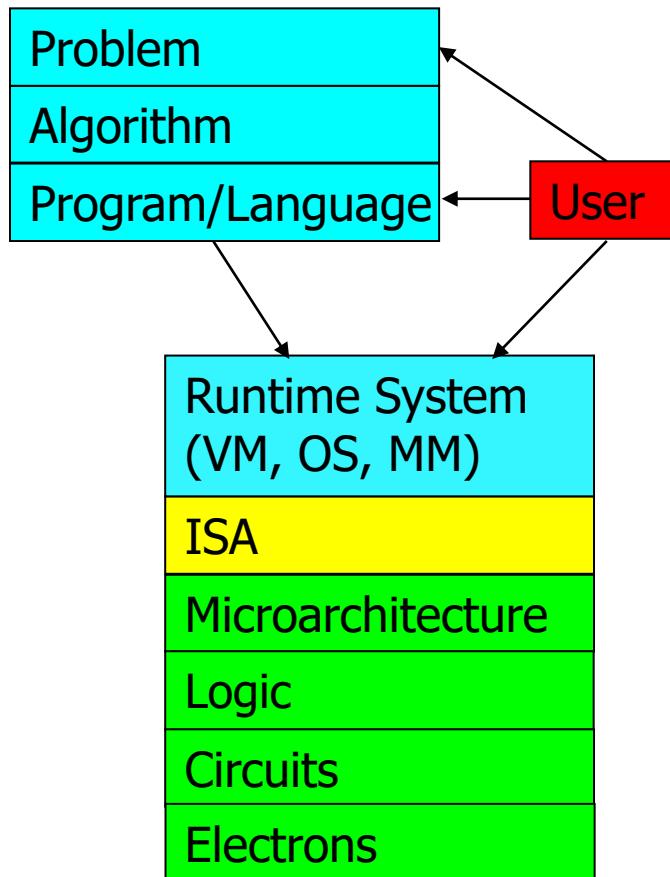
Computer Architecture Today (I)

- 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)

- These problems affect all parts of the computing stack – if we do not change the way we design systems

Many new demands
from the top
(Look Up)



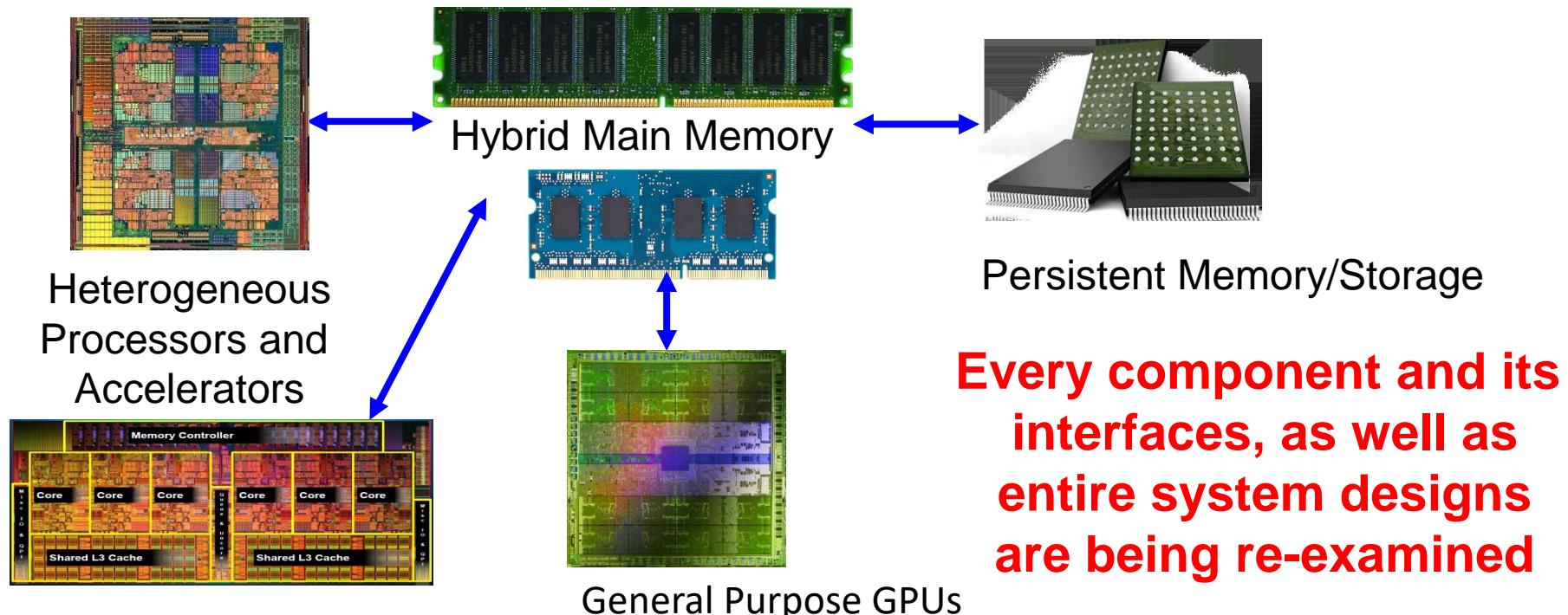
Fast changing
demands and
personalities
of users
(Look Up)

Many new issues
at the bottom
(Look Down)

- No clear, definitive answers to these problems

Computer Architecture Today (III)

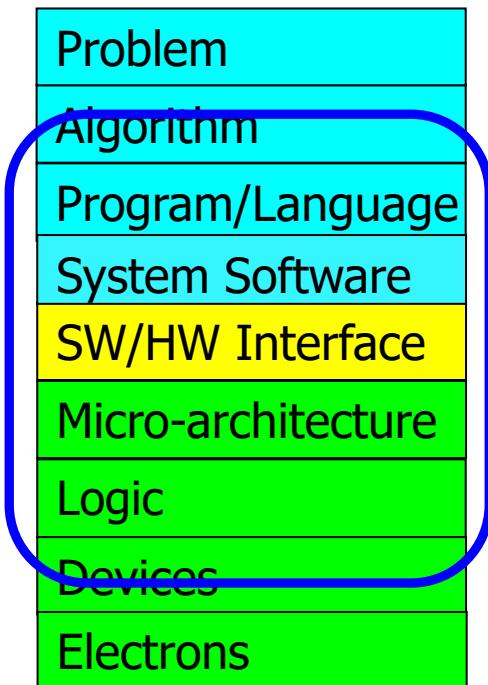
- Computing landscape is very different from 10-20 years ago
- Both UP (software and humanity trends) and DOWN (technologies and their issues), FORWARD and BACKWARD, and the resulting requirements and constraints



Axiom

To achieve the highest energy efficiency and performance:

we must take the expanded view
of computer architecture



**Co-design across the hierarchy:
Algorithms to devices**

**Specialize as much as possible
within the design goals**

Historical: Opportunities at the Bottom

There's Plenty of Room at the Bottom

From Wikipedia, the free encyclopedia

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

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**

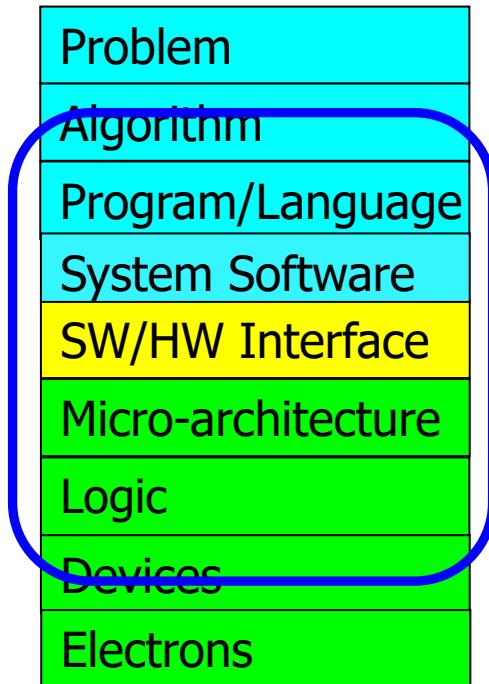
when you

communicate well between and optimize across

the top and the bottom

Hence the Expanded View

Computer Architecture
(expanded view)



Some Cross-Layer Design Examples (Foreshadowing)

EDEN: Data-Aware Efficient DNN Inference

- Skanda Koppula, Lois Orosa, A. Giray Yaglikci, Roknoddin Azizi, Taha Shahroodi, Konstantinos Kanellopoulos, and Onur Mutlu,
"EDEN: Enabling Energy-Efficient, High-Performance Deep Neural Network Inference Using Approximate DRAM"
Proceedings of the 52nd International Symposium on Microarchitecture (MICRO),
Columbus, OH, USA, October 2019.
[[Slides \(pptx\)](#) ([pdf](#))]
[[Lightning Talk Slides \(pptx\)](#) ([pdf](#))]
[[Poster \(pptx\)](#) ([pdf](#))]
[[Lightning Talk Video](#) (90 seconds)]
[[Full Talk Lecture](#) (38 minutes)]

EDEN: Enabling Energy-Efficient, High-Performance Deep Neural Network Inference Using Approximate DRAM

Skanda Koppula Lois Orosa A. Giray Yağlıkçı
Roknoddin Azizi Taha Shahroodi Konstantinos Kanellopoulos Onur Mutlu

ETH Zürich

SMASH: SW/HW Indexing Acceleration

- Konstantinos Kanellopoulos, Nandita Vijaykumar, Christina Giannoula, Roknoddin Azizi, Skanda Koppula, Nika Mansouri Ghiasi, Taha Shahroodi, Juan Gomez-Luna, and Onur Mutlu,

"SMASH: Co-designing Software Compression and Hardware-Accelerated Indexing for Efficient Sparse Matrix Operations"

Proceedings of the 52nd International Symposium on Microarchitecture (MICRO), Columbus, OH, USA, October 2019.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Lightning Talk Slides \(pptx\)](#) ([pdf](#))]

[[Poster \(pptx\)](#) ([pdf](#))]

[[Lightning Talk Video](#) (90 seconds)]

[[Full Talk Lecture](#) (30 minutes)]

SMASH: Co-designing Software Compression and Hardware-Accelerated Indexing for Efficient Sparse Matrix Operations

Konstantinos Kanellopoulos¹ Nandita Vijaykumar^{2,1} Christina Giannoula^{1,3} Roknoddin Azizi¹
Skanda Koppula¹ Nika Mansouri Ghiasi¹ Taha Shahroodi¹ Juan Gomez Luna¹ Onur Mutlu^{1,2}

GenASM: HW/SW Approximate String Matching Accelerator

- Damla Senol Cali, Gurpreet S. Kalsi, Zulal Bingol, 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*

SW/HW Climate Modeling Accelerator

- Gagandeep Singh, Dionysios Diamantopoulos, Christoph Hagleitner, Juan Gómez-Luna, Sander Stuijk, Onur Mutlu, and Henk Corporaal,

"NERO: A Near High-Bandwidth Memory Stencil Accelerator for Weather Prediction Modeling"

Proceedings of the 30th International Conference on Field-Programmable Logic and Applications (FPL), Gothenburg, Sweden, September 2020.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Lightning Talk Slides \(pptx\)](#) ([pdf](#))]

[[Talk Video](#) (23 minutes)]

Nominated for the Stamatis Vassiliadis Memorial Award.

NERO: A Near High-Bandwidth Memory Stencil Accelerator for Weather Prediction Modeling

Gagandeep Singh^{a,b,c}

Dionysios Diamantopoulos^c

Christoph Hagleitner^c

Juan Gómez-Luna^b

Sander Stuijk^a

Onur Mutlu^b

Henk Corporaal^a

^aEindhoven University of Technology

^bETH Zürich

^cIBM Research Europe, Zurich

HW/SW Time Series Analysis Accelerator

- Ivan Fernandez, Ricardo Quislant, Christina Giannoula, Mohammed Alser, Juan Gómez-Luna, Eladio Gutiérrez, Oscar Plata, and Onur Mutlu,
"NATSA: A Near-Data Processing Accelerator for Time Series Analysis"
Proceedings of the 38th IEEE International Conference on Computer Design (ICCD), Virtual, October 2020.
[[Slides \(pptx\)](#) ([pdf](#))]
[[Talk Video](#) (10 minutes)]
[[Source Code](#)]

NATSA: A Near-Data Processing Accelerator for Time Series Analysis

Ivan Fernandez[§]

Ricardo Quislant[§]

Christina Giannoula[†]

Mohammed Alser[‡]

Juan Gómez-Luna[‡]

Eladio Gutiérrez[§]

Oscar Plata[§]

Onur Mutlu[‡]

[§]*University of Malaga*

[†]*National Technical University of Athens*

[‡]*ETH Zürich*

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), 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*

Accelerating Genome Analysis

- Mohammed Alser, Zulal Bingol, 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

Graph Processing Accelerator w/ PIM

- Junwhan Ahn, Sungpack Hong, Sungjoo Yoo, Onur Mutlu, and Kiyoung Choi,

"A Scalable Processing-in-Memory Accelerator for Parallel Graph Processing"

Proceedings of the 42nd International Symposium on Computer Architecture (ISCA), Portland, OR, June 2015.
[Slides (pdf)] [Lightning Session Slides (pdf)]

A Scalable Processing-in-Memory Accelerator for Parallel Graph Processing

Junwhan Ahn Sungpack Hong[§] Sungjoo Yoo Onur Mutlu[†] Kiyoung Choi

junwhan@snu.ac.kr, sungpack.hong@oracle.com, sungjoo.yoo@gmail.com, onur@cmu.edu, kchoi@snu.ac.kr

Seoul National University

[§]Oracle Labs

[†]Carnegie Mellon University

Processing in Memory for Mobile Workloads

- 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.

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}

Accelerating Linked Data Structures

- Kevin Hsieh, Samira Khan, Nandita Vijaykumar, Kevin K. Chang, Amirali Boroumand, Saugata Ghose, and Onur Mutlu,

**"Accelerating Pointer Chasing in 3D-Stacked Memory:
Challenges, Mechanisms, Evaluation"**

*Proceedings of the 34th IEEE International Conference on Computer
Design (ICCD), Phoenix, AZ, USA, October 2016.*

Accelerating Pointer Chasing in 3D-Stacked Memory: Challenges, Mechanisms, Evaluation

Kevin Hsieh[†] Samira Khan[‡] Nandita Vijaykumar[†]
Kevin K. Chang[†] Amirali Boroumand[†] Saugata Ghose[†] Onur Mutlu^{§†}
[†]*Carnegie Mellon University* [‡]*University of Virginia* [§]*ETH Zürich*

Expressive (Memory) Interfaces

- Nandita Vijaykumar, Abhilasha Jain, Diptesh Majumdar, Kevin Hsieh, Gennady Pekhimenko, Eiman Ebrahimi, Nastaran Hajinazar, Phillip B. Gibbons and Onur Mutlu,
["A Case for Richer Cross-layer Abstractions: Bridging the Semantic Gap with Expressive Memory"](#)
Proceedings of the 45th International Symposium on Computer Architecture (ISCA),
Los Angeles, CA, USA, June 2018.
[\[Slides \(pptx\) \(pdf\)\]](#) [\[Lightning Talk Slides \(pptx\) \(pdf\)\]](#)
[\[Lightning Talk Video\]](#)

A Case for Richer Cross-layer Abstractions: Bridging the Semantic Gap with Expressive Memory

Nandita Vijaykumar^{†§} Abhilasha Jain[†] Diptesh Majumdar[†] Kevin Hsieh[†] Gennady Pekhimenko[‡]
Eiman Ebrahimi[¶] Nastaran Hajinazar[†] Phillip B. Gibbons[†] Onur Mutlu^{§†}

[†]**Carnegie Mellon University**

⁺**Simon Fraser University**

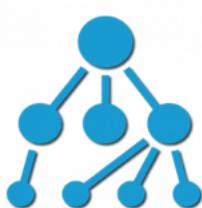
[‡]**University of Toronto**

[§]**ETH Zürich**

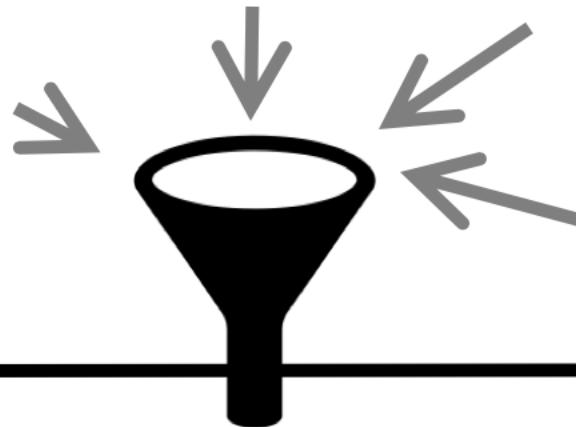
[¶]**NVIDIA**

One Problem: Limited SW/HW Communication

Higher-level information is not visible to HW

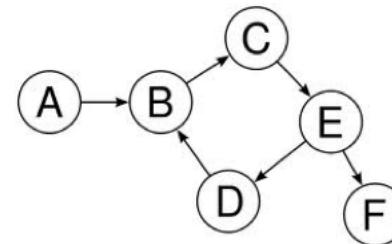


Software



10001111...
101010011...

Hardware



Access Patterns

Integer

Float

Data Type

Char

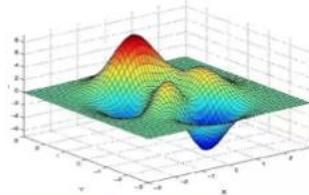
Instructions

Memory Addresses

A Solution: More Expressive Interfaces

Performance

Software



Functionality

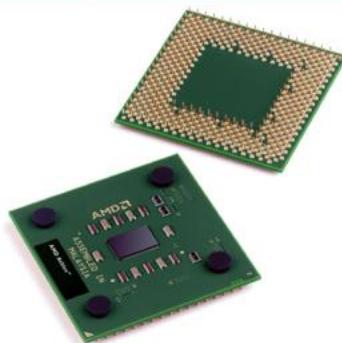


Higher-level
Program
Semantics

ISA
Virtual Memory

Expressive
Memory
“XMem”

Hardware



X-MeM Aids Many Optimizations

Table 1: Summary of the example memory optimizations that XMem aids.

Memory optimization	Example semantics provided by XMem (described in §3.3)	Example Benefits of XMem
Cache management	(i) Distinguishing between data structures or pools of similar data; (ii) Working set size; (iii) Data reuse	Enables: (i) applying different caching policies to different data structures or pools of data; (ii) avoiding cache thrashing by <i>knowing</i> the active working set size; (iii) bypassing/prioritizing data that has no/high reuse. (§5)
Page placement in DRAM e.g., [23, 24]	(i) Distinguishing between data structures; (ii) Access pattern; (iii) Access intensity	Enables page placement at the <i>data structure</i> granularity to (i) isolate data structures that have high row buffer locality and (ii) spread out concurrently-accessed irregular data structures across banks and channels to improve parallelism. (§6)
Cache/memory compression e.g., [25–32]	(i) Data type: integer, float, char; (ii) Data properties: sparse, pointer, data index	Enables using a <i>different compression algorithm</i> for each data structure based on data type and data properties, e.g., sparse data encodings, FP-specific compression, delta-based compression for pointers [27].
Data prefetching e.g., [33–36]	(i) Access pattern: strided, irregular, irregular but repeated (e.g., graphs), access stride; (ii) Data type: index, pointer	Enables (i) <i>highly accurate</i> software-driven prefetching while leveraging the benefits of hardware prefetching (e.g., by being memory bandwidth-aware, avoiding cache thrashing); (ii) using different prefetcher <i>types</i> for different data structures: e.g., stride [33], tile-based [20], pattern-based [34–37], data-based for indices/pointers [38, 39], etc.
DRAM cache management e.g., [40–46]	(i) Access intensity; (ii) Data reuse; (iii) Working set size	(i) Helps avoid cache thrashing by knowing working set size [44]; (ii) Better DRAM cache management via reuse behavior and access intensity information.
Approximation in memory e.g., [47–53]	(i) Distinguishing between pools of similar data; (ii) Data properties: tolerance towards approximation	Enables (i) each memory component to track how approximable data is (at a fine granularity) to inform approximation techniques; (ii) data placement in heterogeneous reliability memories [54].
Data placement: NUMA systems e.g., [55, 56]	(i) Data partitioning across threads (i.e., relating data to threads that access it); (ii) Read-Write properties	Reduces the need for profiling or data migration (i) to co-locate data with threads that access it and (ii) to identify Read-Only data, thereby enabling techniques such as replication.
Data placement: hybrid memories e.g., [16, 57, 58]	(i) Read-Write properties (Read-Only/Read-Write); (ii) Access intensity; (iii) Data structure size; (iv) Access pattern	Avoids the need for profiling/migration of data in hybrid memories to (i) effectively manage the asymmetric read-write properties in NVM (e.g., placing Read-Only data in the NVM) [16, 57]; (ii) make tradeoffs between data structure "hotness" and size to allocate fast/high bandwidth memory [14]; and (iii) leverage row-buffer locality in placement based on access pattern [45].
Managing NUCA systems e.g., [15, 59]	(i) Distinguishing pools of similar data; (ii) Access intensity; (iii) Read-Write or Private-Shared properties	(i) Enables using different cache policies for different data pools (similar to [15]); (ii) Reduces the need for reactive mechanisms that detect sharing and read-write characteristics to inform cache policies.

Expressive (Memory) Interfaces for GPUs

- Nandita Vijaykumar, Eiman Ebrahimi, Kevin Hsieh, Phillip B. Gibbons and Onur Mutlu,
"The Locality Descriptor: A Holistic Cross-Layer Abstraction to Express Data Locality in GPUs"

Proceedings of the 45th International Symposium on Computer Architecture (ISCA),
Los Angeles, CA, USA, June 2018.

[[Slides \(pptx\)](#) ([pdf](#))] [[Lightning Talk Slides \(pptx\)](#) ([pdf](#))]
[[Lightning Talk Video](#)]

The Locality Descriptor:

A Holistic Cross-Layer Abstraction to Express Data Locality in GPUs

Nandita Vijaykumar^{†§} Eiman Ebrahimi[‡] Kevin Hsieh[†]

Phillip B. Gibbons[†] Onur Mutlu^{§†}

[†]Carnegie Mellon University [‡]NVIDIA [§]ETH Zürich

Heterogeneous-Reliability Memory

- Yixin Luo, Sriram Govindan, Bikash Sharma, Mark Santaniello, Justin Meza, Aman Kansal, Jie Liu, Badriddine Khessib, Kushagra Vaid, and Onur Mutlu,
["Characterizing Application Memory Error Vulnerability to Optimize Data Center Cost via Heterogeneous-Reliability Memory"](#)
Proceedings of the 44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Atlanta, GA, June 2014. [Summary] [Slides (pptx) (pdf)] [Coverage on ZDNet]

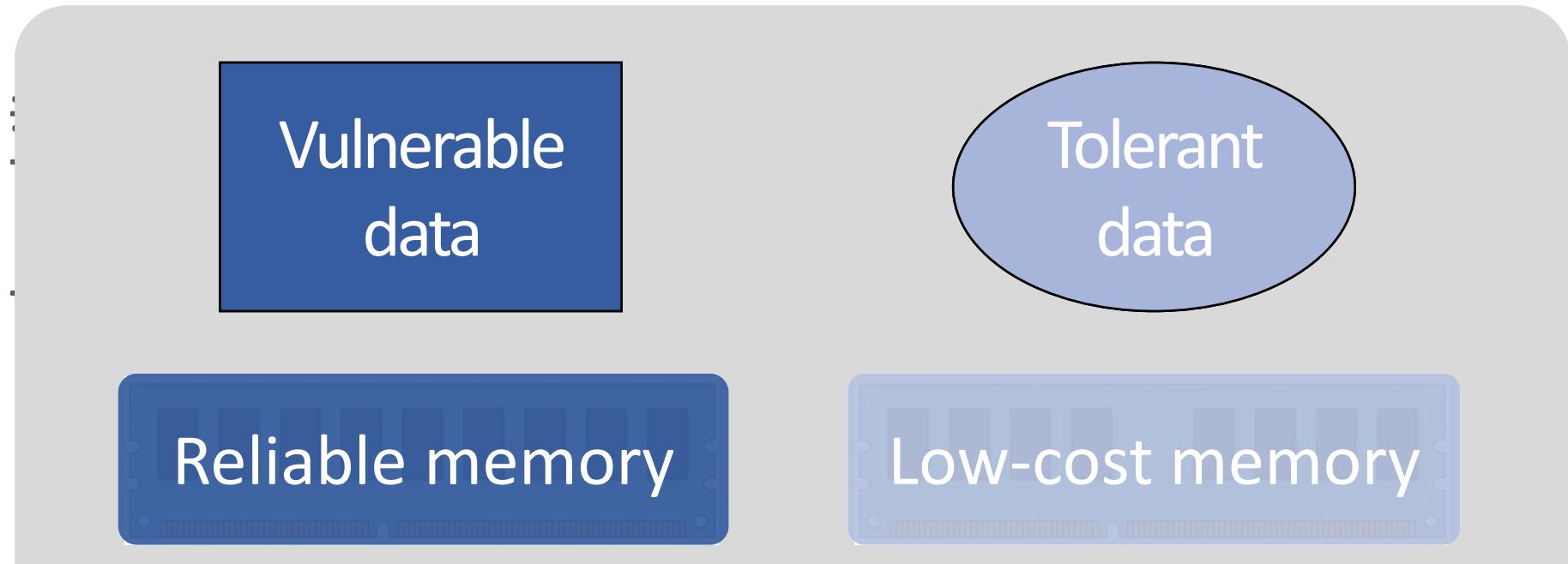
Characterizing Application Memory Error Vulnerability to Optimize Datacenter Cost via Heterogeneous-Reliability Memory

Yixin Luo Sriram Govindan* Bikash Sharma* Mark Santaniello* Justin Meza
Aman Kansal* Jie Liu* Badriddine Khessib* Kushagra Vaid* Onur Mutlu

Carnegie Mellon University, yixinluo@cs.cmu.edu, {meza, onur}@cmu.edu

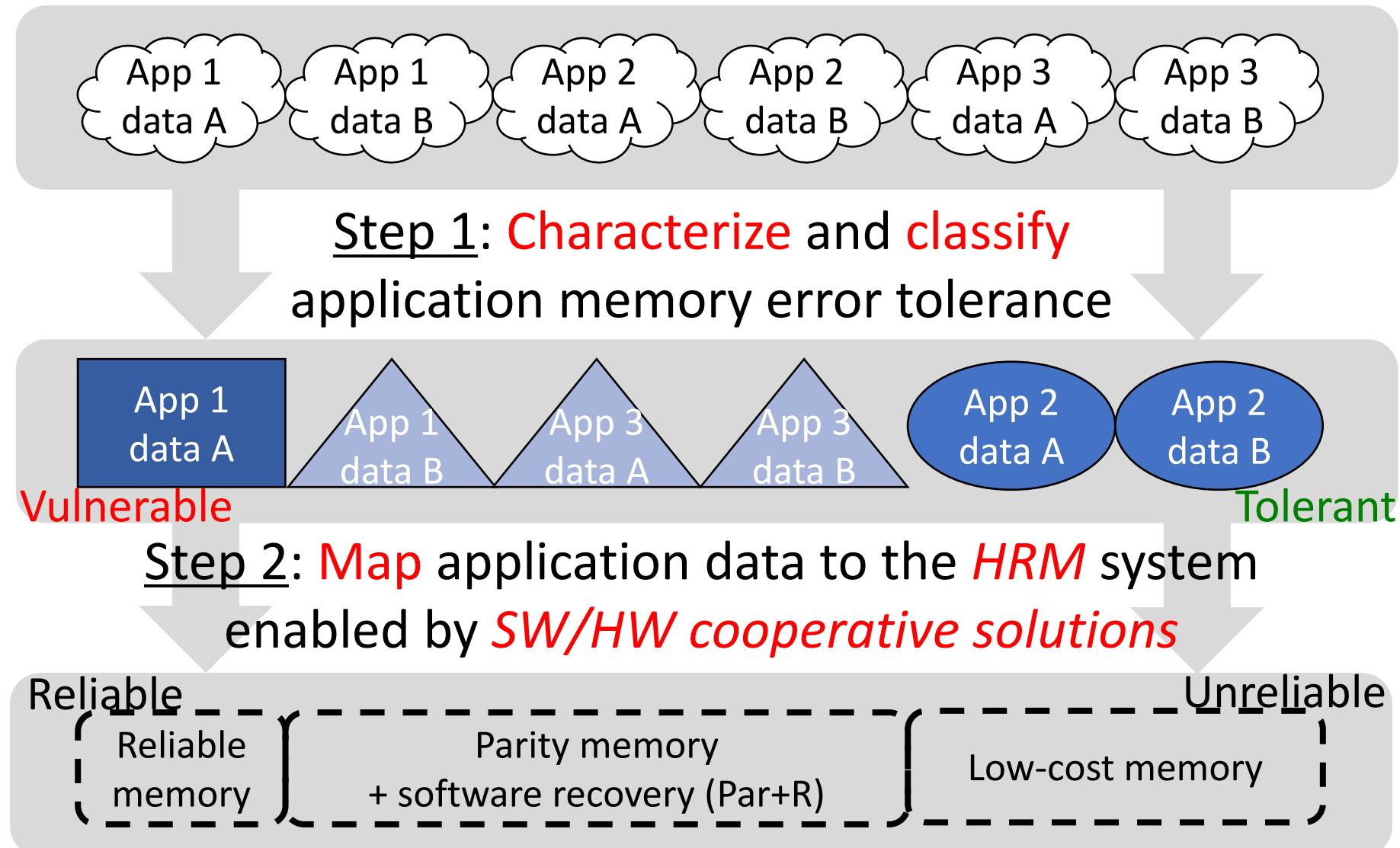
*Microsoft Corporation, {srgovin, bsharma, marksan, kansal, jie.liu, bkhessib, kvaid}@microsoft.com

Exploiting Memory Error Tolerance with Hybrid Memory Systems



On Microsoft's Web Search workload
Reduces server hardware **cost** by **4.7 %**
Achieves single server **availability** target of **99.90 %**
Heterogeneous-Reliability Memory [DSN 2014]

Heterogeneous-Reliability Memory



Rethinking Virtual Memory

- Nastaran Hajinazar, Pratyush Patel, Minesh Patel, Konstantinos Kanellopoulos, Saugata Ghose, Rachata Ausavarungnirun, Geraldo Francisco de Oliveira Jr., Jonathan Appavoo, Vivek Seshadri, and Onur Mutlu,

"The Virtual Block Interface: A Flexible Alternative to the Conventional Virtual Memory Framework"

Proceedings of the 47th International Symposium on Computer Architecture (ISCA), Valencia, Spain, June 2020.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Lightning Talk Slides \(pptx\)](#) ([pdf](#))]

[[ARM Research Summit Poster \(pptx\)](#) ([pdf](#))]

[[Talk Video](#) (26 minutes)]

[[Lightning Talk Video](#) (3 minutes)]

The Virtual Block Interface: A Flexible Alternative to the Conventional Virtual Memory Framework

Nastaran Hajinazar^{*†} Pratyush Patel[✉] Minesh Patel^{*} Konstantinos Kanellopoulos^{*} Saugata Ghose[‡]
Rachata Ausavarungnirun[○] Geraldo F. Oliveira^{*} Jonathan Appavoo[◊] Vivek Seshadri[▽] Onur Mutlu^{*‡}

^{*}*ETH Zürich* [†]*Simon Fraser University* [✉]*University of Washington* [‡]*Carnegie Mellon University*

[○]*King Mongkut's University of Technology North Bangkok* [◊]*Boston University* [▽]*Microsoft Research India*

Many Interesting Things
Are Happening Today
in Computer Architecture

Many Interesting Things
Are Happening Today
in Computer Architecture

**Performance
and
Energy Efficiency**

Intel Optane Persistent Memory (2019)

- Non-volatile main memory
- Based on 3D-XPoint Technology



PCM as Main Memory: Idea in 2009

- Benjamin C. Lee, Engin Ipek, Onur Mutlu, and Doug Burger,
"Architecting Phase Change Memory as a Scalable DRAM Alternative"
Proceedings of the 36th International Symposium on Computer Architecture (ISCA), pages 2-13, Austin, TX, June 2009. [Slides \(pdf\)](#)
One of the 13 computer architecture papers of 2009 selected as Top Picks by IEEE Micro. Selected as a CACM Research Highlight. 2022 Persistent Impact Prize.

Architecting Phase Change Memory as a Scalable DRAM Alternative

Benjamin C. Lee[†] Engin Ipek[†] Onur Mutlu[‡] Doug Burger[†]

[†]Computer Architecture Group
Microsoft Research
Redmond, WA
{blee, ipek, dburger}@microsoft.com

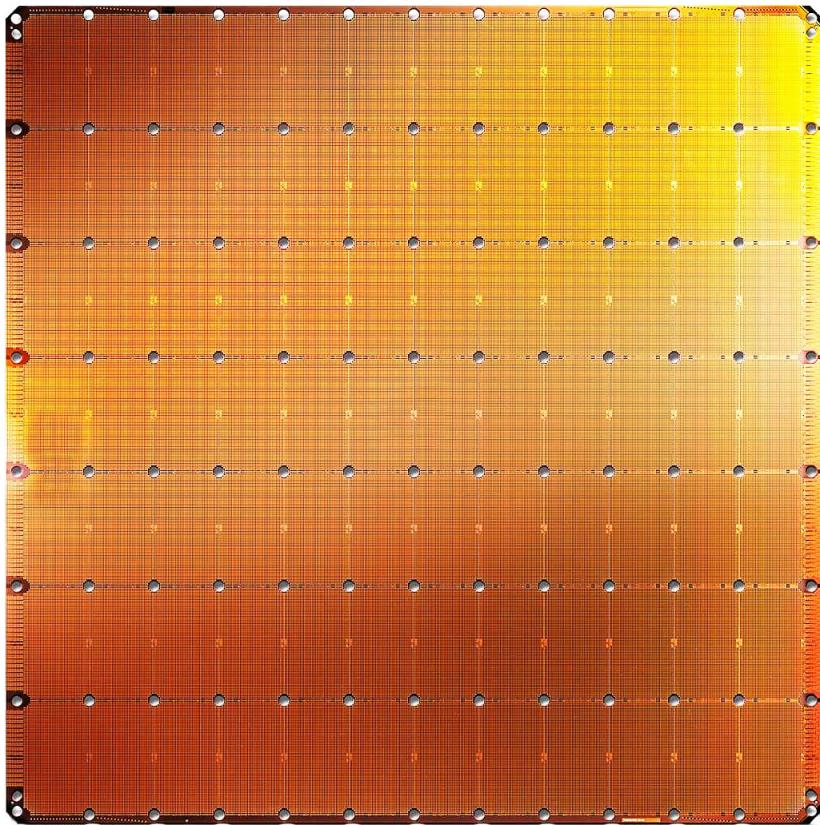
[‡]Computer Architecture Laboratory
Carnegie Mellon University
Pittsburgh, PA
onur@cmu.edu

PCM as Main Memory: Idea in 2009

- Benjamin C. Lee, Ping Zhou, Jun Yang, Youtao Zhang, Bo Zhao, Engin Ipek, Onur Mutlu, and Doug Burger,
"Phase Change Technology and the Future of Main Memory"
IEEE Micro, Special Issue: Micro's Top Picks from 2009 Computer Architecture Conferences (MICRO TOP PICKS), Vol. 30, No. 1, pages 60-70, January/February 2010.

PHASE-CHANGE TECHNOLOGY AND THE FUTURE OF MAIN MEMORY

Cerebras's Wafer Scale Engine (2019)



Cerebras WSE
1.2 Trillion transistors
46,225 mm²

- The largest ML accelerator chip
- 400,000 cores



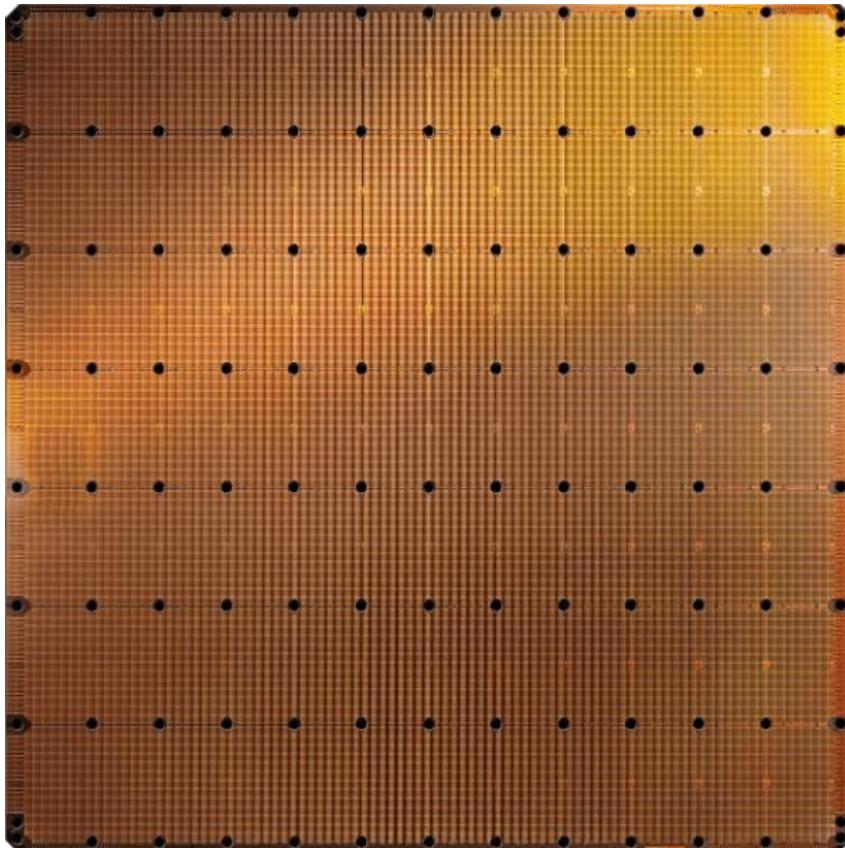
Largest GPU
21.1 Billion transistors
815 mm²

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 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

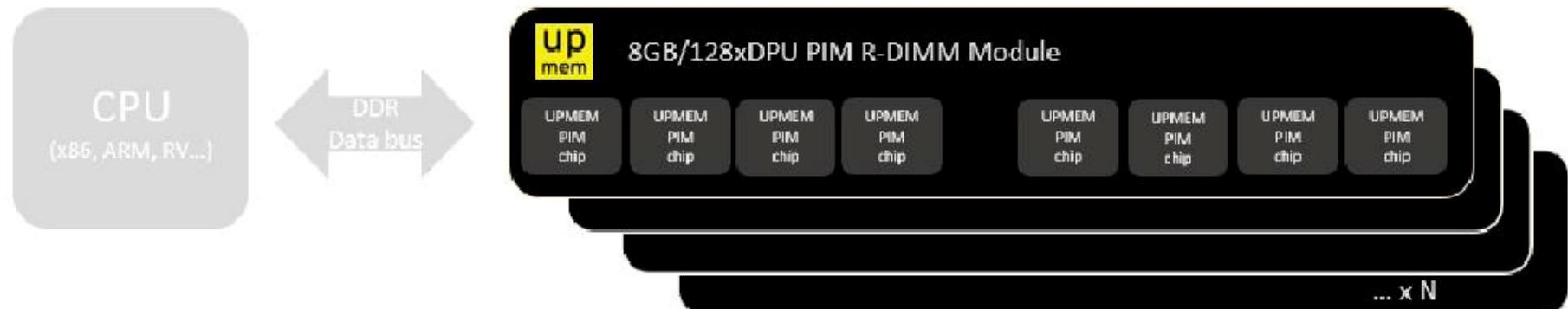
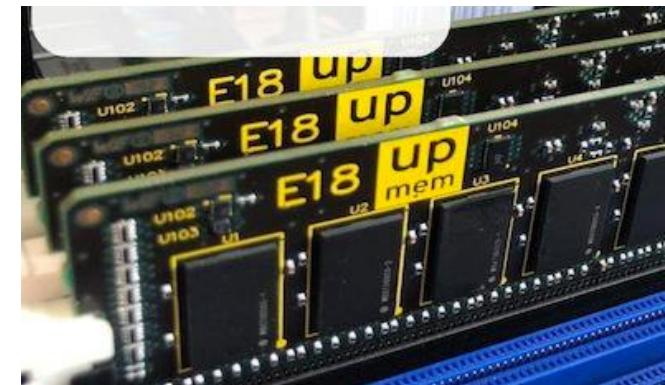
NVIDIA Ampere GA100

<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/>

UPMEM Processing-in-DRAM Engine (2019)

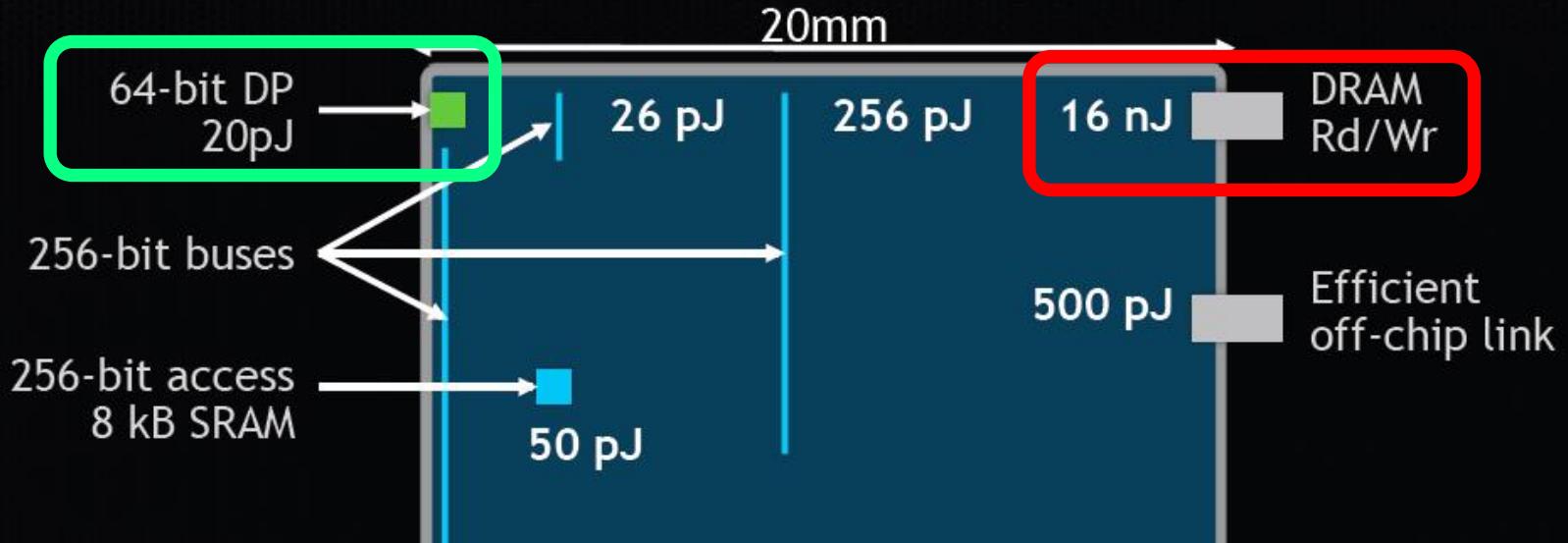
- Processing in DRAM Engine
- Includes **standard DIMM modules**, with a **large number of DPU processors** combined with DRAM chips.
- Replaces **standard** DIMMs
 - DDR4 R-DIMM modules
 - 8GB+128 DPUs (16 PIM chips)
 - Standard 2x-nm DRAM process
 - **Large amounts** of compute & memory bandwidth



Data Movement vs. Computation Energy

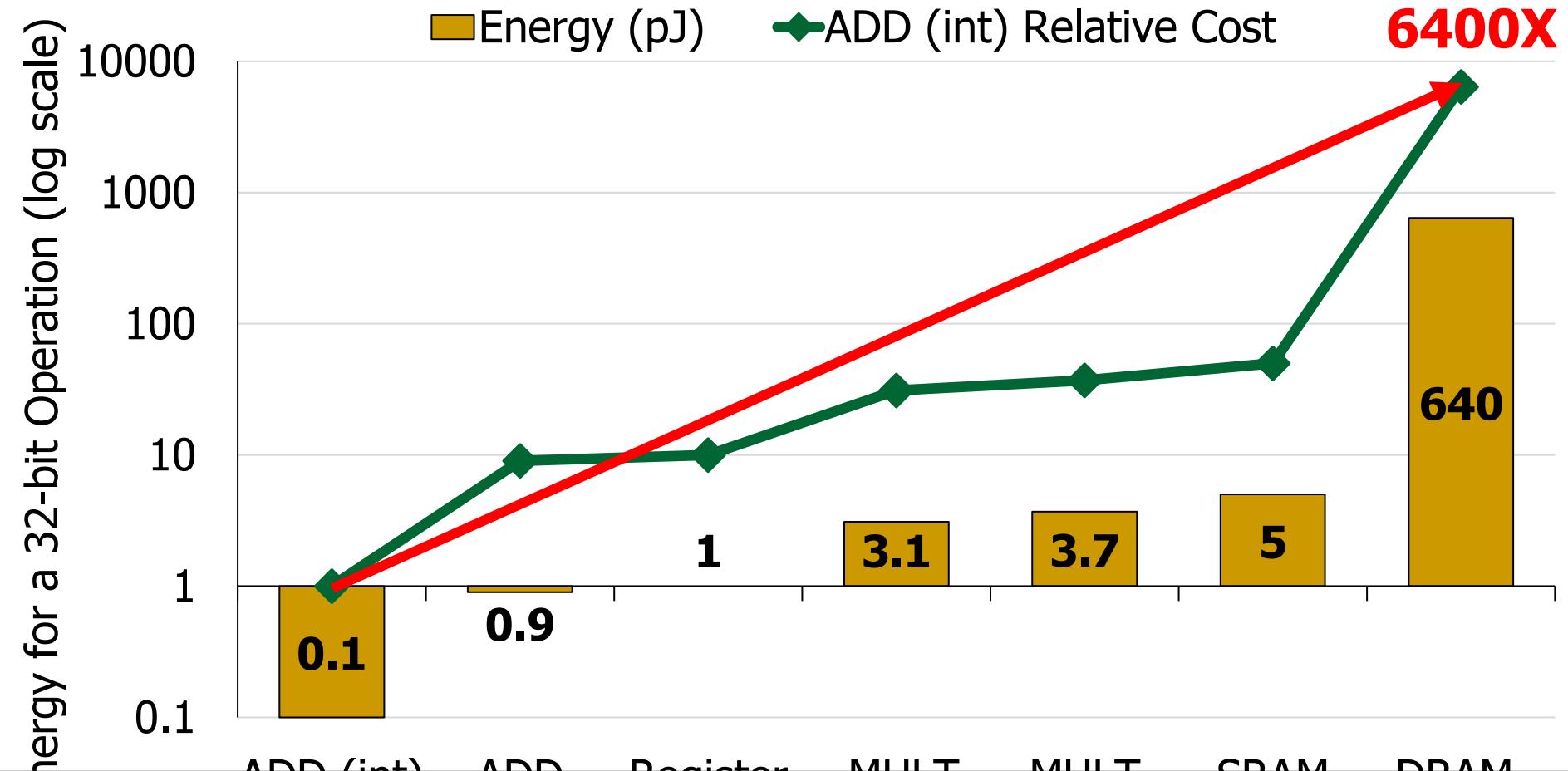
Communication Dominates Arithmetic

Dally, HiPEAC 2015



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

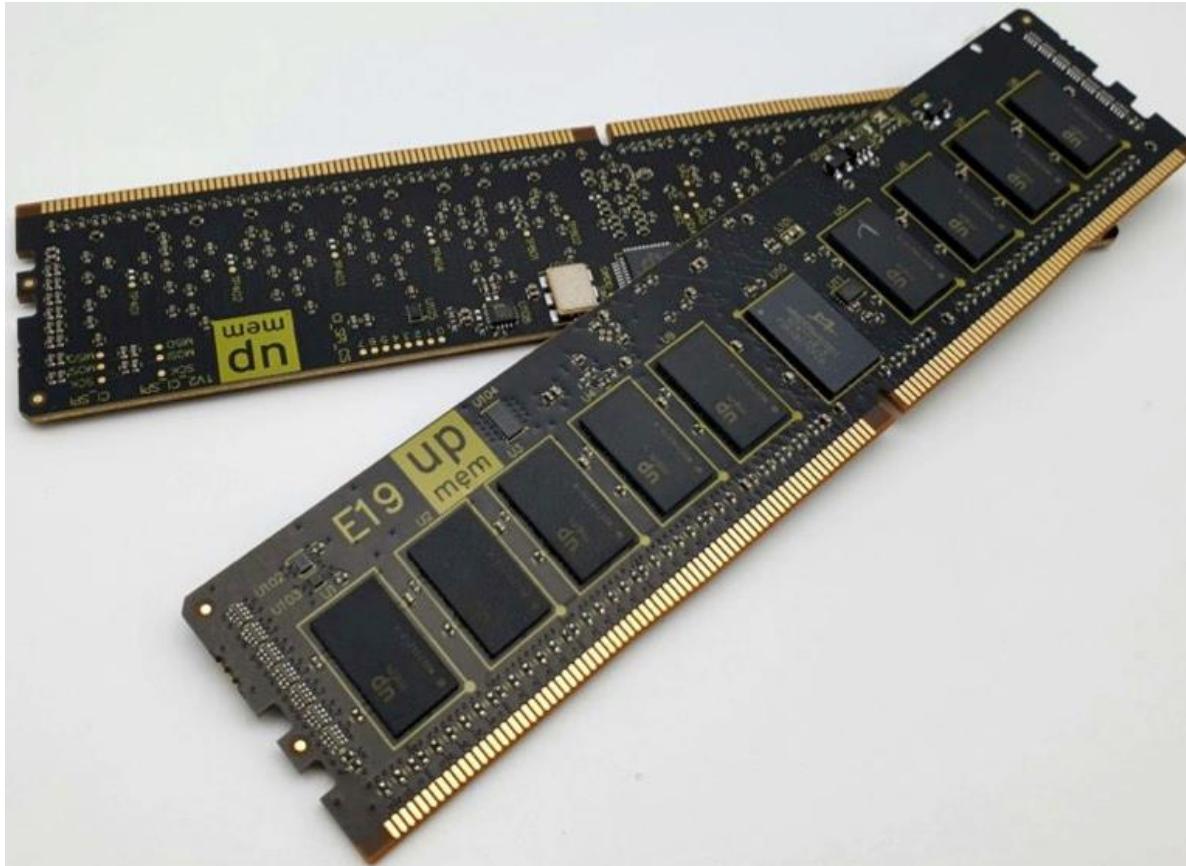
Data Movement vs. Computation Energy



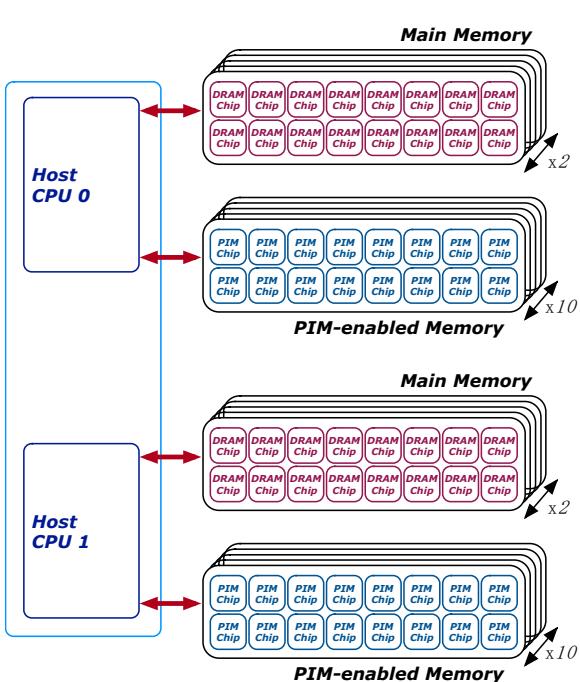
A memory access consumes 6400X
the energy of a simple integer addition

UPMEM Memory Modules

- E19: 8 chips DIMM (1 rank). DPUs @ 267 MHz
- P21: 16 chips DIMM (2 ranks). DPUs @ 350 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 Málaga, Spain

CHRISTINA GIANNOULA, ETH Zürich, Switzerland and NTUA, Greece

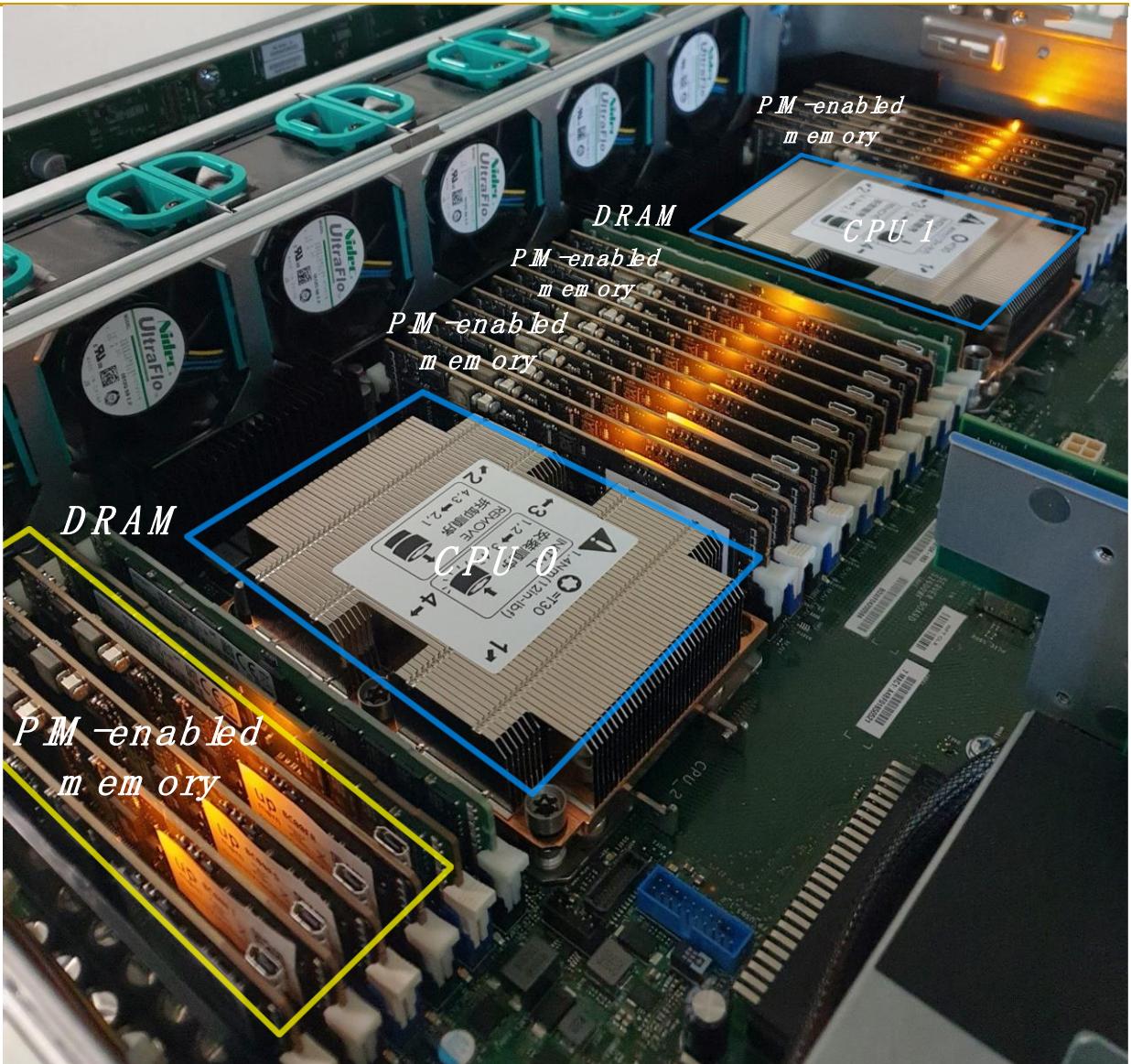
GERALDO F. OLIVEIRA, ETH Zürich, Switzerland

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 computation happens through a slow bus with high latency and limited bandwidth, and the low data reuse in memory-bound workloads is insufficient to amortize the cost of main memory access. Fundamentally addressing this *data movement bottleneck* requires a paradigm where the memory system assumes an active role in computing by integrating processing capabilities. This paradigm is known as *processing-in-memory* (PIM).

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

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



SRC TECHCON Presentation

■ Dr. Juan Gomez-Luna

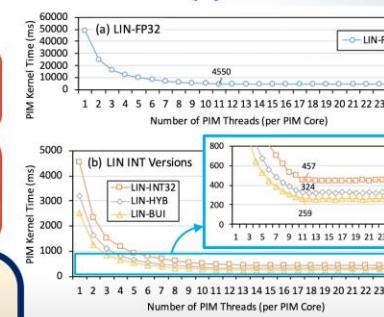
- ❑ Evaluating Machine Learning Workloads on Memory-Centric Computing Systems
- ❑ <https://arxiv.org/pdf/2207.07886.pdf>
- ❑ https://people.inf.ethz.ch/omutlu/pub/MLonUPMEM-PIM_ispass23.pdf

Evaluation: Analysis of PIM Kernels (I)

- Linear regression
 - All versions saturate at 11 or more PIM threads
 - Fixed-point representation accelerates the kernel by an order of magnitude over FP32

Key Takeaway 1. Workloads with arithmetic operations or datatypes not natively supported by PIM cores run at low performance due to instruction emulation (e.g., FP in UPMEM PIM).

Recommendation 1. Use fixed-point representation, without much accuracy loss, if PIM cores do not support FP.



Machine Learning Training on Memory-centric Computing Systems, Juan Gómez-Luna for SRC TECHCON 2023

Onur Mutlu Lectures Subscribed 35.5K subscribers 7.32 / 12:11 • Performance

478 views 12 days ago Livestream - Data-Centric Architectures: Fundamentally Improving Performance and Energy (Spring 2023)
Evaluating Machine Learning Workloads on Memory-centric Computing Systems
Speaker: Dr. Juan Gómez Luna (<https://safari.ethz.ch/juan-gomez-luna/>) ...more

UPMEM PIM System Summary & Analysis

- Juan Gomez-Luna, Izzat El Hajj, Ivan Fernandez, Christina Giannoula, Geraldo F. Oliveira, and Onur Mutlu,
"Benchmarking Memory-Centric Computing Systems: Analysis of Real Processing-in-Memory Hardware"
Invited Paper at Workshop on Computing with Unconventional Technologies (CUT), Virtual, October 2021.
[[arXiv version](#)]
[[PrIM Benchmarks Source Code](#)]
[[Slides \(pptx\) \(pdf\)](#)]
[[Talk Video](#) (37 minutes)]
[[Lightning Talk Video](#) (3 minutes)]

Benchmarking Memory-Centric Computing Systems: Analysis of Real Processing-in-Memory Hardware

Juan Gómez-Luna Izzat El Hajj Ivan Fernandez Christina Giannoula Geraldo F. Oliveira Onur Mutlu
ETH Zürich American University of Beirut University of Malaga National Technical University of Athens ETH Zürich ETH Zürich

PrIM Benchmarks: Application Domains

Domain	Benchmark	Short name
Dense linear algebra	Vector Addition	VA
	Matrix-Vector Multiply	GEMV
Sparse linear algebra	Sparse Matrix-Vector Multiply	SpMV
Databases	Select	SEL
	Unique	UNI
Data analytics	Binary Search	BS
	Time Series Analysis	TS
Graph processing	Breadth-First Search	BFS
Neural networks	Multilayer Perceptron	MLP
Bioinformatics	Needleman-Wunsch	NW
Image processing	Image histogram (short)	HST-S
	Image histogram (large)	HST-L
Parallel primitives	Reduction	RED
	Prefix sum (scan-scan-add)	SCAN-SSA
	Prefix sum (reduce-scan-scan)	SCAN-RSS
	Matrix transposition	TRNS

PrIM Benchmarks are Open Source

- All microbenchmarks, benchmarks, and scripts
- <https://github.com/CMU-SAFARI/prim-benchmarks>

The screenshot shows the GitHub repository page for 'CMU-SAFARI / prim-benchmarks'. The repository has 2 stars, 1 fork, and 1 issue. The 'Code' tab is selected. The README.md file is open, showing its content and statistics (168 lines, 132 sloc, 5.79 KB). The README content discusses PrIM as the first benchmark suite for PIM architecture, its purpose to evaluate, analyze, and characterize the UPMEM PIM architecture, and its use for programming, architecture, and system research. It also mentions baseline CPU and GPU implementations for comparison.

CMU-SAFARI / **prim-benchmarks**

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main prim-benchmarks / README.md Go to file ...

Juan Gomez Luna PrIM -- first commit Latest commit 3de4b49 9 days ago History

1 contributor

168 lines (132 sloc) 5.79 KB Raw Blame

PrIM (Processing-In-Memory Benchmarks)

PrIM is the first benchmark suite for a real-world processing-in-memory (PIM) architecture. PrIM is developed to evaluate, analyze, and characterize the first publicly-available real-world processing-in-memory (PIM) architecture, the [UPMEM](#) 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.

PrIM provides a common set of workloads to evaluate the UPMEM PIM architecture with and can be useful for programming, architecture and system researchers all alike to improve multiple aspects of future PIM hardware and software. The workloads have different characteristics, exhibiting heterogeneity in their memory access patterns, operations and data types, and communication patterns. This repository also contains baseline CPU and GPU implementations of PrIM benchmarks for comparison purposes.

PrIM also includes a set of microbenchmarks can be used to assess various architecture limits such as compute throughput and memory bandwidth.

Understanding a Modern PIM Architecture

Benchmarking a New Paradigm: Experimental Analysis and Characterization of a Real Processing-in-Memory System

JUAN GÓMEZ-LUNA¹, IZZAT EL HAJJ², IVAN FERNANDEZ^{1,3}, CHRISTINA GIANNOULA^{1,4},
GERALDO F. OLIVEIRA¹, AND ONUR MUTLU¹

¹ETH Zürich

²American University of Beirut

³University of Malaga

⁴National Technical University of Athens

Corresponding author: Juan Gómez-Luna (e-mail: juang@ethz.ch).

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

<https://github.com/CMU-SAFARI/prim-benchmarks>

Understanding a Modern PIM Architecture

**Understanding a Modern Processing-in-Memory Architecture:
Benchmarking and Experimental Characterization**

Juan Gómez Luna, Izzat El Hajj,
Ivan Fernandez, Christina Giannoula,
Geraldo F. Oliveira, Onur Mutlu

<https://arxiv.org/pdf/2105.03814.pdf>
<https://github.com/CMU-SAFARI/prim-benchmarks>

ETH Zürich SAFARI zoom

2:26 / 2:57:10

SAFARI Live Seminar: Understanding a Modern Processing-in-Memory Architecture

2,579 views • Streamed live on Jul 12, 2021

93 likes 0 dislikes SHARE SAVE ...



Onur Mutlu Lectures
18.7K subscribers

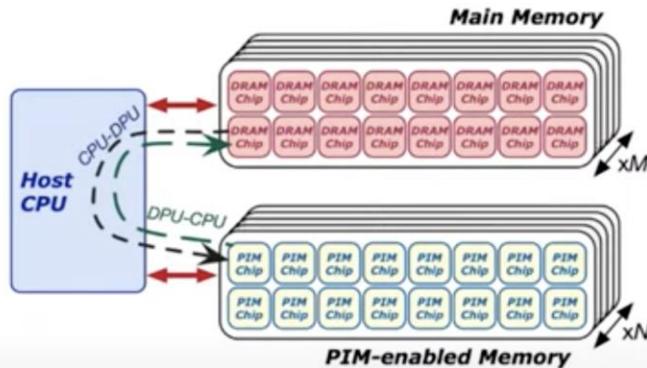
SUBSCRIBED



More on Analysis of the UPMEM PIM Engine

Inter-DPU Communication

- There is no direct communication channel between DPUs



- Inter-DPU communication takes place via the host CPU using CPU-DPU and DPU-CPU transfers
- Example communication patterns:
 - Merging of partial results to obtain the final result
 - Only DPU-CPU transfers
 - Redistribution of intermediate results for further computation
 - DPU-CPU transfers and CPU-DPU transfers



SAFARI Live Seminar: Understanding a Modern Processing-in-Memory Architecture

1,868 views • Streamed live on Jul 12, 2021

81 likes 0 dislikes SHARE SAVE ...



Onur Mutlu Lectures
17.6K subscribers

Talk Title: Understanding a Modern Processing-in-Memory Architecture: Benchmarking and Experimental Characterization
Dr. Juan Gómez-Luna, SAFARI Research Group, D-ITET, ETH Zurich

ANALYTICS EDIT VIDEO

More on Analysis of the UPMEM PIM Engine

Data Movement in Computing Systems

- Data movement dominates performance and is a major system energy bottleneck
- Total system energy: data movement accounts for
 - 62% in consumer applications*,
 - 40% in scientific applications*,
 - 35% in mobile applications*

* Boroumand et al., "Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks," ASPLOS 2018
* Kestor et al., "Quantifying the Energy Cost of Data Movement in Scientific Applications," IISWC 2013
* Pandian and Wu, "Quantifying the energy cost of data movement for emerging smart phone workloads on mobile platforms," IISWC 2014

SAFARI

3

2:27 / 21:28

▶ ▶ 🔍

CC HD

3,482 views • Premiered Jul 25, 2021

38 0 SHARE SAVE ...

Onur Mutlu Lectures 17.9K subscribers

ANALYTICS EDIT VIDEO

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



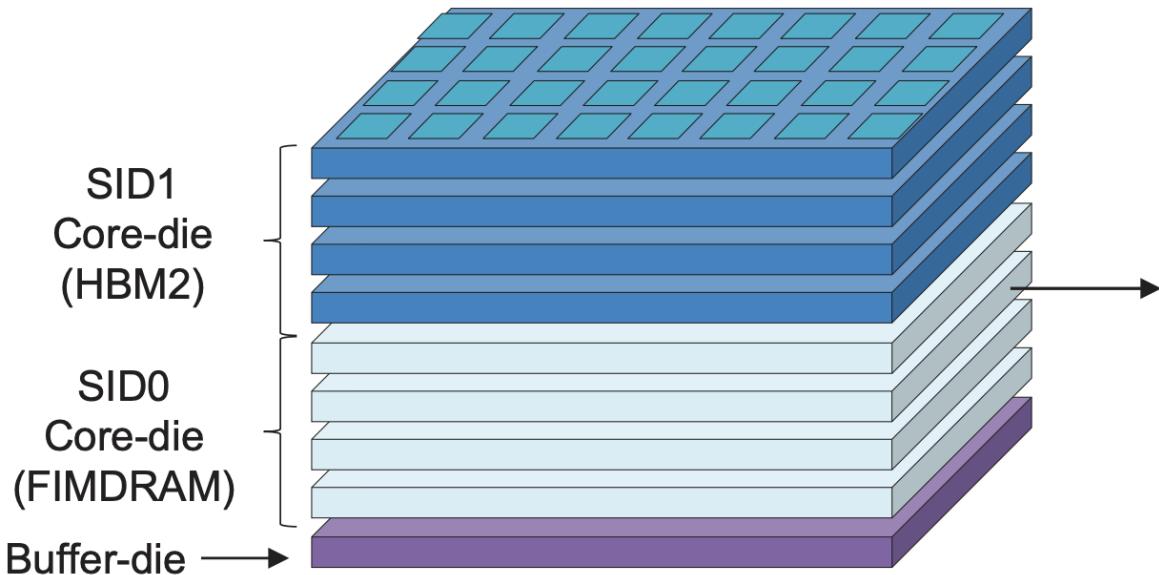
The new architecture will deliver over twice the system performance and reduce energy consumption by more than 70%

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 Phua¹, 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

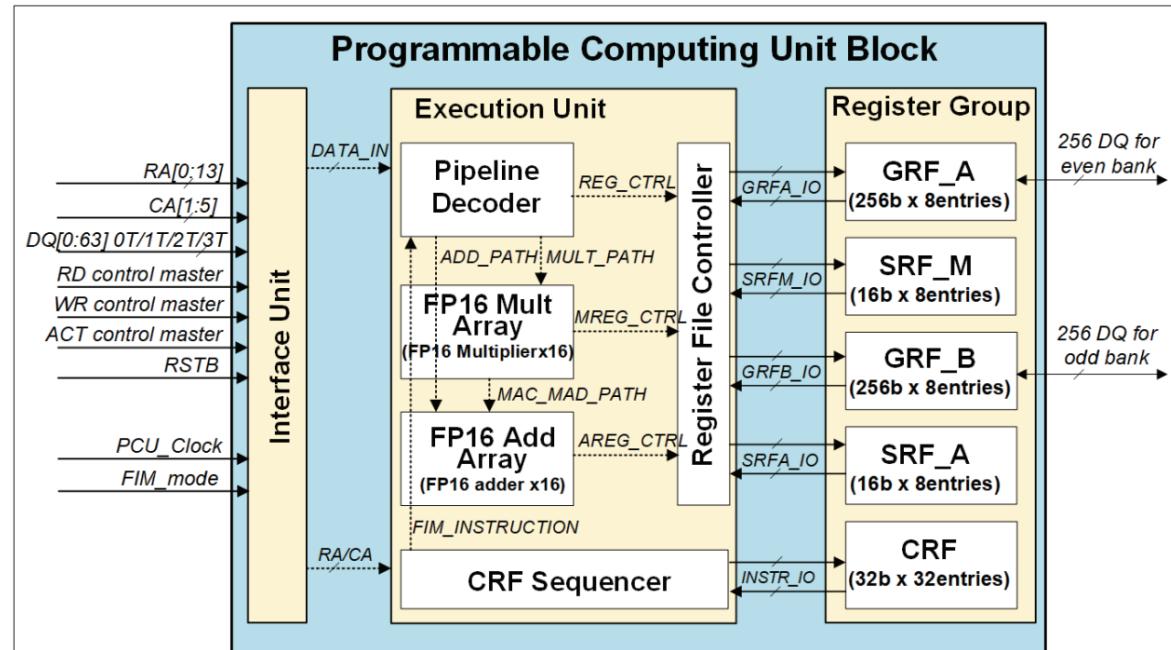
²Samsung Electronics, San Jose, CA

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, Seungil Oh¹, Hak-Soo Yu¹, Haesuk Lee¹,
Soo Young Kim¹, Youngmin Cho¹, Jin Guk Kim¹, Jongyoon Choi¹,
Hyun-Sung Shin¹, Jin Kim¹, BengSeng Phua², HyoungMin Kim¹,
Myeong Jun Song¹, Ahn Choi¹, Daeho Kim¹, SooYoung Kim¹, Eun-Bong Kim¹,
David Wang¹, Shinhwa Kang¹, Yuhwan Ro¹, Seungwoo Seo¹, JoonHo Song¹,
Jayoun 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)

[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, Seungil Oh¹, Hak-Soo Yu¹, Haesuk Lee¹,
Soo Young Kim¹, Youngmin Cho¹, Jin Guk Kim¹, Jongyoon Choi¹,
Hyun-Sung Shin¹, Jin Kim¹, BengSeng Phua², HyoungMin Kim¹,
Myeong Jun Song¹, Ahn Choi¹, Daeho Kim¹, SooYoung Kim¹, Eun-Bong Kim¹,
David Wang¹, Shinhaeng Kang¹, Yuhwan Ro¹, Seungwoo Seo¹, JoonHo Song¹,
Jayoun Youn¹, Kyomin Sohn¹, Nam Sung Kim¹

¹Samsung Electronics, Hwaseong, Korea

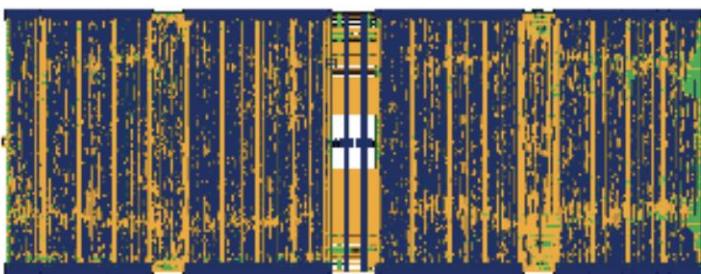
²Samsung Electronics, San Jose, CA

³Samsung Electronics, Suwon, Korea

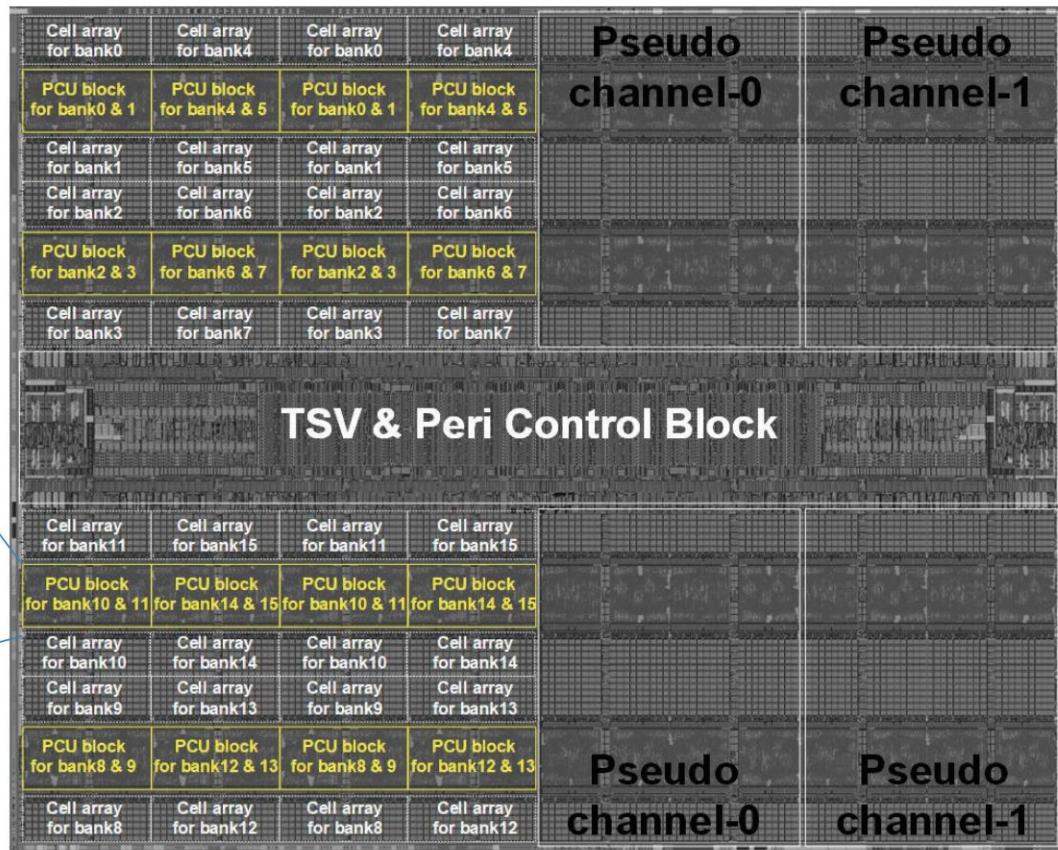
Samsung Function-in-Memory DRAM (2021)

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

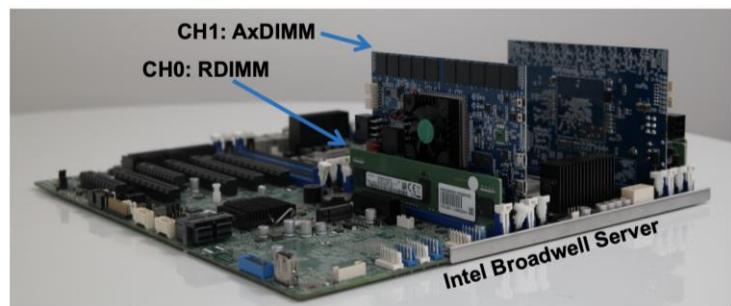
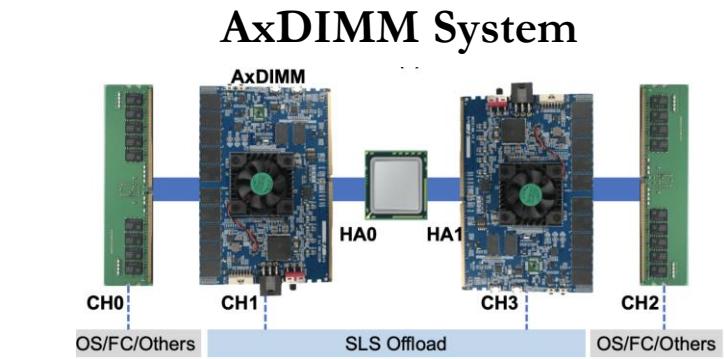
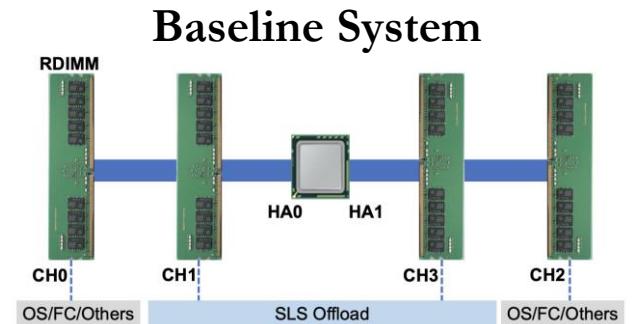
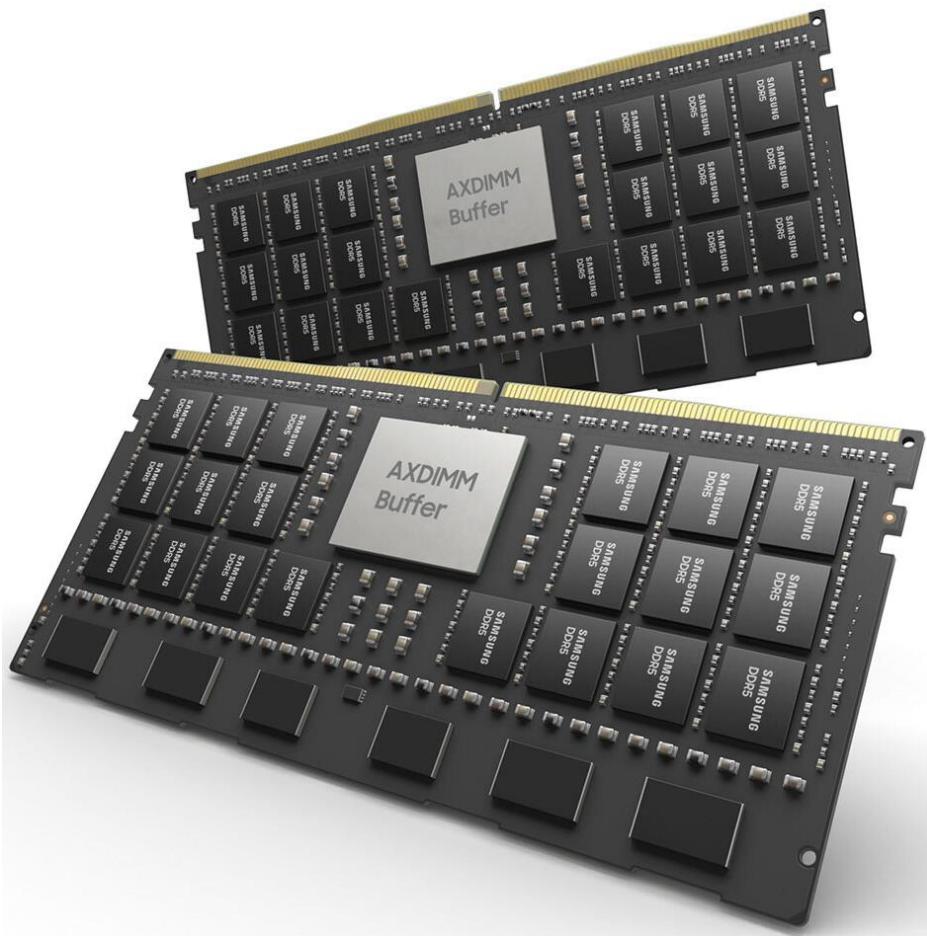
25.4 A 20nm 6GB Function-In-Memory DRAM, Based on HBM2 with a 1.2TFLOPS Programmable Computing Unit Using Bank-Level Parallelism, for Machine Learning Applications

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, Jongyoon Choi¹, Hyun-Sung Shin¹, Jin Kim¹, BengSeon Phua², HyoungMin Kim¹, Myeong Jun Song¹, Ahn Choi¹, Daeho Kim¹, SooYoung Kim¹, Eun-Bong Kim¹, David Wang¹, Shinhwang Kang¹, Yuhwan Ro¹, Seungwoo Seo¹, JoonHo Song¹, Jaeyoun Youn¹, Kyomin Sohn¹, Nam Sung Kim¹

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

Samsung AxDIMM (2021)

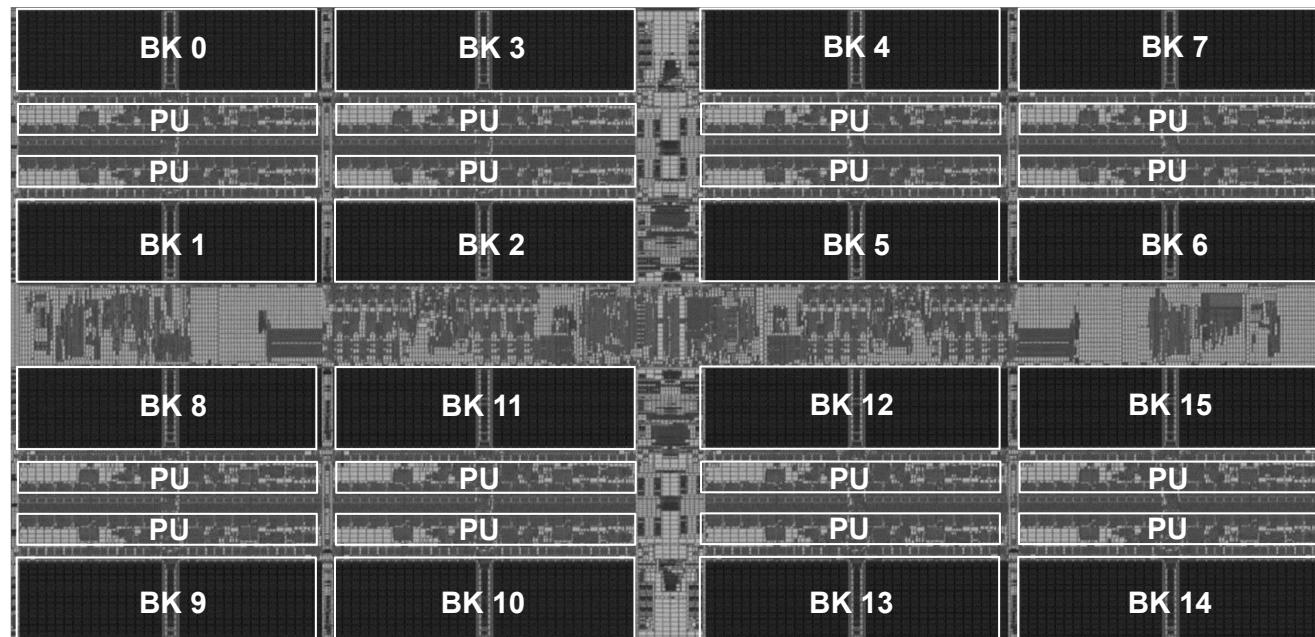
- DIMM-based PIM
 - DLRM recommendation system



SK Hynix AiM: Chip Implementation (2022)

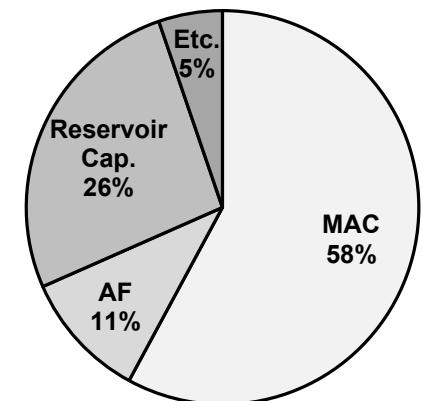
- 4 Gb AiM die with 16 processing units (PUs)

AiM Die Photograph



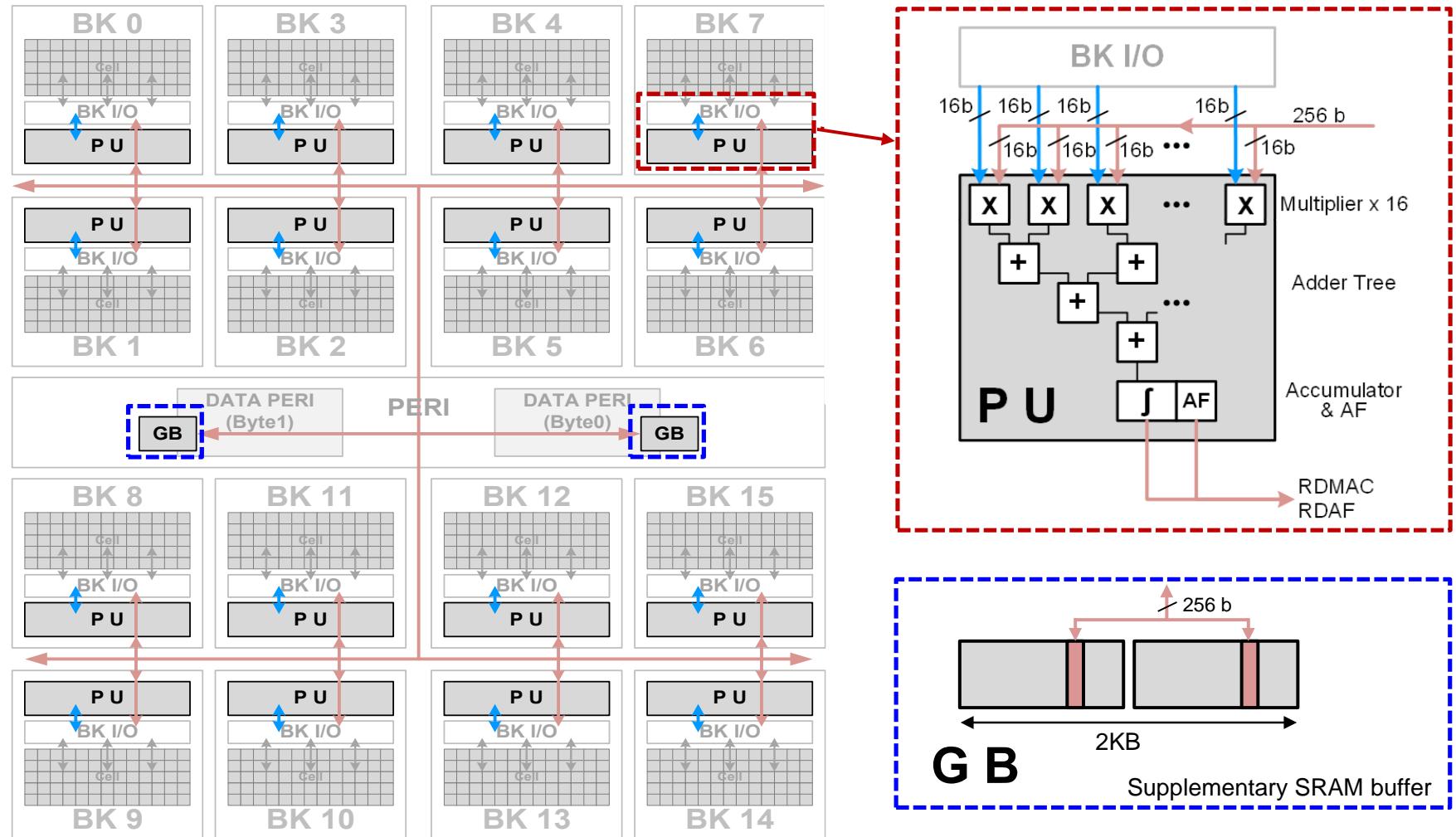
1 Process Unit (PU) Area

Total	0.19mm ²
MAC	0.11mm ²
Activation Function (AF)	0.02mm ²
Reservoir Cap.	0.05mm ²
Etc.	0.01mm ²



SK Hynix AiM: System Organization (2022)

GDDR6-based AiM architecture



AliBaba PIM Recommendation System (2022)

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

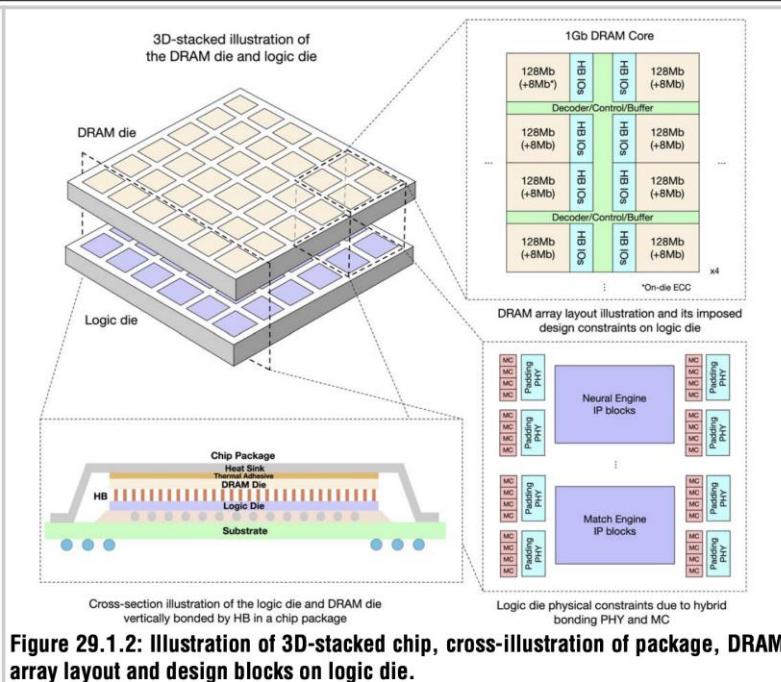
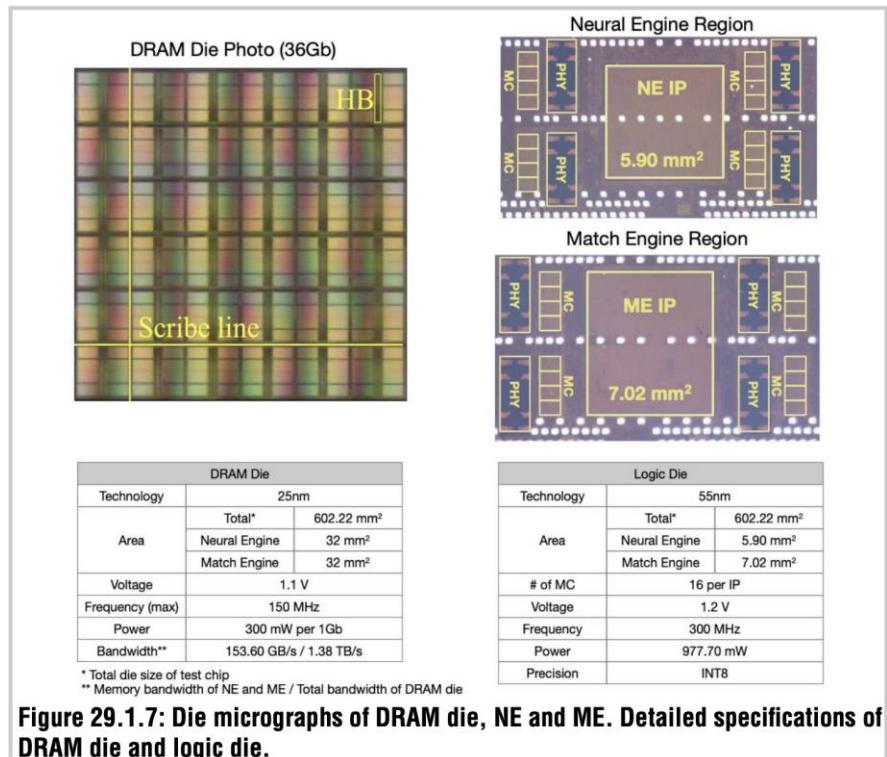


Figure 29.1.2: Illustration of 3D-stacked chip, cross-illustration of package, DRAM array layout and design blocks on logic die.



* Total die size of test chip

** Memory bandwidth of NE and ME / Total bandwidth of DRAM die

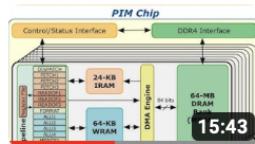
Figure 29.1.7: Die micrographs of DRAM die, NE and ME. Detailed specifications of DRAM die and logic die.

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

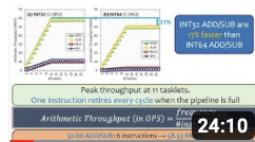
Lectures about Real PIM Systems

- https://www.youtube.com/playlist?list=PL5Q2soXY2Zi_EObuoAZVSq_o6UySWQHvZ



PIM Course: Lecture 3: Real-world PIM: UPMEM PIM (Spring 2023)

Onur Mutlu Lectures • 411 views • 2 months ago



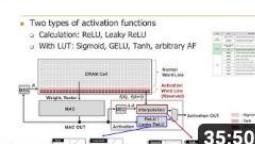
PIM Course: Lecture 4: Real-world PIM: Microbenchmarking of UPMEM PIM (Spring 2023)

Onur Mutlu Lectures • 188 views • 2 months ago



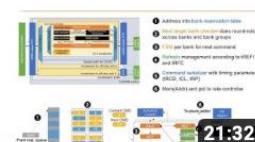
PIM Course: Lecture 5: Real-world PIM: Samsung HBM-PIM (Spring 2023)

Onur Mutlu Lectures • 483 views • 2 months ago



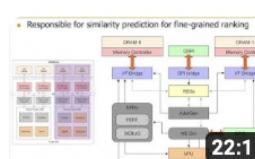
PIM Course: Lecture 6: Real-world PIM: SK Hynix AiM (Spring 2023)

Onur Mutlu Lectures • 569 views • 1 month ago



PIM Course: Lecture 7: Real-world PIM: Samsung AxDIMM (Spring 2023)

Onur Mutlu Lectures • 325 views • 1 month ago



PIM Course: Lecture 8: Real-world PIM: Alibaba HB-PNM (Spring 2023)

Onur Mutlu Lectures • 277 views • 1 month ago

Processing in Memory: Two Approaches

1. Processing near Memory
2. Processing using Memory

Computer Architecture

Lecture 1: Introduction and Basics

Prof. Onur Mutlu

ETH Zürich

Fall 2023

28 September 2023

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

Specialized Processing in Memory (2015)

- Junwhan Ahn, Sungpack Hong, Sungjoo Yoo, Onur Mutlu, and Kiyoung Choi,

"A Scalable Processing-in-Memory Accelerator for Parallel Graph Processing"

Proceedings of the 42nd International Symposium on Computer Architecture (ISCA), Portland, OR, June 2015.
[Slides (pdf)] [Lightning Session Slides (pdf)]

A Scalable Processing-in-Memory Accelerator for Parallel Graph Processing

Junwhan Ahn Sungpack Hong[§] Sungjoo Yoo Onur Mutlu[†] Kiyoung Choi

junwhan@snu.ac.kr, sungpack.hong@oracle.com, sungjoo.yoo@gmail.com, onur@cmu.edu, kchoi@snu.ac.kr

Seoul National University

[§]Oracle Labs

[†]Carnegie Mellon University

Simple Processing in Memory (2015)

- Junwhan Ahn, Sungjoo Yoo, Onur Mutlu, and Kiyoung Choi,
"PIM-Enabled Instructions: A Low-Overhead, Locality-Aware Processing-in-Memory Architecture"

Proceedings of the 42nd International Symposium on Computer Architecture (ISCA), Portland, OR, June 2015.
[Slides (pdf)] [Lightning Session Slides (pdf)]

PIM-Enabled Instructions: A Low-Overhead, Locality-Aware Processing-in-Memory Architecture

Junwhan Ahn Sungjoo Yoo Onur Mutlu[†] Kiyoung Choi

junwhan@snu.ac.kr, sungjoo.yoo@gmail.com, onur@cmu.edu, kchoi@snu.ac.kr

Seoul National University

[†]Carnegie Mellon University

Processing in Memory on Mobile Devices

- 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.

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}

Efficient Synchronization for NDP

- Christina Giannoula, Nandita Vijaykumar, Nikela Papadopoulou, Vasileios Karakostas, Ivan Fernandez, Juan Gómez-Luna, Lois Orosa, Nectarios Koziris, Georgios Goumas, and Onur Mutlu,
"SynCron: Efficient Synchronization Support for Near-Data-Processing Architectures"

Proceedings of the 27th International Symposium on High-Performance Computer Architecture (HPCA), Virtual, February-March 2021.

SynCron: Efficient Synchronization Support for Near-Data-Processing Architectures

Christina Giannoula^{†‡} Nandita Vijaykumar^{*‡} Nikela Papadopoulou[†] Vasileios Karakostas[†] Ivan Fernandez^{§‡}
Juan Gómez-Luna[‡] Lois Orosa[‡] Nectarios Koziris[†] Georgios Goumas[†] Onur Mutlu[‡]

[†]*National Technical University of Athens* [‡]*ETH Zürich* ^{*}*University of Toronto* [§]*University of Malaga*

Accelerating GPU Execution with PIM (I)

- Kevin Hsieh, Eiman Ebrahimi, Gwangsun Kim, Niladrish Chatterjee, Mike O'Connor, Nandita Vijaykumar, Onur Mutlu, and Stephen W. Keckler,

"Transparent Offloading and Mapping (TOM): Enabling Programmer-Transparent Near-Data Processing in GPU Systems"

Proceedings of the 43rd International Symposium on Computer Architecture (ISCA), Seoul, South Korea, June 2016.

[Slides (pptx) (pdf)]

[Lightning Session Slides (pptx) (pdf)]

Transparent Offloading and Mapping (TOM):

Enabling Programmer-Transparent Near-Data Processing in GPU Systems

Kevin Hsieh[‡] Eiman Ebrahimi[†] Gwangsun Kim^{*} Niladrish Chatterjee[†] Mike O'Connor[†]
Nandita Vijaykumar[‡] Onur Mutlu^{§‡} Stephen W. Keckler[†]

[‡]Carnegie Mellon University [†]NVIDIA ^{*}KAIST [§]ETH Zürich

Accelerating Linked Data Structures

- Kevin Hsieh, Samira Khan, Nandita Vijaykumar, Kevin K. Chang, Amirali Boroumand, Saugata Ghose, and Onur Mutlu,

"Accelerating Pointer Chasing in 3D-Stacked Memory: Challenges, Mechanisms, Evaluation"

Proceedings of the 34th IEEE International Conference on Computer Design (ICCD), Phoenix, AZ, USA, October 2016.

Accelerating Pointer Chasing in 3D-Stacked Memory: Challenges, Mechanisms, Evaluation

Kevin Hsieh[†] Samira Khan[‡] Nandita Vijaykumar[†]

Kevin K. Chang[†] Amirali Boroumand[†] Saugata Ghose[†] Onur Mutlu^{§†}

[†]*Carnegie Mellon University* [‡]*University of Virginia* [§]*ETH Zürich*

DAMOV Analysis Methodology & Workloads

DAMOV: A New Methodology and Benchmark Suite for Evaluating Data Movement Bottlenecks

GERALDO F. OLIVEIRA, ETH Zürich, Switzerland

JUAN GÓMEZ-LUNA, ETH Zürich, Switzerland

LOIS OROSA, ETH Zürich, Switzerland

SAUGATA GHOSE, University of Illinois at Urbana–Champaign, USA

NANDITA VIJAYKUMAR, University of Toronto, Canada

IVAN FERNANDEZ, University of Malaga, Spain & ETH Zürich, Switzerland

MOHAMMAD SADROSADATI, Institute for Research in Fundamental Sciences (IPM), Iran & ETH Zürich, Switzerland

ONUR MUTLU, ETH Zürich, Switzerland

Data movement between the CPU and main memory is a first-order obstacle against improving performance, scalability, and energy efficiency in modern systems. Computer systems employ a range of techniques to reduce overheads tied to data movement, spanning from traditional mechanisms (e.g., deep multi-level cache hierarchies, aggressive hardware prefetchers) to emerging techniques such as Near-Data Processing (NDP), where some computation is moved close to memory. Prior NDP works investigate the root causes of data movement bottlenecks using different profiling methodologies and tools. However, there is still a lack of understanding about the key metrics that can identify different data movement bottlenecks and their relation to traditional and emerging data movement mitigation mechanisms. Our goal is to methodically identify potential sources of data movement over a broad set of applications and to comprehensively compare traditional compute-centric data movement mitigation techniques (e.g., caching and prefetching) to more memory-centric techniques (e.g., NDP), thereby developing a rigorous understanding of the best techniques to mitigate each source of data movement.

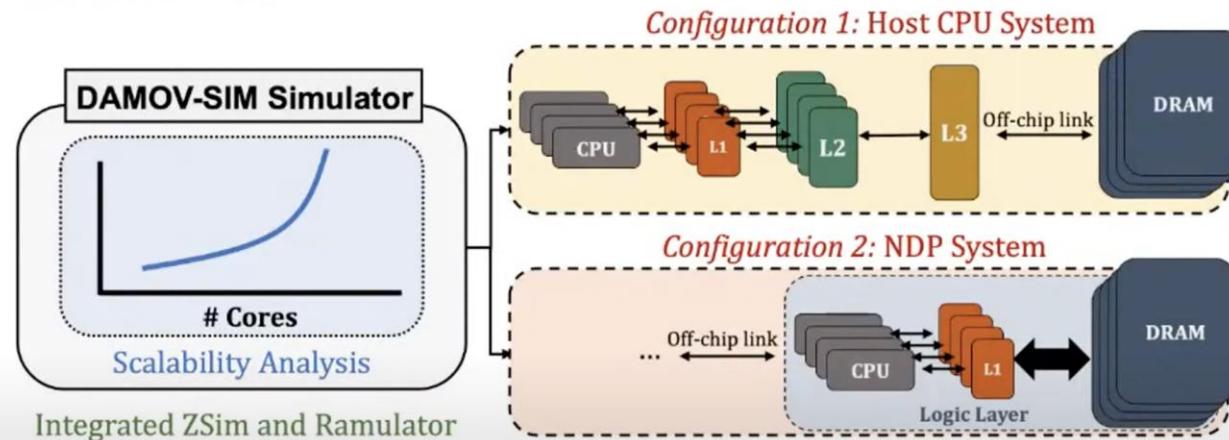
With this goal in mind, we perform the first large-scale characterization of a wide variety of applications, across a wide range of application domains, to identify fundamental program properties that lead to data movement to/from main memory. We develop the first systematic methodology to classify applications based on the sources contributing to data movement bottlenecks. From our large-scale characterization of 77K functions across 345 applications, we select 144 functions to form the first open-source benchmark suite (DAMOV) for main memory data movement studies. We select a diverse range of functions that (1) represent different types of data movement bottlenecks, and (2) come from a wide range of application domains. Using NDP as a case study, we identify new insights about the different data movement bottlenecks and use these insights to determine the most suitable data movement mitigation mechanism for a particular application. We open-source DAMOV and the complete source code for our new characterization methodology at <https://github.com/CMU-SAFARI/DAMOV>.

<https://arxiv.org/pdf/2105.03725.pdf>

More on DAMOV Analysis Methodology & Workloads

Step 3: Memory Bottleneck Classification (2/2)

- **Goal:** identify the specific sources of data movement bottlenecks



- **Scalability Analysis:**
 - 1, 4, 16, 64, and 256 out-of-order/in-order host and NDP CPU cores
 - 3D-stacked memory as main memory



DAMOV-SIM: <https://github.com/CMU-SAFARI/DAMOV>



SAFARI Live Seminar: DAMOV: A New Methodology & Benchmark Suite for Data Movement Bottlenecks

352 views • Streamed live on Jul 22, 2021

18 likes 0 dislikes SHARE SAVE ...



Onur Mutlu Lectures
17.7K subscribers

ANALYTICS

EDIT VIDEO

DAMOV is Open-Source

- We open-source our benchmark suite and our toolchain

CMU-SAFARI/DAMOV

Code Issues Pull requests Actions Projects Security Insights Settings

main 1 branch 0 tags Go to file Add file Code

omutlu Update README.md ce1b4ea 17 days ago 5 commits

simulator Cleaning 19 days ago

README.md Update README.md 17 days ago

get_workloads.sh DAMOV -- first commit 19 days ago

About

DAMOV is a benchmark suite and a methodical framework targeting the study of data movement bottlenecks in modern applications. It is intended to study new architectures, such as near-data processing. Described by Oliveira et al. (preliminary version at <https://arxiv.org/pdf/2105.03725.pdf>)

DAMOV-SIM

DAMOV Benchmarks

README.md

DAMOV: A New Methodology and Benchmark Suite for Evaluating Data Movement Bottlenecks

DAMOV is a benchmark suite and a methodical framework targeting the study of data movement bottlenecks in modern applications. It is intended to study new architectures, such as near-data processing.

The DAMOV benchmark suite is the first open-source benchmark suite for main memory data movement-related studies, based on our systematic characterization methodology. This suite consists of 144 functions representing different sources of data movement bottlenecks and can be used as a baseline benchmark set for future data-movement mitigation research. The applications in the DAMOV benchmark suite belong to popular benchmark suites, including [BWA](#), [Chai](#), [Darknet](#), [GASE](#), [Hardware Effects](#), [Hashjoin](#), [HPCC](#), [HPCG](#), [Ligra](#), [PARSEC](#), [Parboil](#), [PolyBench](#), [Phoenix](#), [Rodinia](#), [SPLASH-2](#), [STREAM](#).

Readme

Releases

No releases published Create a new release

Packages

No packages published Publish your first package

Languages



More on DAMOV Methods & Benchmarks

- Geraldo F. Oliveira, Juan Gomez-Luna, Lois Orosa, Saugata Ghose, Nandita Vijaykumar, Ivan Fernandez, Mohammad Sadrosadati, and Onur Mutlu,
"DAMOV: A New Methodology and Benchmark Suite for Evaluating Data Movement Bottlenecks"
IEEE Access, 8 September 2021.
Preprint in arXiv, 8 May 2021.
[[arXiv preprint](#)]
[[IEEE Access version](#)]
[[DAMOV Suite and Simulator Source Code](#)]
[[SAFARI Live Seminar Video](#) (2 hrs 40 mins)]
[[Short Talk Video](#) (21 minutes)]

DAMOV: A New Methodology and Benchmark Suite for Evaluating Data Movement Bottlenecks

GERALDO F. OLIVEIRA, ETH Zürich, Switzerland

JUAN GÓMEZ-LUNA, ETH Zürich, Switzerland

LOIS OROSA, ETH Zürich, Switzerland

SAUGATA GHOSE, University of Illinois at Urbana–Champaign, USA

NANDITA VIJAYKUMAR, University of Toronto, Canada

IVAN FERNANDEZ, University of Malaga, Spain & ETH Zürich, Switzerland

MOHAMMAD SADROSADATI, ETH Zürich, Switzerland

ONUR MUTLU, ETH Zürich, Switzerland

Processing in Memory: Two Approaches

1. Processing near Memory
2. Processing using Memory

In-DRAM Processing (2013)

- Vivek Seshadri et al., "**Ambit: In-Memory Accelerator for Bulk Bitwise Operations Using Commodity DRAM Technology**," MICRO 2017.

Ambit: In-Memory Accelerator for Bulk Bitwise Operations Using Commodity DRAM Technology

Vivek Seshadri^{1,5} Donghyuk Lee^{2,5} Thomas Mullins^{3,5} Hasan Hassan⁴ Amirali Boroumand⁵
Jeremie Kim^{4,5} Michael A. Kozuch³ Onur Mutlu^{4,5} Phillip B. Gibbons⁵ Todd C. Mowry⁵

¹Microsoft Research India ²NVIDIA Research ³Intel ⁴ETH Zürich ⁵Carnegie Mellon University

In-DRAM Bulk Bitwise Execution (2017)

- Vivek Seshadri and Onur Mutlu,
"In-DRAM Bulk Bitwise Execution Engine"
Invited Book Chapter in Advances in Computers, to appear
in 2020.
[Preliminary arXiv version]

In-DRAM Bulk Bitwise Execution Engine

Vivek Seshadri
Microsoft Research India
visesha@microsoft.com

Onur Mutlu
ETH Zürich
onur.mutlu@inf.ethz.ch

SIMDRAM Framework (2021)

- Nastaran Hajinazar, Geraldo F. Oliveira, Sven Gregorio, Joao Dinis Ferreira, Nika Mansouri Ghiasi, Minesh Patel, Mohammed Alser, Saugata Ghose, Juan Gomez-Luna, and Onur Mutlu,
"SIMDRAM: An End-to-End Framework for Bit-Serial SIMD Computing in DRAM"

Proceedings of the 26th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), Virtual, March-April 2021.

[[2-page Extended Abstract](#)]

[[Short Talk Slides \(pptx\)](#) ([pdf](#))]

[[Talk Slides \(pptx\)](#) ([pdf](#))]

[[Short Talk Video \(5 mins\)](#)]

[[Full Talk Video \(27 mins\)](#)]

SIMDRAM: A Framework for Bit-Serial SIMD Processing using DRAM

*Nastaran Hajinazar^{1,2}

Nika Mansouri Ghiasi¹

*Geraldo F. Oliveira¹

Minesh Patel¹

Sven Gregorio¹

Mohammed Alser¹

João Dinis Ferreira¹

Saugata Ghose³

Juan Gómez-Luna¹

Onur Mutlu¹

¹ETH Zürich

²Simon Fraser University

³University of Illinois at Urbana–Champaign

Bulk Data Copy and Initialization in DRAM

- Vivek Seshadri, Yoongu Kim, Chris Fallin, Donghyuk Lee, Rachata Ausavarungnirun, Gennady Pekhimenko, Yixin Luo, Onur Mutlu, Michael A. Kozuch, Phillip B. Gibbons, and Todd C. Mowry,

"RowClone: Fast and Energy-Efficient In-DRAM Bulk Data Copy and Initialization"

Proceedings of the 46th International Symposium on Microarchitecture (MICRO), Davis, CA, December 2013. [[Slides \(pptx\)](#) [\(pdf\)](#)] [[Lightning Session Slides \(pptx\)](#) [\(pdf\)](#)] [[Poster \(pptx\)](#) [\(pdf\)](#)]

RowClone: Fast and Energy-Efficient In-DRAM Bulk Data Copy and Initialization

Vivek Seshadri Yoongu Kim Chris Fallin* Donghyuk Lee
vseshadr@cs.cmu.edu yoongukim@cmu.edu cfallin@c1f.net donghyuk1@cmu.edu

Rachata Ausavarungnirun Gennady Pekhimenko Yixin Luo
rachata@cmu.edu gpekhime@cs.cmu.edu yixinluo@andrew.cmu.edu

Onur Mutlu Phillip B. Gibbons[†] Michael A. Kozuch[†] Todd C. Mowry
onur@cmu.edu phillip.b.gibbons@intel.com michael.a.kozuch@intel.com tcm@cs.cmu.edu

LISA: Increasing Connectivity in DRAM

- Kevin K. Chang, Prashant J. Nair, Saugata Ghose, Donghyuk Lee, Moinuddin K. Qureshi, and Onur Mutlu,

"Low-Cost Inter-Linked Subarrays (LISA): Enabling Fast Inter-Subarray Data Movement in DRAM"

Proceedings of the 22nd International Symposium on High-Performance Computer Architecture (HPCA), Barcelona, Spain, March 2016.

[Slides (pptx) (pdf)]

[Source Code]

Low-Cost Inter-Linked Subarrays (LISA): Enabling Fast Inter-Subarray Data Movement in DRAM

Kevin K. Chang[†], Prashant J. Nair^{*}, Donghyuk Lee[†], Saugata Ghose[†], Moinuddin K. Qureshi^{*}, and Onur Mutlu[†]

[†]*Carnegie Mellon University* ^{*}*Georgia Institute of Technology*

FIGARO: Fine-Grained In-DRAM Copy

- Yaohua Wang, Lois Orosa, Xiangjun Peng, Yang Guo, Saugata Ghose, Minesh Patel, Jeremie S. Kim, Juan Gómez Luna, Mohammad Sadrosadati, Nika Mansouri Ghiasi, and Onur Mutlu,
"FIGARO: Improving System Performance via Fine-Grained In-DRAM Data Relocation and Caching"

Proceedings of the 53rd International Symposium on Microarchitecture (MICRO), Virtual, October 2020.

FIGARO: Improving System Performance via Fine-Grained In-DRAM Data Relocation and Caching

Yaohua Wang^{*} Lois Orosa[†] Xiangjun Peng^{○*} Yang Guo^{*} Saugata Ghose^{◊‡} Minesh Patel[†]
Jeremie S. Kim[†] Juan Gómez Luna[†] Mohammad Sadrosadati[§] Nika Mansouri Ghiasi[†] Onur Mutlu^{†‡}

^{*}*National University of Defense Technology* [†]*ETH Zürich* [○]*Chinese University of Hong Kong*

[◊]*University of Illinois at Urbana-Champaign* [‡]*Carnegie Mellon University* [§]*Institute of Research in Fundamental Sciences*

Network-On-Memory: Fast Inter-Bank Copy

- Seyyed Hossein SeyyedAghaei Rezaei, Mehdi Modarressi, Rachata Ausavarungnirun, Mohammad Sadrosadati, Onur Mutlu, and Masoud Daneshtalab,

"NoM: Network-on-Memory for Inter-Bank Data Transfer in Highly-Banked Memories"

IEEE Computer Architecture Letters (CAL), to appear in 2020.

NoM: NETWORK-ON-MEMORY FOR INTER-BANK DATA TRANSFER IN HIGHLY-BANKED MEMORIES

Seyyed Hossein SeyyedAghaei Rezaei¹
Mohammad Sadrosadati³

Mehdi Modarressi^{1,3}
Onur Mutlu⁴

Rachata Ausavarungnirun²
Masoud Daneshtalab⁵

¹University of Tehran

²King Mongkut's University of Technology North Bangkok

³Institute for Research in Fundamental Sciences

⁴ETH Zürich

⁵Mälardalens University

In-DRAM Physical Unclonable Functions

- Jeremie S. Kim, Minesh Patel, Hasan Hassan, and Onur Mutlu,
"The DRAM Latency PUF: Quickly Evaluating Physical Unclonable Functions by Exploiting the Latency-Reliability Tradeoff in Modern DRAM Devices"
Proceedings of the 24th International Symposium on High-Performance Computer Architecture (HPCA), Vienna, Austria, February 2018.
[[Lightning Talk Video](#)] [[Slides \(pptx\) \(pdf\)](#)] [[Lightning Session Slides \(pptx\) \(pdf\)](#)]
[[Full Talk Lecture Video](#) (28 minutes)]

The DRAM Latency PUF: Quickly Evaluating Physical Unclonable Functions by Exploiting the Latency-Reliability Tradeoff in Modern Commodity DRAM Devices

Jeremie S. Kim^{†§} Minesh Patel[§] Hasan Hassan[§] Onur Mutlu^{§†}
†Carnegie Mellon University §ETH Zürich

In-DRAM True Random Number Generation

- Jeremie S. Kim, Minesh Patel, Hasan Hassan, Lois Orosa, and Onur Mutlu,
"D-RaNGe: Using Commodity DRAM Devices to Generate True Random Numbers with Low Latency and High Throughput"

Proceedings of the 25th International Symposium on High-Performance Computer Architecture (HPCA), Washington, DC, USA, February 2019.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Full Talk Video](#) (21 minutes)]

[[Full Talk Lecture Video](#) (27 minutes)]

Top Picks Honorable Mention by IEEE Micro.

D-RaNGe: Using Commodity DRAM Devices to Generate True Random Numbers with Low Latency and High Throughput

Jeremie S. Kim^{†§}

Minesh Patel[§]

Hasan Hassan[§]

Lois Orosa[§]

Onur Mutlu^{§‡}

[†]Carnegie Mellon University

[§]ETH Zürich

In-DRAM True Random Number Generation

- Ataberk Olgun, Minesh Patel, A. Giray Yaglikci, Haocong Luo, Jeremie S. Kim, F. Nisa Bostanci, Nandita Vijaykumar, Oguz Ergin, and Onur Mutlu,

"QUAC-TRNG: High-Throughput True Random Number Generation Using Quadruple Row Activation in Commodity DRAM Chips"

Proceedings of the 48th International Symposium on Computer Architecture (ISCA),
Virtual, June 2021.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Short Talk Slides \(pptx\)](#) ([pdf](#))]

[[Talk Video](#) (25 minutes)]

[[SAFARI Live Seminar Video](#) (1 hr 26 mins)]

QUAC-TRNG: High-Throughput True Random Number Generation Using Quadruple Row Activation in Commodity DRAM Chips

Ataberk Olgun^{§†}

Minesh Patel[§]

A. Giray Yağlıkçı[§]

Haocong Luo[§]

Jeremie S. Kim[§]

F. Nisa Bostancı^{§†}

Nandita Vijaykumar^{§○}

Oğuz Ergin[†]

Onur Mutlu[§]

[§]*ETH Zürich*

[†]*TOBB University of Economics and Technology*

[○]*University of Toronto*

In-DRAM True Random Number Generation

- F. Nisa Bostancı, Ataberk Olgun, Lois Orosa, A. Giray Yaglikci, Jeremie S. Kim, Hasan Hassan, Oguz Ergin, and Onur Mutlu,

"DR-STRaNGe: End-to-End System Design for DRAM-based True Random Number Generators"

Proceedings of the 28th International Symposium on High-Performance Computer Architecture (HPCA), Virtual, April 2022.

[Slides (pptx) (pdf)]

[Short Talk Slides (pptx) (pdf)]

DR-STRaNGe: End-to-End System Design for DRAM-based True Random Number Generators

F. Nisa Bostancı^{†§}
Jeremie S. Kim[§]

Ataberk Olgun^{†§}
Hasan Hassan[§]

Lois Orosa[§]
Oğuz Ergin[†]

A. Giray Yağlıkçı[§]
Onur Mutlu[§]

[†]*TOBB University of Economics and Technology*

[§]*ETH Zürich*

In-DRAM Lookup-Table Based Execution

- João Dinis Ferreira, Gabriel Falcao, Juan Gómez-Luna, Mohammed Alser, Lois Orosa, Mohammad Sadrosadati, Jeremie S. Kim, Geraldo F. Oliveira, Taha Shahroodi, Anant Nori, and Onur Mutlu,
"pLUTO: Enabling Massively Parallel Computation in DRAM via Lookup Tables"
Proceedings of the 55th International Symposium on Microarchitecture (MICRO), Chicago, IL, USA, October 2022.
[Slides (pptx) (pdf)]
[Longer Lecture Slides (pptx) (pdf)]
[Lecture Video (26 minutes)]
[arXiv version]
[Source Code (Officially Artifact Evaluated with All Badges)]

Officially artifact evaluated as available, reusable and reproducible.



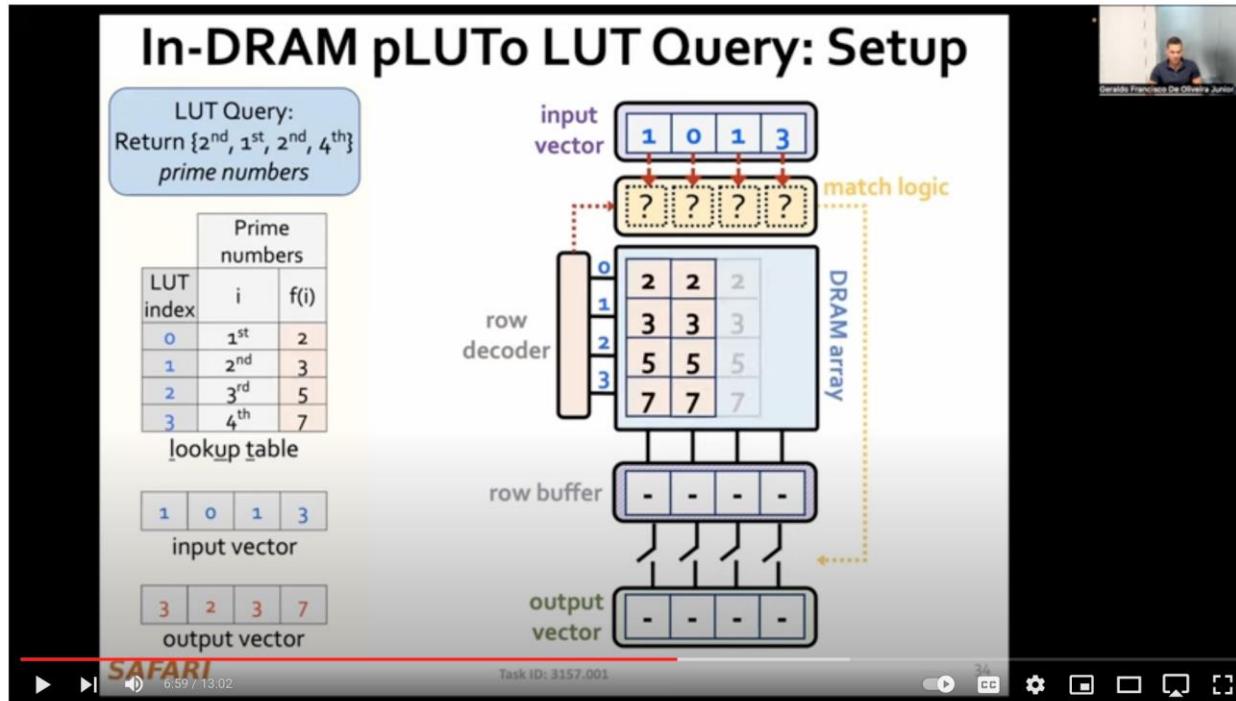
pLUTO: Enabling Massively Parallel Computation in DRAM via Lookup Tables

João Dinis Ferreira[§] Gabriel Falcao[†] Juan Gómez-Luna[§] Mohammed Alser[§]
Lois Orosa[§] ∇ Mohammad Sadrosadati[§] Jeremie S. Kim[§] Geraldo F. Oliveira[§]
Taha Shahroodi[‡] Anant Nori^{*} Onur Mutlu[§]

[§]*ETH Zürich* [†]*IT, University of Coimbra* ∇ *Galicia Supercomputing Center* [‡]*TU Delft* ^{*}*Intel*

SRC TECHCON Presentation

- Geraldo F. Oliveira
 - pLUTO: Enabling Massively Parallel Computation in DRAM via Lookup Tables
 - <https://arxiv.org/pdf/2104.07699.pdf>



pLUTO: Enabling Massively Parallel Computation in DRAM via Lookup Tables, SRC TECHCON 2023



Onur Mutlu Lectures
35.5K subscribers

Subscribed

17

Clip

Save

...

321 views 9 days ago

pLUTO: Enabling Massively Parallel Computation in DRAM via Lookup Tables

Speaker: Geraldo F. Oliveira ...more

In-Flash Bulk Bitwise Execution

- Jisung Park, Roknoddin Azizi, Geraldo F. Oliveira, Mohammad Sadrosadati, Rakesh Nadig, David Novo, Juan Gómez-Luna, Myungsuk Kim, and Onur Mutlu,
"Flash-Cosmos: In-Flash Bulk Bitwise Operations Using Inherent Computation Capability of NAND Flash Memory"

Proceedings of the 55th International Symposium on Microarchitecture (MICRO), Chicago, IL, USA, October 2022.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Longer Lecture Slides \(pptx\)](#) ([pdf](#))]

[[Lecture Video](#) (44 minutes)]

[[arXiv version](#)]

Flash-Cosmos: In-Flash Bulk Bitwise Operations Using Inherent Computation Capability of NAND Flash Memory

Jisung Park^{§∇} Roknoddin Azizi[§] Geraldo F. Oliveira[§] Mohammad Sadrosadati[§]
Rakesh Nadig[§] David Novo[†] Juan Gómez-Luna[§] Myungsuk Kim[‡] Onur Mutlu[§]

[§]*ETH Zürich*

[∇]*POSTECH*

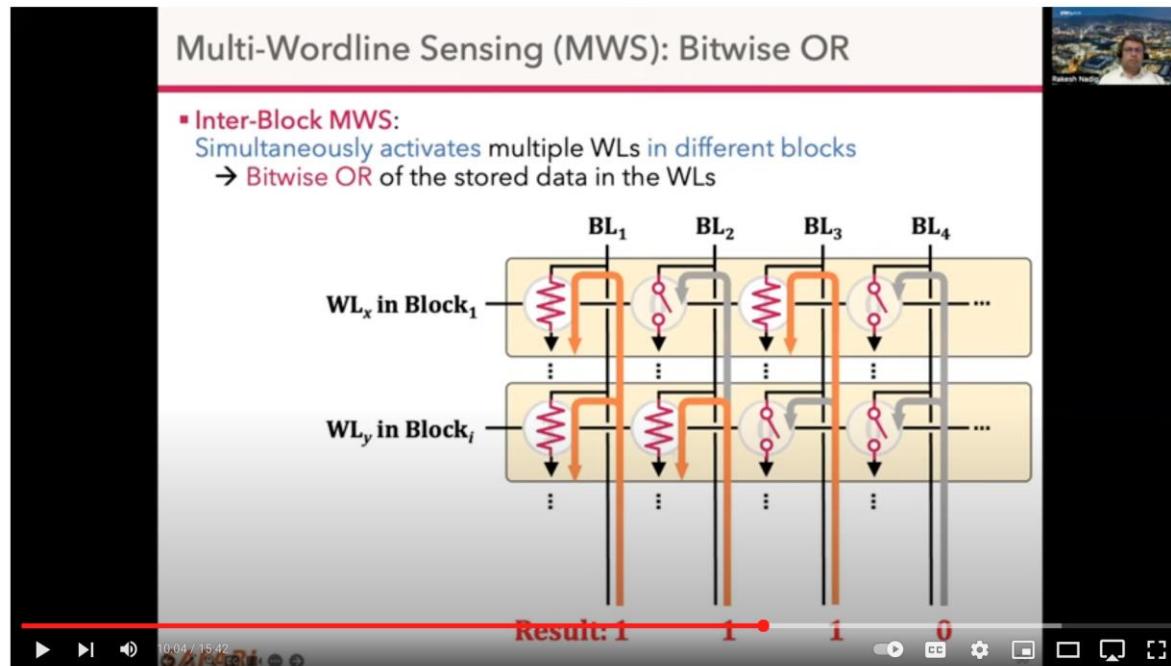
[†]*LIRMM, Univ. Montpellier, CNRS*

[‡]*Kyungpook National University*

SRC TECHCON Presentation

■ Rakesh Nadig

- ❑ Flash-Cosmos: In-Flash Bulk Bitwise Operations Using Inherent Computation Capability of NAND Flash Memory
- ❑ <https://arxiv.org/pdf/2209.05566.pdf>



Flash-Cosmos: In-Flash Bulk Bitwise Operations, SRC TECHCON 2023



Onur Mutlu Lectures

Subscribed

12 Share Clip Save ...

143 views 8 days ago

Flash-Cosmos: In-Flash Bulk Bitwise Operations Using Inherent Computation Capability of NAND Flash Memory

Speaker: Rakesh Nadig ...more

Processing in Memory: Two Approaches

1. Processing near Memory
2. Processing using Memory

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 2023

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

^aETH Zürich

^bCarnegie Mellon University

^cUniversity of Illinois at Urbana-Champaign

^dKing Mongkut's University of Technology North Bangkok

Abstract

Modern computing systems are overwhelmingly designed to move data to computation. This design choice goes directly against at least three key trends in computing that cause performance, scalability and energy bottlenecks: (1) data access is a key bottleneck as many important applications are increasingly data-intensive, and memory bandwidth and energy do not scale well, (2) energy consumption is a key limiter in almost all computing platforms, especially server and mobile systems, (3) data movement, especially off-chip to on-chip, is very expensive in terms of bandwidth, energy and latency, much more so than computation. These trends are especially severely-felt in the data-intensive server and energy-constrained mobile systems of today.

At the same time, conventional memory technology is facing many technology scaling challenges in terms of reliability, energy, and performance. As a result, memory system architects are open to organizing memory in different ways and making it more intelligent, at the expense of higher cost. The emergence of 3D-stacked memory plus logic, the adoption of error correcting codes inside the latest DRAM chips, proliferation of different main memory standards and chips, specialized for different purposes (e.g., graphics, low-power, high bandwidth, low latency), and the necessity of designing new solutions to serious reliability and security issues, such as the RowHammer phenomenon, are an evidence of this trend.

This chapter discusses recent research that aims to practically enable computation close to data, an approach we call *processing-in-memory* (PIM). PIM places computation mechanisms in or near where the data is stored (i.e., inside the memory chips, in the logic layer of 3D-stacked memory, or in the memory controllers), so that data movement between the computation units and memory is reduced or eliminated. While the general idea of PIM is not new, we discuss motivating trends in applications as well as memory circuits/technology that greatly exacerbate the need for enabling it in modern computing systems. We examine at least two promising new approaches to designing PIM systems to accelerate important data-intensive applications: (1) *processing using memory* by exploiting analog operational properties of DRAM chips to perform massively-parallel operations in memory, with low-cost changes, (2) *processing near memory* by exploiting 3D-stacked memory technology design to provide high memory bandwidth and low memory latency to in-memory logic. In both approaches, we describe and tackle relevant cross-layer research, design, and adoption challenges in devices, architecture, systems, and programming models. Our focus is on the development of in-memory processing designs that can be adopted in real computing platforms at low cost. We conclude by discussing work on solving key challenges to the practical adoption of PIM.

Keywords: memory systems, data movement, main memory, processing-in-memory, near-data processing, computation-in-memory, processing using memory, processing near memory, 3D-stacked memory, non-volatile memory, energy efficiency, high-performance computing, computer architecture, computing paradigm, emerging technologies, memory scaling, technology scaling, dependable systems, robust systems, hardware security, system security, latency, low-latency computing

Contents

1	Introduction	2
2	Major Trends Affecting Main Memory	4
3	The Need for Intelligent Memory Controllers to Enhance Memory Scaling	6
4	Perils of Processor-Centric Design	9
5	Processing-in-Memory (PIM): Technology Enablers and Two Approaches	11
5.1	New Technology Enablers: 3D-Stacked Memory and Non-Volatile Memory	12
5.2	Two Approaches: Processing Using Memory (PUM) vs. Processing Near Memory (PNM)	13
6	Processing Using Memory (PUM)	14
6.1	RowClone	14
6.2	Ambit	15
6.3	SIMDRAM	17
6.4	Gather-Scatter DRAM	18
6.5	In-DRAM Security Primitives	18
7	Processing Near Memory (PNM)	20
7.1	Tesseract: Coarse-Grained Application-Level PNM Acceleration of Graph Processing	20
7.2	Function-Level PNM Acceleration of Mobile Consumer Workloads	21
7.3	Programmer-Transparent Function-Level PNM Acceleration of GPU Applications	22
7.4	Instruction-Level PNM Acceleration with PIM-Enabled Instructions (PEI)	23
7.5	Function-Level PNM Acceleration of Genome Analysis Workloads	24
7.6	Application-Level PNM Acceleration of Time Series Analysis	26
8	Enabling the Adoption of PIM	26
8.1	Programming Models and Code Generation for PIM	26
8.2	PIM Runtime: Scheduling and Data Mapping	27
8.3	Memory Coherence	29
8.4	Virtual Memory Support	30
8.5	Data Structures for PIM	30
8.6	Benchmarks and Simulation Infrastructures	31
8.7	Real PIM Hardware Systems and Prototypes	33
8.8	Security Considerations	36
9	Other Resources on PIM	37
10	Conclusion and Future Outlook	37

1. Introduction

Main memory, built using the Dynamic Random Access Memory (DRAM) technology, is a major component in nearly all computing systems, including servers, cloud platforms, mobile/embedded devices, and sensor systems. Across all of these systems, the data working set sizes of modern applications are rapidly growing, while the need for fast analysis of such data is increasing. Thus, main memory is becoming an increasingly significant bottleneck across a wide variety of computing systems and applications [1–26]. Alleviating the main memory bottleneck requires the memory capacity, energy, cost, and performance to all scale in an efficient manner across technology generations. Unfortunately, it has become increasingly difficult in recent years, especially the past decade, to scale all of these dimensions [1, 2, 27–59], and thus the main memory bottleneck has been worsening.

A major reason for the main memory bottleneck is the high energy and latency cost associated with *data movement*. In modern computers, to perform any operation on data that resides in main memory, the processor must retrieve the data from main memory. This requires the memory controller to issue commands to a DRAM module across a relatively slow and power-hungry off-chip bus (known as the *memory channel*). The DRAM module sends the requested data across the memory channel, after which the data is placed in the caches and registers. The CPU can perform computation on the data once the data is in its registers. Data movement from the DRAM to the CPU incurs long latency and consumes a significant amount of energy [7–9, 60–64]. These costs are often exacerbated by the fact that much of the data brought into the caches is *not reused* by the CPU [62, 63, 65, 66], providing little benefit in return for the high latency and energy cost.

The cost of data movement is a fundamental issue with the *processor-centric* nature of contemporary computer systems. The CPU is considered to be the master in the system, and computation is performed only in the processor (and accelerators). In contrast, data storage and communication units, including the main memory, are treated as unintelligent workers that are incapable of computation. As a result of this processor-centric design paradigm, data moves a lot in the system between the computation units and communication/storage units so that computation can be done on it. With the increasingly *data-centric* nature of contemporary and emerging applications, the processor-centric design paradigm leads to great inefficiency in performance, energy and cost. For example, most of the real estate within a single compute

PIM Review and Open Problems (II)

A Workload and Programming Ease Driven Perspective of Processing-in-Memory

Saugata Ghose[†] Amirali Boroumand[†] Jeremie S. Kim^{†\\$} Juan Gómez-Luna^{\\$} Onur Mutlu^{\\$†}

[†]*Carnegie Mellon University*

^{\\$}*ETH Zürich*

Saugata Ghose, Amirali Boroumand, Jeremie S. Kim, Juan Gomez-Luna, and Onur Mutlu,
"Processing-in-Memory: A Workload-Driven Perspective"

*Invited Article in IBM Journal of Research & Development, Special Issue on
Hardware for Artificial Intelligence, to appear in November 2019.
[Preliminary arXiv version]*

A Tutorial on PIM

- Onur Mutlu,

"Memory-Centric Computing Systems"

Invited Tutorial at *66th International Electron Devices Meeting (IEDM)*, Virtual, 12 December 2020.

[Slides (pptx) (pdf)]

[Executive Summary Slides (pptx) (pdf)]

[Tutorial Video (1 hour 51 minutes)]

[Executive Summary Video (2 minutes)]

[Abstract and Bio]

[Related Keynote Paper from VLSI-DAT 2020]

[Related Review Paper on Processing in Memory]

<https://www.youtube.com/watch?v=H3sEaINPBOE>



Memory-Centric Computing Systems

Onur Mutlu

omutlu@gmail.com

<https://people.inf.ethz.ch/omutlu>

12 December 2020

IEDM Tutorial



SAFARI

ETH zürich

Carnegie Mellon



0:06 / 1:51:05

CC G S Q E

IEDM 2020 Tutorial: Memory-Centric Computing Systems, Onur Mutlu, 12 December 2020

1,641 views • Dec 23, 2020

48

0

SHARE

SAVE

...



Onur Mutlu Lectures
13.9K subscribers

ANALYTICS

EDIT VIDEO

<https://www.youtube.com/onurmutlulectures>

205

Detailed Lectures on PIM (I)

- Computer Architecture, Fall 2020, Lecture 6
 - **Computation in Memory** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=oGcZAGwfEUE&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=12>
- Computer Architecture, Fall 2020, Lecture 7
 - **Near-Data Processing** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=j2GIigqn1Qw&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=13>
- Computer Architecture, Fall 2020, Lecture 11a
 - **Memory Controllers** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=TeG773OgiMQ&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=20>
- Computer Architecture, Fall 2020, Lecture 12d
 - **Real Processing-in-DRAM with UPMEM** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=Sscy1Wrr22A&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=25>

Detailed Lectures on PIM (II)

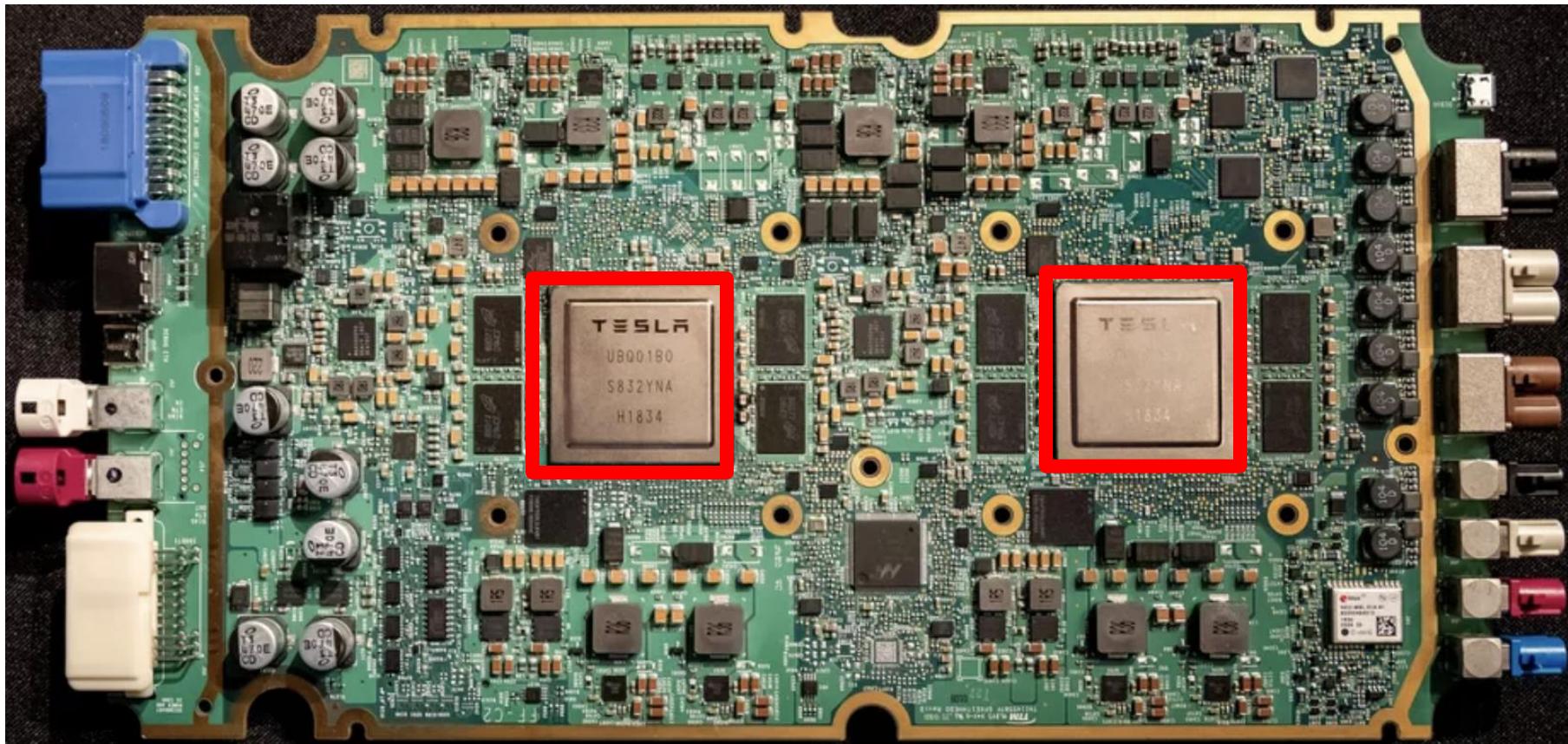
- Computer Architecture, Fall 2020, Lecture 15
 - **Emerging Memory Technologies** (ETH Zürich, Fall 2020)
 - https://www.youtube.com/watch?v=AIE1rD9G_YU&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=28
- Computer Architecture, Fall 2020, Lecture 16a
 - **Opportunities & Challenges of Emerging Memory Technologies** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=pmLszWGmMGQ&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=29>
- Computer Architecture, Fall 2020, Guest Lecture
 - **In-Memory Computing: Memory Devices & Applications** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=wNmqQHiEZNk&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=41>

Many Interesting Things
Are Happening Today
in Computer Architecture

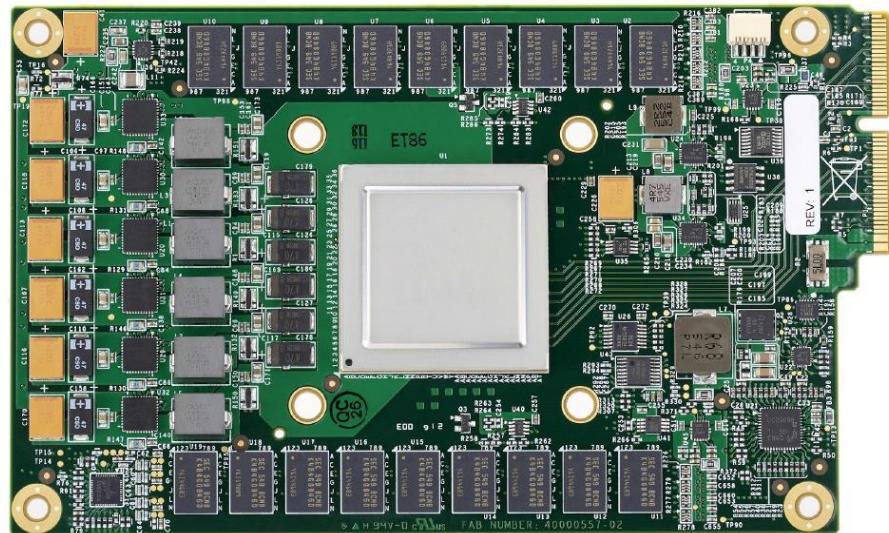
**Performance
and
Energy Efficiency**

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.



Google TPU Generation I (~2016)



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

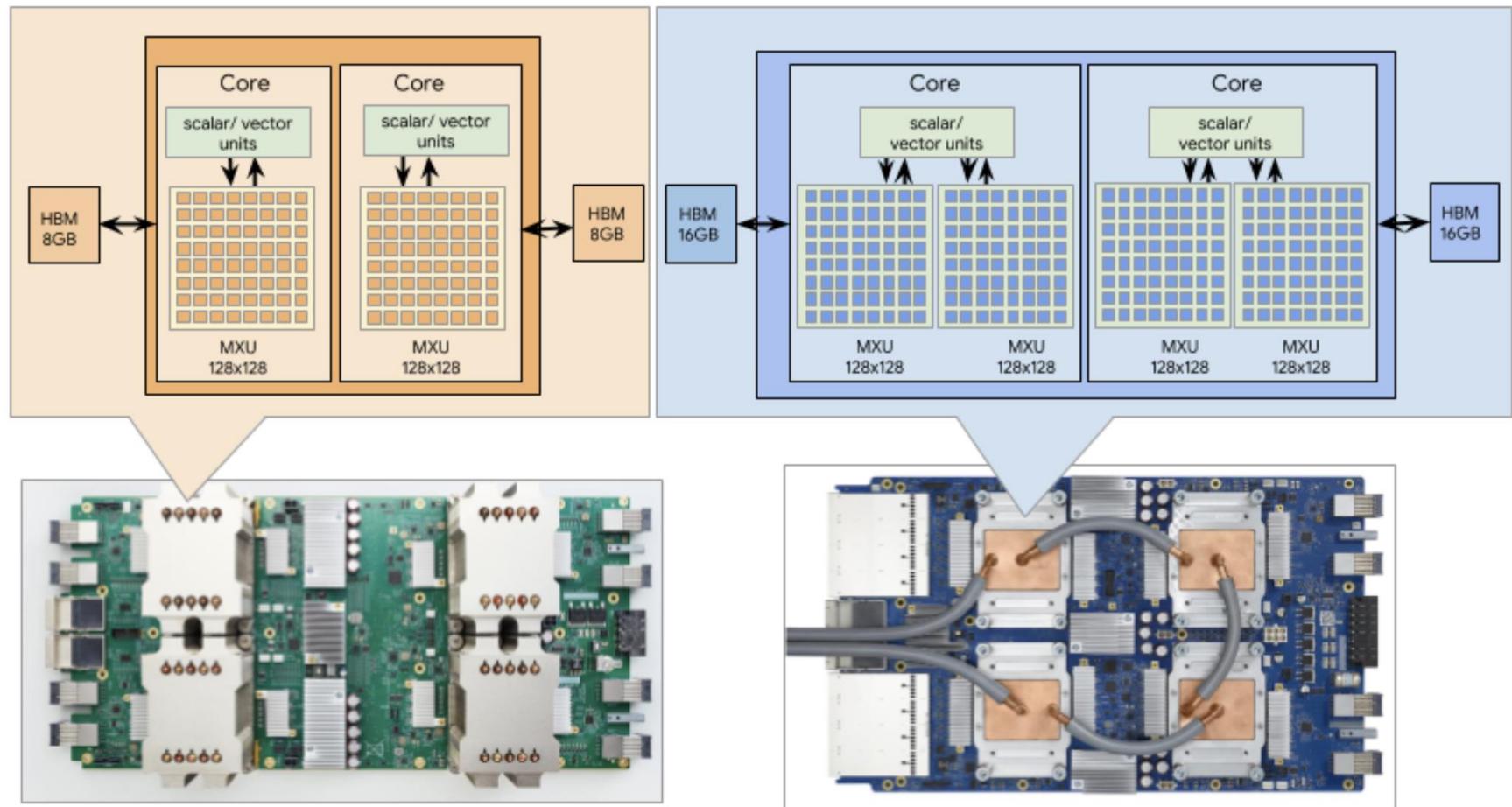
High Bandwidth Memory
vs DDR3

Floating point operations
vs FP16

45 TFLOPS per chip
vs 23 TOPS

Designed for training
and inference
vs only inference

Google TPU Generation III (2019)



TPU v2 - 4 chips, 2 cores per chip

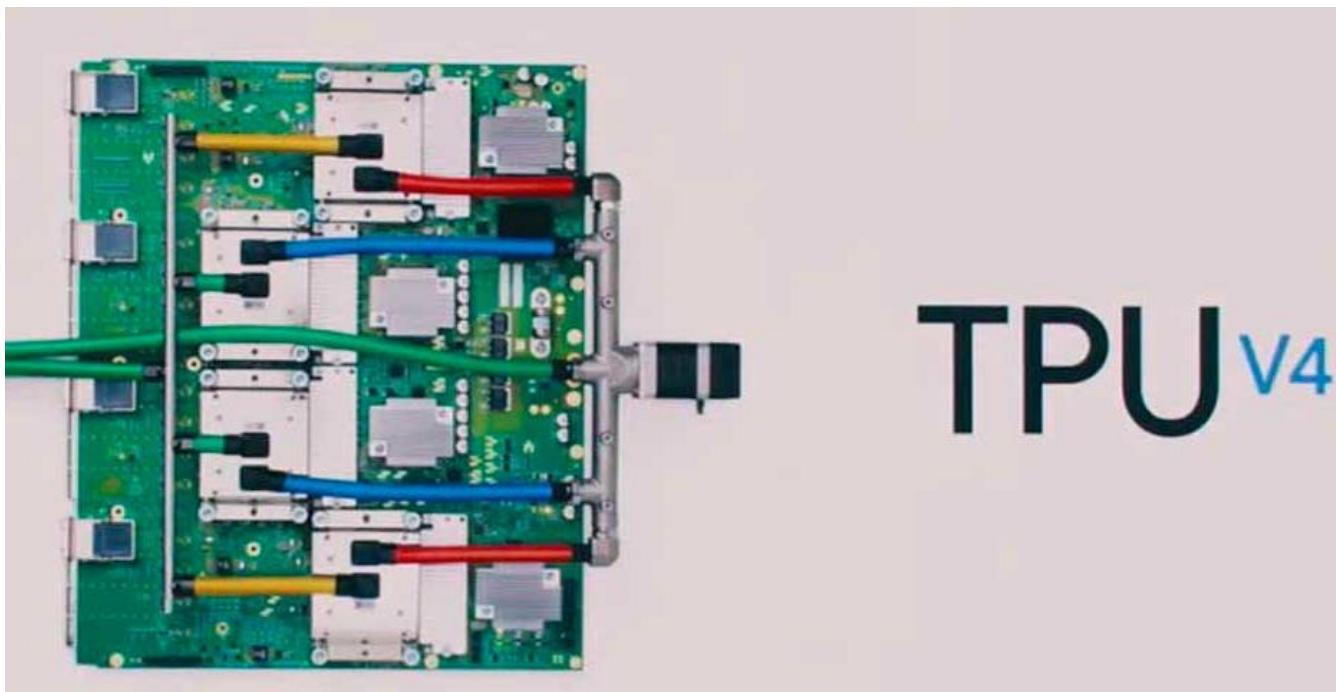
TPU v3 - 4 chips, 2 cores per chip

32GB HBM per chip
vs 16GB HBM in TPU2

4 Matrix Units per chip
vs 2 Matrix Units in TPU2

90 TFLOPS per chip
vs 45 TFLOPS in TPU2

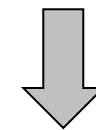
Google TPU Generation IV (2021)



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

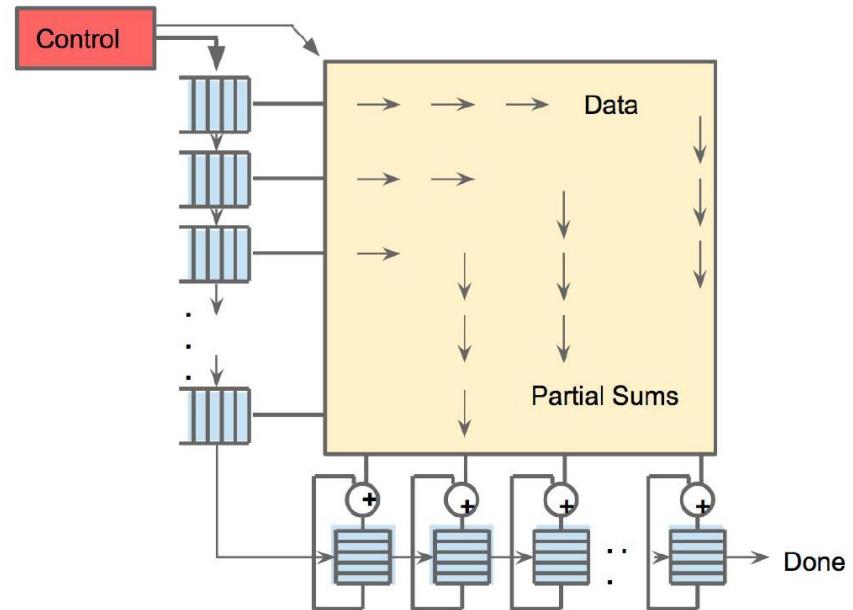


1 ExaFLOPS per board

<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)

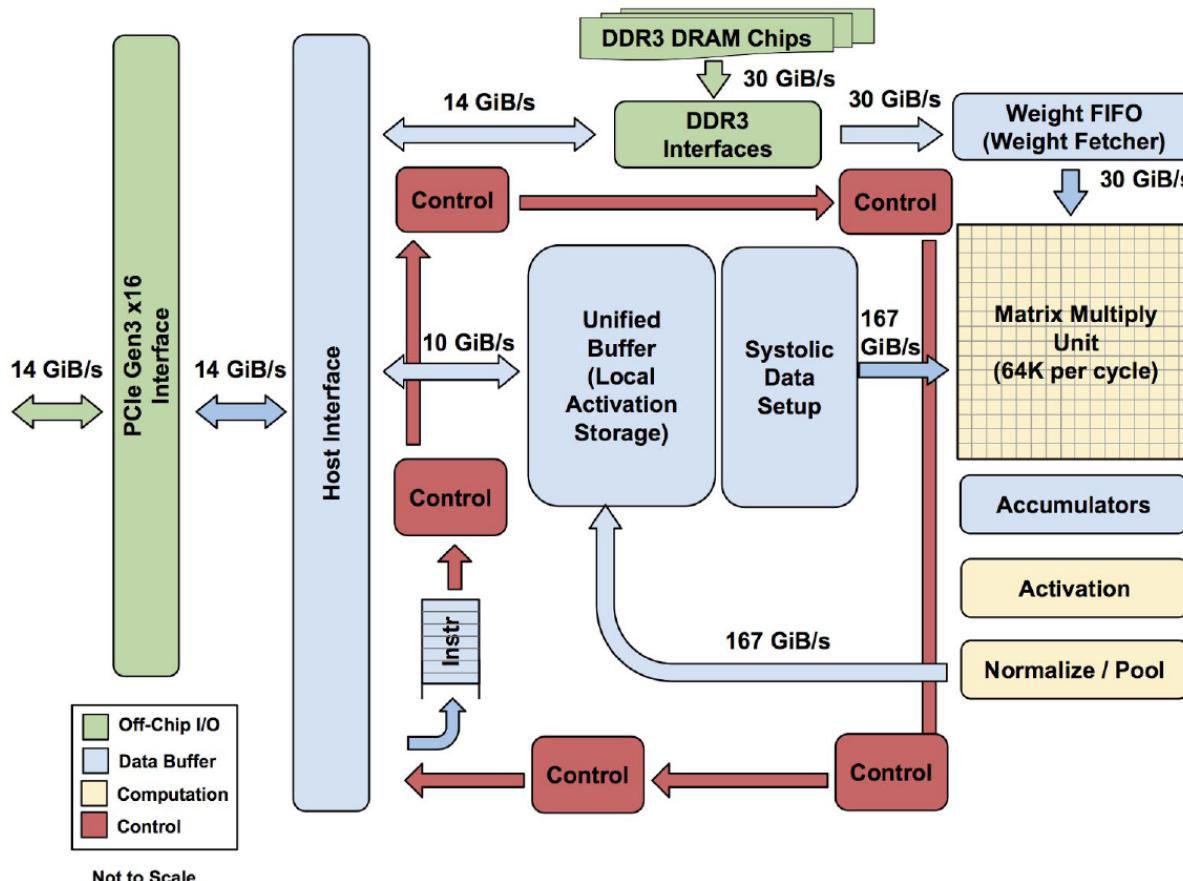


Figure 1. TPU Block Diagram. The main computation part is the yellow Matrix Multiply unit in the upper right hand corner. Its inputs are the blue Weight FIFO and the blue Unified Buffer (UB) and its output is the blue Accumulators (Acc). The yellow Activation Unit performs the nonlinear functions on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

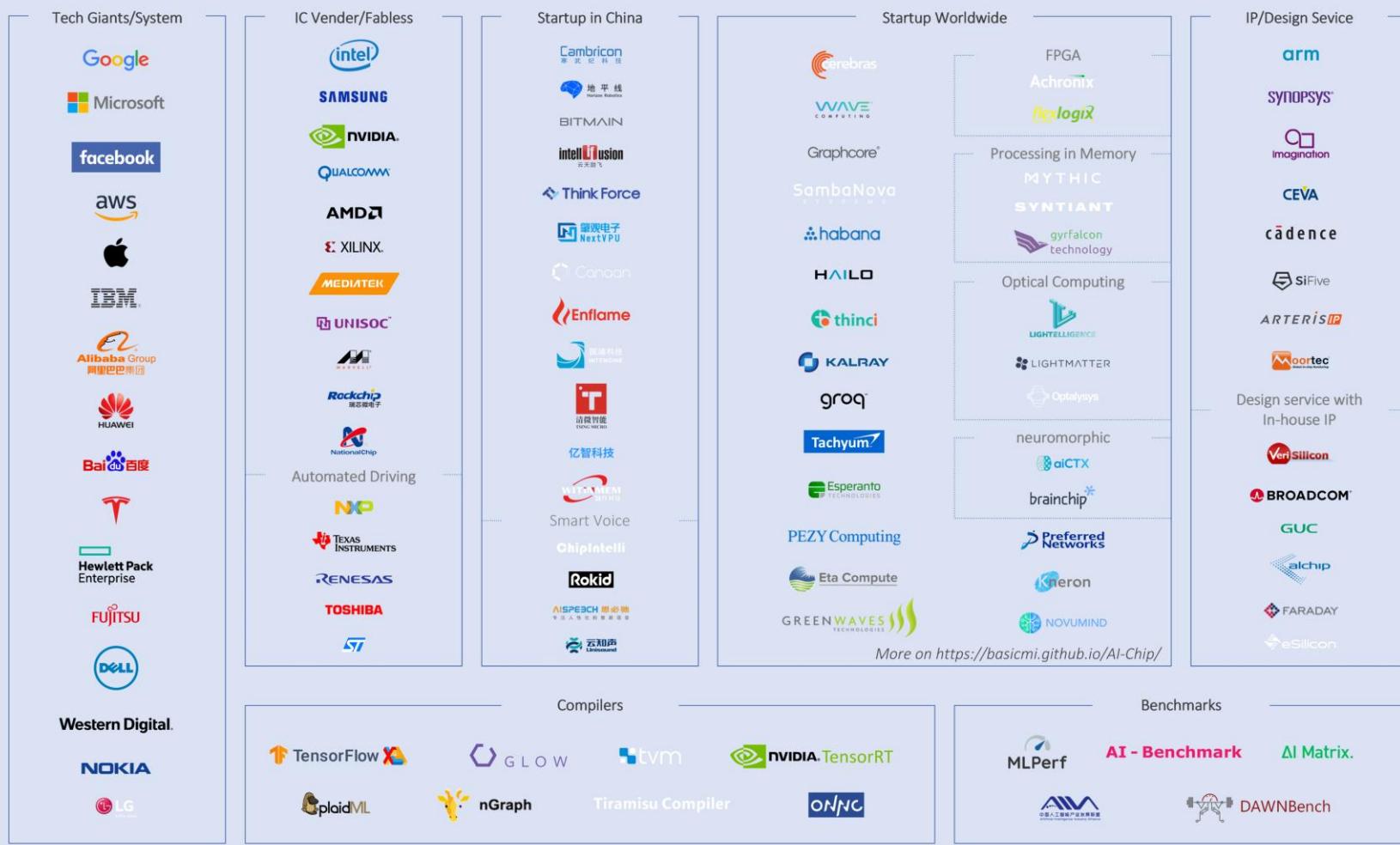
Many (Other) AI/ML Chips

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

Many (Other) AI/ML Chips (2019)

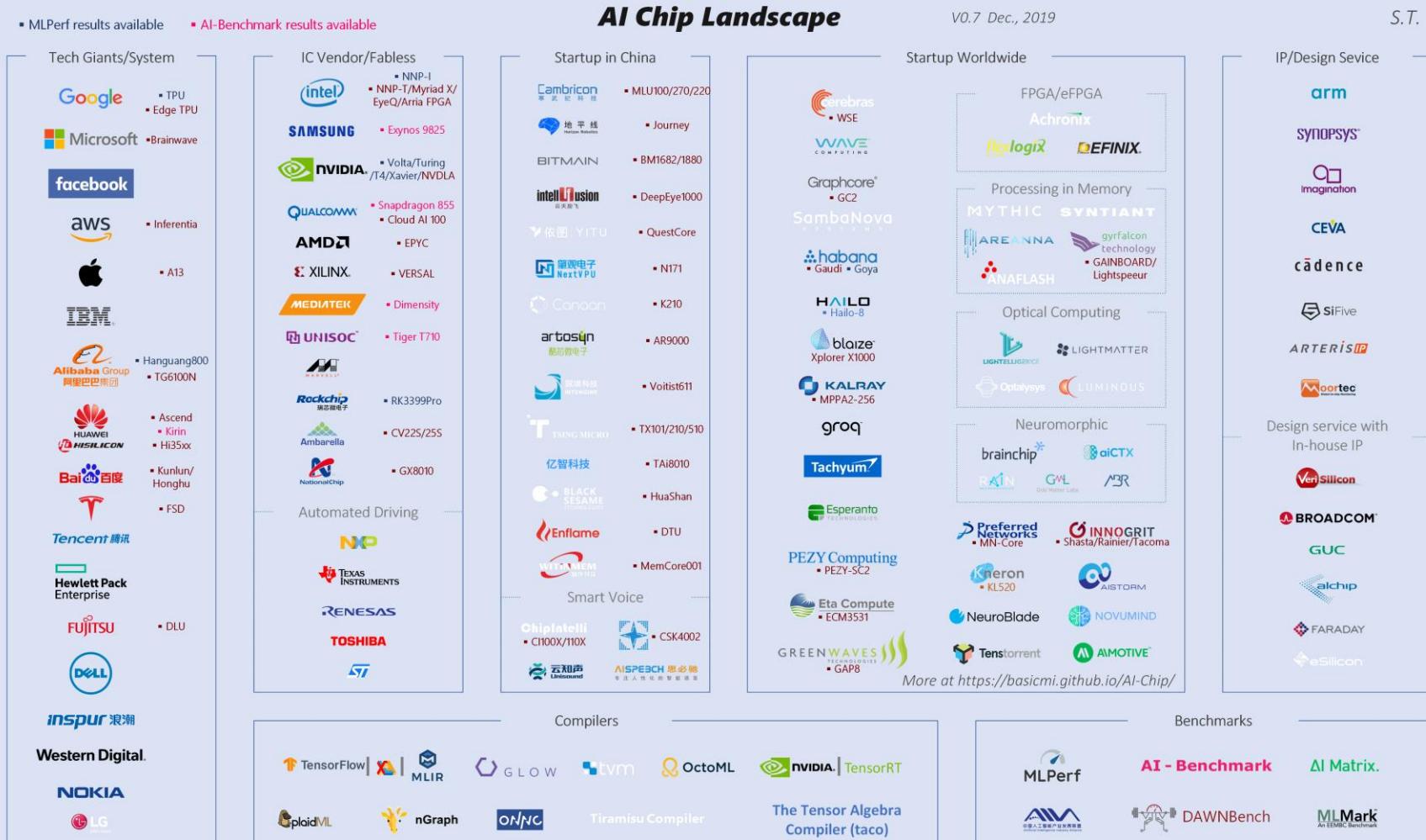
AI Chip Landscape

S.T.



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.

Many (Other) AI/ML Chips (2021)



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.

Many Interesting Things
Are Happening Today
in Computer Architecture

Many Interesting Things
Are Happening Today
in Computer Architecture

**Reliability
Security
Safety**

Security: RowHammer (2014)



The Story of RowHammer

- One can predictably induce bit flips in commodity DRAM chips
 - >80% of the 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

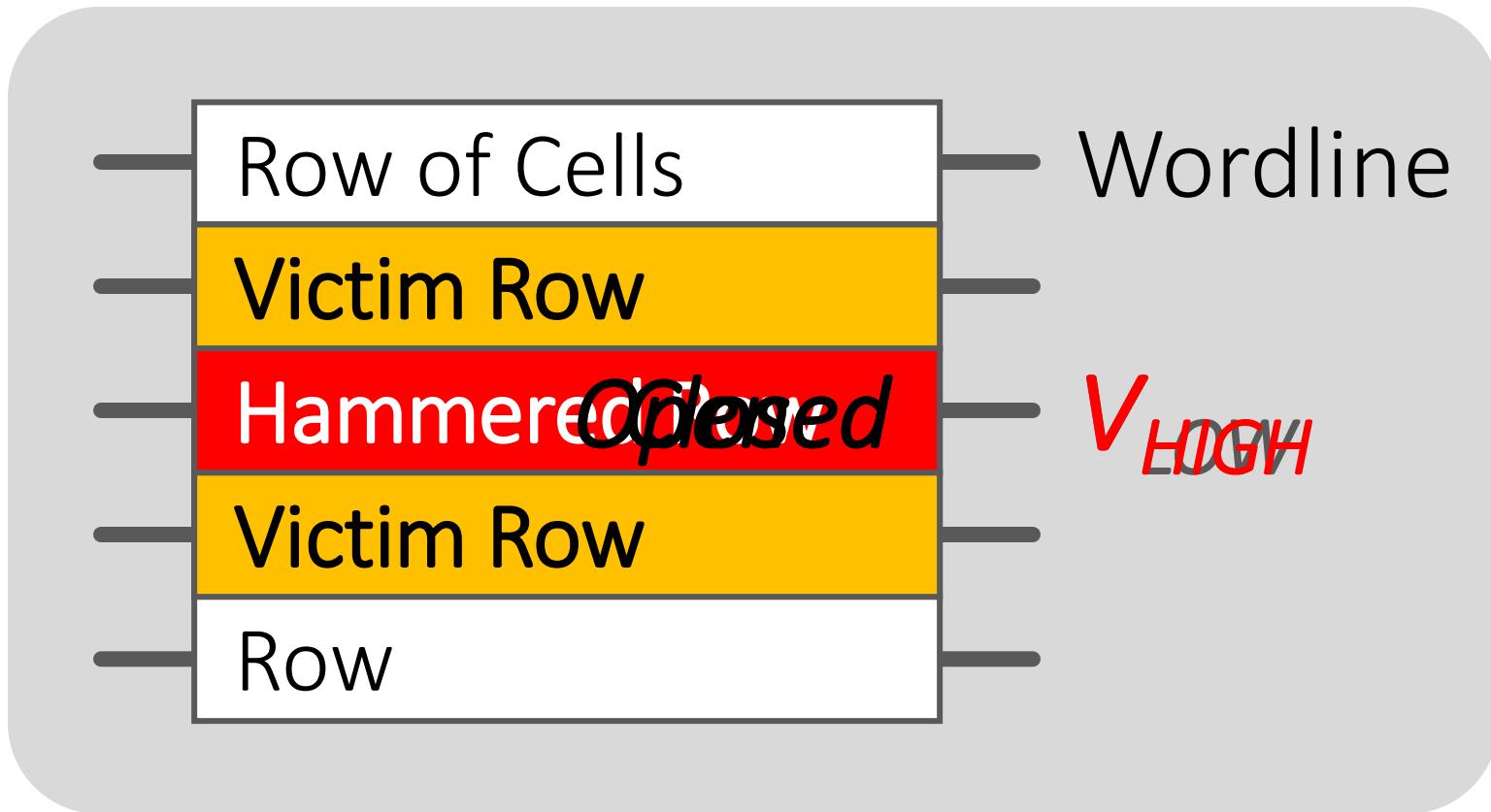
SHARE

 SHARE
18276

 TWEET

FORGET SOFTWARE—NOW HACKERS ARE EXPLOITING PHYSICS

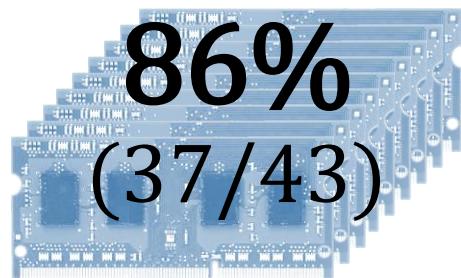
Modern DRAM is Prone to Disturbance Errors



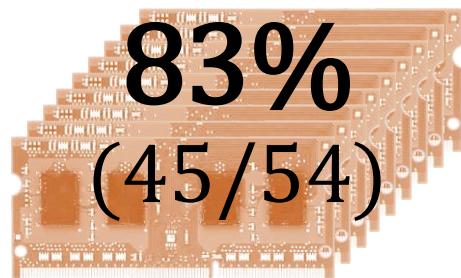
Repeatedly reading a row enough times (before memory gets refreshed) induces **disturbance errors** in adjacent rows in **most real DRAM chips you can buy today**

Most DRAM Modules Are Vulnerable

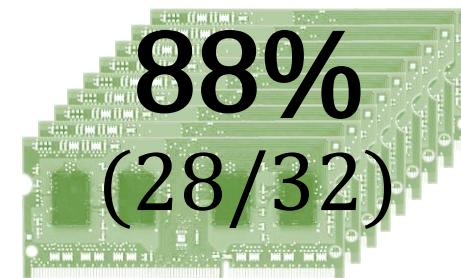
A company



B company



C company



Up to
 1.0×10^7
errors

Up to
 2.7×10^6
errors

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 (I)

"We can gain unrestricted access to systems of website visitors."

www.iaik.tugraz.at ■

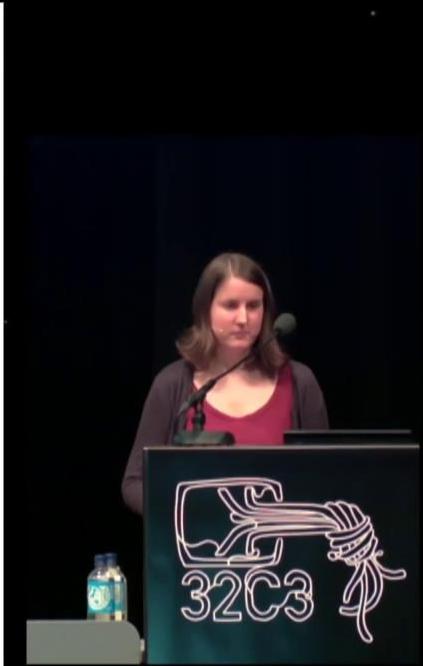
Not there yet, but ...



ROOT privileges for web apps!

29

Daniel Gruss (@lavados), Clémentine Maurice (@BloodyTangerine),
December 28, 2015 — 32c3, Hamburg, Germany



Rowhammer.js: A Remote Software-Induced Fault Attack in JavaScript (DIMVA'16)

More Security Implications (II)

"Can gain control of a smart phone deterministically"



Drammer: Deterministic Rowhammer
Attacks on Mobile Platforms, CCS'16²²⁶

More Security Implications (III)

- Using an integrated GPU in a mobile system to remotely escalate privilege via the WebGL interface

The screenshot shows a news article from Ars Technica. The header features the site's logo "ars TECHNICA" with "TECHNICA" in white on a black background and "ars" in white on an orange circle. Below the logo is a navigation bar with categories: BIZ & IT (highlighted in orange), TECH, SCIENCE, POLICY, CARS, and GAMING & CULTURE. The main title of the article is "GRAND PWNING UNIT" — Drive-by Rowhammer attack uses GPU to compromise an Android phone. A subtitle below it reads: "JavaScript based GLitch pwns browsers by flipping bits inside memory chips." The author is listed as DAN GOODIN - 5/3/2018, 12:00 PM.

"GRAND PWNING UNIT" —

Drive-by Rowhammer attack uses GPU to compromise an Android phone

JavaScript based GLitch pwns browsers by flipping bits inside memory chips.

DAN GOODIN - 5/3/2018, 12:00 PM

Grand Pwning Unit: Accelerating Microarchitectural Attacks with the GPU

Pietro Frigo
Vrije Universiteit
Amsterdam
p.frigo@vu.nl

Cristiano Giuffrida
Vrije Universiteit
Amsterdam
giuffrida@cs.vu.nl

Herbert Bos
Vrije Universiteit
Amsterdam
herbertb@cs.vu.nl

Kaveh Razavi
Vrije Universiteit
Amsterdam
kaveh@cs.vu.nl

More Security Implications (IV)

- Rowhammer over RDMA (I)

ars

TECHNICA

BIZ & IT TECH SCIENCE POLICY CARS GAMING & CULTURE

[THROWHAMMER —](#)

Packets over a LAN are all it takes to trigger serious Rowhammer bit flips

The bar for exploiting potentially serious DDR weakness keeps getting lower.

DAN GOODIN - 5/10/2018, 5:26 PM

Throwhammer: Rowhammer Attacks over the Network and Defenses

Andrei Tatar
VU Amsterdam

Radhesh Krishnan
VU Amsterdam

Elias Athanasopoulos
University of Cyprus

Cristiano Giuffrida
VU Amsterdam

Herbert Bos
VU Amsterdam

Kaveh Razavi
VU Amsterdam

More Security Implications (V)

- Rowhammer over RDMA (II)



Nethammer—Exploiting DRAM Rowhammer Bug Through Network Requests



Nethammer: Inducing Rowhammer Faults through Network Requests

Moritz Lipp
Graz University of Technology

Daniel Gruss
Graz University of Technology

Misiker Tadesse Aga
University of Michigan

Clémentine Maurice
Univ Rennes, CNRS, IRISA

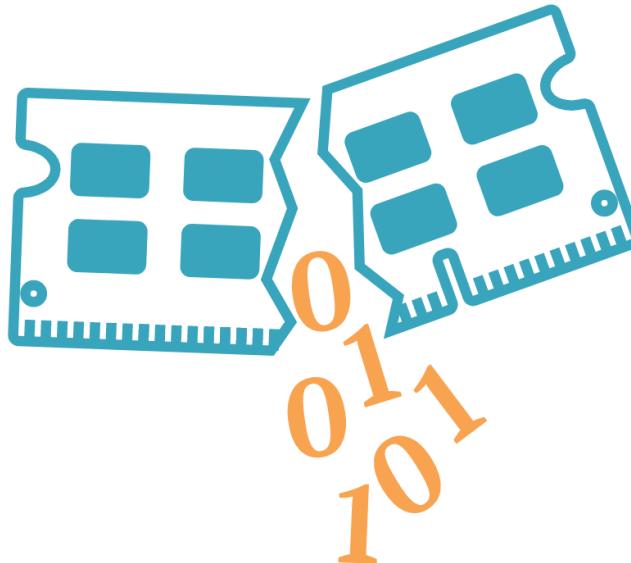
Michael Schwarz
Graz University of Technology

Lukas Raab
Graz University of Technology

Lukas Lamster
Graz University of Technology

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)

- USENIX Security 2019

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)

■ USENIX Security 2020

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

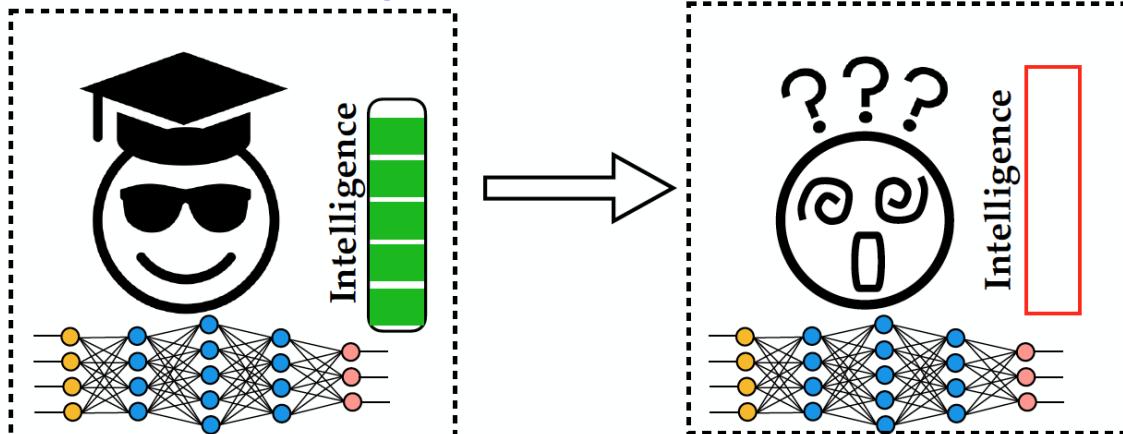
Deliang Fan

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)**



More Security Implications (IX)

- Rowhammer on MLC NAND Flash (based on [Cai+, HPCA 2017])



Security

Rowhammer RAM attack adapted to hit flash storage

Project Zero's two-year-old dog learns a new trick

By [Richard Chirgwin](#) 17 Aug 2017 at 04:27

17 SHARE ▼

**From random block corruption to privilege escalation:
A filesystem attack vector for rowhammer-like attacks**

Anil Kurmus

Nikolas Ioannou

Matthias Neugschwandtner
Thomas Parnell

Nikolaos Papandreou

IBM Research – Zurich

RowHammer: Seven Years Ago...

- Yoongu Kim, Ross Daly, Jeremie Kim, Chris Fallin, Ji Hye Lee, Donghyuk Lee, Chris Wilkerson, Konrad Lai, and Onur Mutlu,

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

Proceedings of the 41st International Symposium on Computer Architecture (ISCA), Minneapolis, MN, June 2014.

[[Slides \(pptx\)](#) ([pdf](#))] [[Lightning Session Slides \(pptx\)](#) ([pdf](#))] [[Source Code and Data](#)]

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

Yoongu Kim¹ Ross Daly* Jeremie Kim¹ Chris Fallin* Ji Hye Lee¹
Donghyuk Lee¹ Chris Wilkerson² Konrad Lai Onur Mutlu¹

¹Carnegie Mellon University

²Intel Labs

RowHammer: 2019 and Beyond...

- Onur Mutlu and Jeremie Kim,

"RowHammer: A Retrospective"

IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD) Special Issue on Top Picks in Hardware and Embedded Security, 2019.

[Preliminary arXiv version]

[Slides from COSADE 2019 (pptx)]

[Slides from VLSI-SOC 2020 (pptx) (pdf)]

[Talk Video (1 hr 15 minutes, with Q&A)]

RowHammer: A Retrospective

Onur Mutlu^{§‡}

[§]ETH Zürich

Jeremie S. Kim^{†§}

[†]Carnegie Mellon University

RowHammer in 2020-2023

RowHammer in 2020 (I)

- Jeremie S. Kim, Minesh Patel, A. Giray Yaglikci, Hasan Hassan, Roknoddin Azizi, Lois Orosa, and Onur Mutlu,
"Revisiting RowHammer: An Experimental Analysis of Modern Devices and Mitigation Techniques"

Proceedings of the 47th International Symposium on Computer Architecture (ISCA), Valencia, Spain, June 2020.

[Slides (pptx) (pdf)]

[Lightning Talk Slides (pptx) (pdf)]

[Talk Video (20 minutes)]

[Lightning Talk Video (3 minutes)]

Revisiting RowHammer: An Experimental Analysis of Modern DRAM Devices and Mitigation Techniques

Jeremie S. Kim^{§†} Minesh Patel[§] A. Giray Yağlıkçı[§]
Hasan Hassan[§] Roknoddin Azizi[§] Lois Orosa[§] Onur Mutlu^{§†}

[§]*ETH Zürich*

[†]*Carnegie Mellon University*

Key Takeaways from 1580 Chips

- Newer DRAM chips are more vulnerable to RowHammer
- There are chips today whose weakest cells fail after **only 4800 hammers**
- Chips of newer DRAM technology nodes can exhibit RowHammer bit flips 1) in **more rows** and 2) **farther away** from the victim row.
- Existing mitigation mechanisms are NOT effective

RowHammer in 2020 (II)

- Pietro Frigo, Emanuele Vannacci, Hasan Hassan, Victor van der Veen, Onur Mutlu, Cristiano Giuffrida, Herbert Bos, and Kaveh Razavi,

"TRRespass: Exploiting the Many Sides of Target Row Refresh"

Proceedings of the 41st IEEE Symposium on Security and Privacy (S&P), San Francisco, CA, USA, May 2020.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Lecture Slides \(pptx\)](#) ([pdf](#))]

[[Talk Video](#) (17 minutes)]

[[Lecture Video](#) (59 minutes)]

[[Source Code](#)]

[[Web Article](#)]

Best paper award.

Pwnie Award 2020 for Most Innovative Research. [Pwnie Awards 2020](#)

TRRespass: Exploiting the Many Sides of Target Row Refresh

Pietro Frigo*† Emanuele Vannacci*† Hasan Hassan§ Victor van der Veen¶
Onur Mutlu§ Cristiano Giuffrida* Herbert Bos* Kaveh Razavi*

RowHammer in 2020 (III)

- Lucian Cojocar, Jeremie Kim, Minesh Patel, Lillian Tsai, Stefan Saroiu, Alec Wolman, and Onur Mutlu,

"Are We Susceptible to Rowhammer? An End-to-End Methodology for Cloud Providers"

Proceedings of the 41st IEEE Symposium on Security and Privacy (S&P), San Francisco, CA, USA, May 2020.

[Slides (pptx) (pdf)]

[Talk Video (17 minutes)]

Are We Susceptible to Rowhammer?

An End-to-End Methodology for Cloud Providers

Lucian Cojocar, Jeremie Kim^{§†}, Minesh Patel[§], Lillian Tsai[‡],
Stefan Saroiu, Alec Wolman, and Onur Mutlu^{§†}
Microsoft Research, [§]ETH Zürich, [†]CMU, [‡]MIT

BlockHammer Solution in 2021

- A. Giray Yaglikci, Minesh Patel, Jeremie S. Kim, Roknoddin Azizi, Ataberk Olgun, Lois Orosa, Hasan Hassan, Jisung Park, Konstantinos Kanellopoulos, Taha Shahroodi, Saugata Ghose, and Onur Mutlu,

"BlockHammer: Preventing RowHammer at Low Cost by Blacklisting Rapidly-Accessed DRAM Rows"

Proceedings of the 27th International Symposium on High-Performance Computer Architecture (HPCA), Virtual, February-March 2021.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Short Talk Slides \(pptx\)](#) ([pdf](#))]

[[Talk Video](#) (22 minutes)]

[[Short Talk Video](#) (7 minutes)]

BlockHammer: Preventing RowHammer at Low Cost by Blacklisting Rapidly-Accessed DRAM Rows

A. Giray Yağlıkçı¹ Minesh Patel¹ Jeremie S. Kim¹ Roknoddin Azizi¹ Ataberk Olgun¹ Lois Orosa¹
Hasan Hassan¹ Jisung Park¹ Konstantinos Kanellopoulos¹ Taha Shahroodi¹ Saugata Ghose² Onur Mutlu¹

¹*ETH Zürich*

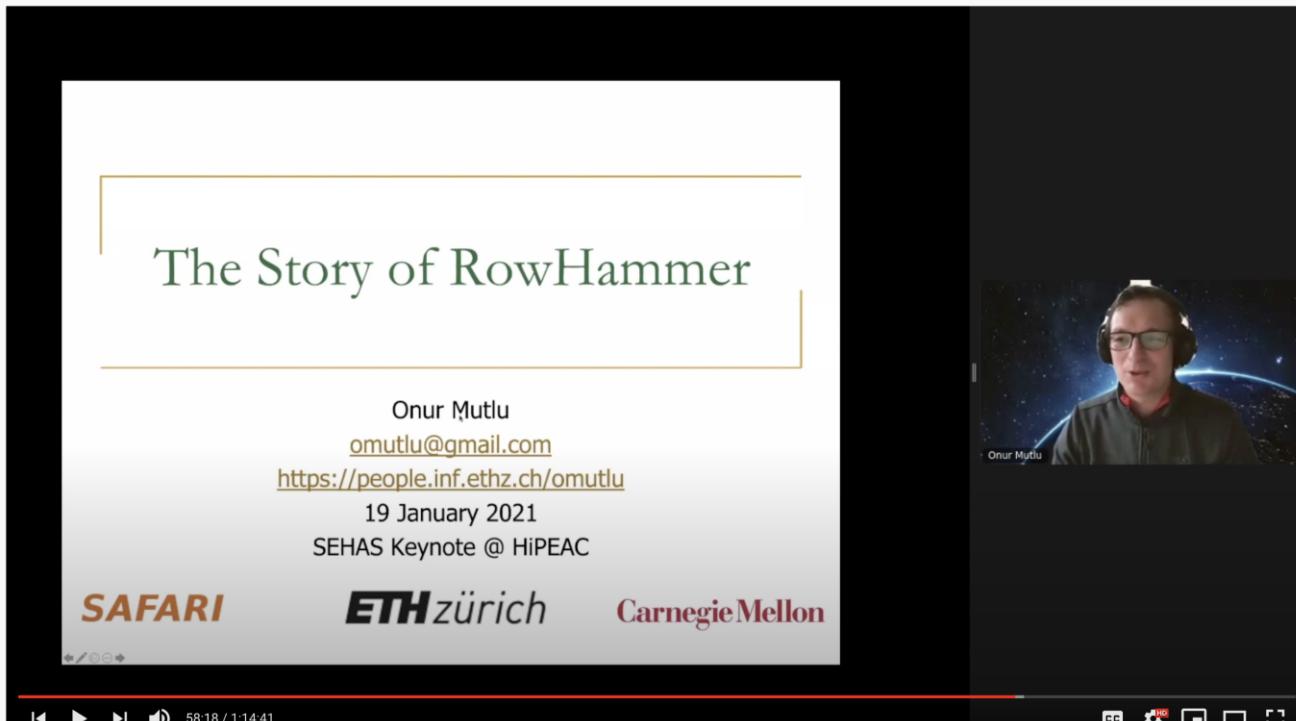
²*University of Illinois at Urbana-Champaign*

Detailed Lectures on RowHammer

- Computer Architecture, Fall 2020, Lecture 4b
 - RowHammer (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=KDy632z23UE&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=8>
- Computer Architecture, Fall 2020, Lecture 5a
 - RowHammer in 2020: TRRespass (ETH Zürich, Fall 2020)
 - https://www.youtube.com/watch?v=pwRw7QqK_qA&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=9
- Computer Architecture, Fall 2020, Lecture 5b
 - RowHammer in 2020: Revisiting RowHammer (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=gR7XR-Eepcg&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=10>
- Computer Architecture, Fall 2020, Lecture 5c
 - Secure and Reliable Memory (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=HvswnsfG3oQ&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=11>

The Story of RowHammer Lecture ...

- Onur Mutlu,
[**"The Story of RowHammer"**](#)
Keynote Talk at *Secure Hardware, Architectures, and Operating Systems Workshop (SeHAS)*, held with *HiPEAC 2021 Conference*, Virtual, 19 January 2021.
[[Slides \(pptx\)](#) ([pdf](#))]
[[Talk Video](#) (1 hr 15 minutes, with Q&A)]



Rowhammer



Two RowHammer Papers at MICRO 2021

- Lois Orosa, Abdullah Giray Yaglikci, Haocong Luo, Ataberk Olgun, Jisung Park, Hasan Hassan, Minesh Patel, Jeremie S. Kim, and Onur Mutlu,
"A Deeper Look into RowHammer's Sensitivities: Experimental Analysis of Real DRAM Chips and Implications on Future Attacks and Defenses"

Proceedings of the 54th International Symposium on Microarchitecture (MICRO), Virtual, October 2021.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Short Talk Slides \(pptx\)](#) ([pdf](#))]

[[Lightning Talk Slides \(pptx\)](#) ([pdf](#))]

[[Talk Video](#) (21 minutes)]

[[Lightning Talk Video](#) (1.5 minutes)]

[[arXiv version](#)]

A Deeper Look into RowHammer's Sensitivities: Experimental Analysis of Real DRAM Chips and Implications on Future Attacks and Defenses

Lois Orosa*
ETH Zürich

A. Giray Yağlıkçı*
ETH Zürich

Haocong Luo
ETH Zürich

Ataberk Olgun
ETH Zürich, TOBB ETÜ

Jisung Park
ETH Zürich

Hasan Hassan
ETH Zürich

Minesh Patel
ETH Zürich

Jeremie S. Kim
ETH Zürich

Onur Mutlu
ETH Zürich

Two RowHammer Papers at MICRO 2021

- Hasan Hassan, Yahya Can Tugrul, Jeremie S. Kim, Victor van der Veen, Kaveh Razavi, and Onur Mutlu,

"Uncovering In-DRAM RowHammer Protection Mechanisms: A New Methodology, Custom RowHammer Patterns, and Implications"

Proceedings of the 54th International Symposium on Microarchitecture (MICRO), Virtual, October 2021.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Short Talk Slides \(pptx\)](#) ([pdf](#))]

[[Lightning Talk Slides \(pptx\)](#) ([pdf](#))]

[[Talk Video](#) (25 minutes)]

[[Lightning Talk Video](#) (100 seconds)]

[[arXiv version](#)]

Uncovering In-DRAM RowHammer Protection Mechanisms: A New Methodology, Custom RowHammer Patterns, and Implications

Hasan Hassan[†]

Yahya Can Tuğrul^{†‡}
Kaveh Razavi[†]

Jeremie S. Kim[†]
Onur Mutlu[†]

Victor van der Veen^σ

[†]*ETH Zürich*

[‡]*TOBB University of Economics & Technology*

^σ*Qualcomm Technologies Inc.*

A New RowHammer Paper at DSN 2022

- A. Giray Yağlıkçı, Haocong Luo, Geraldo F. de Oliviera, Ataberk Olgun, Minesh Patel, Jisung Park, Hasan Hassan, Jeremie S. Kim, Lois Orosa, and Onur Mutlu,
"Understanding RowHammer Under Reduced Wordline Voltage: An Experimental Study Using Real DRAM Devices"

Proceedings of the 52nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Baltimore, MD, USA, June 2022.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Lightning Talk Slides \(pptx\)](#) ([pdf](#))]

[[arXiv version](#)]

[[Talk Video](#) (34 minutes, including Q&A)]

[[Lightning Talk Video](#) (2 minutes)]

Understanding RowHammer Under Reduced Wordline Voltage: An Experimental Study Using Real DRAM Devices

A. Giray Yağlıkçı¹ Haocong Luo¹ Geraldo F. de Oliviera¹ Ataberk Olgun¹ Minesh Patel¹
Jisung Park¹ Hasan Hassan¹ Jeremie S. Kim¹ Lois Orosa^{1,2} Onur Mutlu¹

¹*ETH Zürich*

²*Galicia Supercomputing Center (CESGA)*

TRRespass Key Takeaways

RowHammer is still
an open problem

Security by obscurity
is likely not a good solution

Security: Meltdown and Spectre (2018)



MELTDOWN



SPECTRE

Meltdown and Spectre

- Someone can steal secret data from the system **even though**
 - your program and data are perfectly correct and
 - your hardware behaves according to the specification and
 - there are no software vulnerabilities/bugs
- Why?
 - Speculative execution leaves traces of secret data in the processor's cache (internal storage)
 - It brings data that is not supposed to be brought/accessible if there was no speculative execution
 - A malicious program can inspect the contents of the cache to "infer" secret data that it is not supposed to access
 - A malicious program can actually force another program to speculatively execute code that leaves traces of secret data

More on Meltdown/Spectre Vulnerabilities

Project Zero

News and updates from the Project Zero team at Google

Wednesday, January 3, 2018

Reading privileged memory with a side-channel

Posted by Jann Horn, Project Zero

We have discovered that CPU data cache timing can be abused to efficiently leak information out of mis-speculated execution, leading to (at worst) arbitrary virtual memory read vulnerabilities across local security boundaries in various contexts.

Are We Now RowHammer Free in 2023?

- **Appeared at ISCA 2023**

RowPress: Amplifying Read-Disturbance in Modern DRAM Chips

Haocong Luo Ataberk Olgun A. Giray Yağlıkçı Yahya Can Tuğrul Steve Rhyner
Meryem Banu Cavlak Joël Lindegger Mohammad Sadrosadati Onur Mutlu

ETH Zürich

[**https://people.inf.ethz.ch/omutlu/pub/RowPress_isca23.pdf**](https://people.inf.ethz.ch/omutlu/pub/RowPress_isca23.pdf)



RowPress

RowPress [ISCA 2023]

- Haocong Luo, Ataberk Olgun, Giray Yaglikci, Yahya Can Tugrul, Steve Rhyner, M. Banu Cavlak, Joel Lindegger, Mohammad Sadrosadati, and Onur Mutlu,
"[RowPress: Amplifying Read Disturbance in Modern DRAM Chips](#)"

Proceedings of the [50th International Symposium on Computer Architecture \(ISCA\)](#), Orlando, FL, USA, June 2023.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Lightning Talk Slides \(pptx\)](#) ([pdf](#))]

[[Lightning Talk Video](#) (3 minutes)]

[[RowPress Source Code and Datasets \(Officially Artifact Evaluated with All Badges\)](#)]

***Officially artifact evaluated as available, reusable and reproducible.
Best artifact award at ISCA 2023.***

RowPress: Amplifying Read-Disturbance in Modern DRAM Chips

Haocong Luo Ataberk Olgun A. Giray Yağlıkçı Yahya Can Tuğrul Steve Rhyner
Meryem Banu Cavlak Joël Lindegger Mohammad Sadrosadati Onur Mutlu

ETH Zürich



RowPress

Amplifying Read Disturbance in Modern DRAM Chips

ISCA 2023 Session 2B: Monday 19 June, 2:15 PM EDT

Haocong Luo

Ataberk Olgun

A. Giray Yağlıkçı

Yahya Can Tuğrul

Steve Rhyner

Meryem Banu Cavlak

Joël Lindegger

Mohammad Sadrosadati

Onur Mutlu

SAFARI

ETH Zürich

High-Level Summary

- We demonstrate and analyze **RowPress, a new read disturbance phenomenon** that causes bitflips in real DRAM chips
- We show that RowPress is **different from the RowHammer vulnerability**
- We demonstrate RowPress **using a user-level program** on a real Intel system with real DRAM chips
- We provide **effective solutions** to RowPress

What is RowPress?

Keeping a DRAM row **open for a long time**
causes bitflips in adjacent rows

These bitflips do **NOT** require many row activations

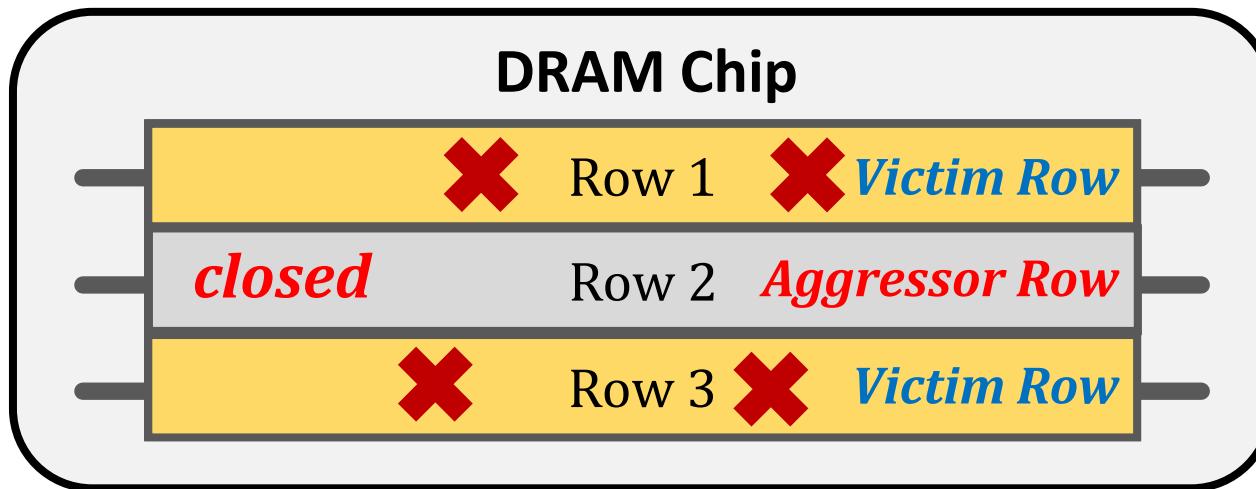
Only one activation is enough in some cases!



Now, let's delve into some background and
see how this is **different from RowHammer**

Read Disturbance in DRAM

- Read disturbance in DRAM breaks memory isolation
- **Prominent example: RowHammer**



Repeatedly **opening (activating)** and **closing** a DRAM row
many times causes **RowHammer bitflips** in adjacent rows

Are There Other Read-Disturb Issues in DRAM?

- RowHammer is the only studied read-disturb phenomenon
- Mitigations work by detecting **high row activation count**

What if there is another read-disturb phenomenon
that **does NOT rely on high row activation count?**

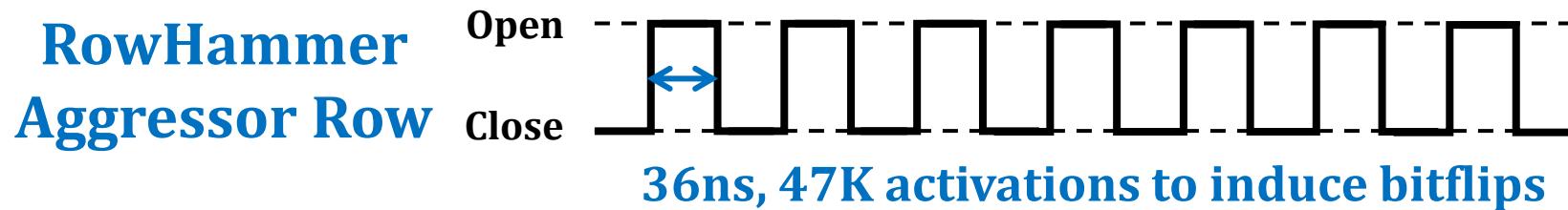


https://www.reddit.com/r/CrappyDesign/comments/arw0q8/now_this_this_is_poor_fencing/

RowPress vs. RowHammer

Instead of using a high activation count,

- ☛ increase the time that the aggressor row stays open



We observe bitflips even with **ONLY ONE activation** in extreme cases where the row stays open for 30ms

Real DRAM Chip Characterization (I)

FPGA-Based DDR4 Testing Infrastructure

- Based on [SoftMC \[Hassan+, HPCA'17\]](#) and [DRAM Bender \[Olgun+, TCAD'23\]](#)
- Fine-grained control over DRAM commands, timings, and temperature



Real DRAM Chip Characterization (II)

DRAM Chips Tested

- 164 DDR4 chips from all 3 major DRAM manufacturers

Mfr.	#DIMMs	#Chips	Density	Die Rev.	Org.	Date
Mfr. S (Samsung)	2	8	8Gb	B	x8	20-53
	1	8	8Gb	C	x8	N/A
	3	8	8Gb	D	x8	21-10
	2	8	4Gb	F	x8	N/A
Mfr. H (SK Hynix)	1	8	4Gb	A	x8	19-46
	1	8	4Gb	X	x8	N/A
	2	8	16Gb	A	x8	20-51
	2	8	16Gb	C	x8	21-36
Mfr. M (Micron)	1	16	8Gb	B	x4	N/A
	2	4	16Gb	B	x16	21-26
	1	16	16Gb	E	x4	20-14
	2	4	16Gb	E	x16	20-46
	1	4	16Gb	F	x16	21-50

Major Takeaways from Real DRAM Chips

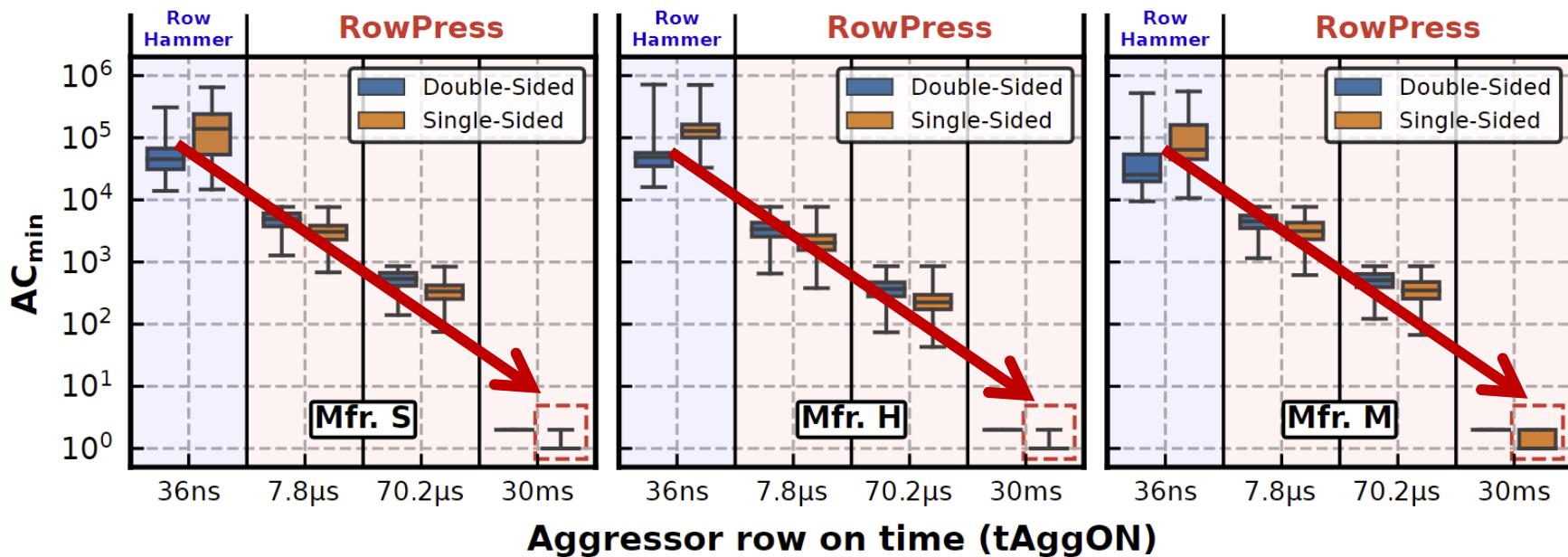
RowPress significantly **amplifies** DRAM's vulnerability to **read disturbance**

RowPress has a **different** underlying error **mechanism** from RowHammer

Key Characteristics of RowPress (I)

Amplifying Read Disturbance in DRAM

- Reduces the minimum number of row activations needed to induce a bitflip (AC_{min}) by **1-2 orders of magnitude**
- In extreme cases, activating a row **only once** induces bitflips



Key Characteristics of RowPress (II)

Amplifying Read Disturbance in DRAM

- Reduces the minimum number of row activations needed to induce a bitflip (AC_{min}) by **1-2 orders of magnitude**
- In extreme cases, activating a row **only once** induces bitflips
- Gets worse as **temperature increases**

Different From RowHammer

- Affects a **different set of cells** compared to RowHammer and retention failures
- **Behaves differently** as access pattern and temperature changes compared to RowHammer

Real-System Demonstration (I)



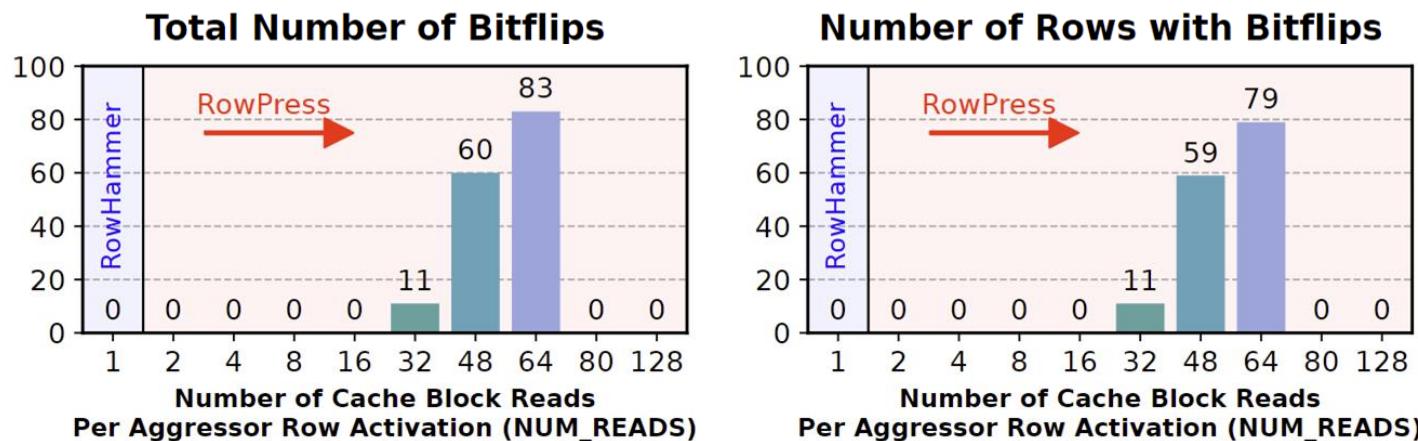
Intel Core i5-10400
(Comet Lake)



Samsung DDR4 Module
M378A2K43CB1-CTD
(Date Code: 20-10)
w/ TRR RowHammer Mitigation

Key Idea: A proof-of-concept RowPress program keeps a DRAM row open for a longer period by **keeping on accessing different cache blocks in the row**

Real-System Demonstration (II)



Key Idea: A proof-of-concept RowPress program keeps a DRAM row open for a longer period by **keeping on accessing different cache blocks in the row**

Leveraging RowPress, a user-level program
induces bitflips when RowHammer cannot

How to Avoid RowPress Bitflips

We propose a methodology to **adapt existing RowHammer mitigations to also mitigate RowPress**

Key Mechanisms:

1. Limit the **maximum time** that a row can stay open
2. Configure the RowHammer mitigation to account for the **RowPress-induced reduction in the number of activations needed to cause bitflips**

Our solutions **mitigate RowPress at low additional performance overhead**

More Results & Source Code

Many more results & analyses in the paper

- 6 major takeaways
- 19 major empirical observations
- 3 more potential mitigations



Fully open source and artifact evaluated

- <https://github.com/CMU-SAFARI/RowPress>





RowPress

Amplifying Read Disturbance in Modern DRAM Chips

Haocong Luo

Ataberk Olgun

A. Giray Yağlıkçı

Yahya Can Tuğrul

Steve Rhyner

Meryem Banu Cavlak

Joël Lindegger

Mohammad Sadrosadati *Onur Mutlu*

<https://github.com/CMU-SAFARI/RowPress>

SAFARI

ETH Zürich

Many Interesting Things
Are Happening Today
in Computer Architecture

Many Interesting Things
Are Happening Today
in Computer Architecture

More Demanding Workloads

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.

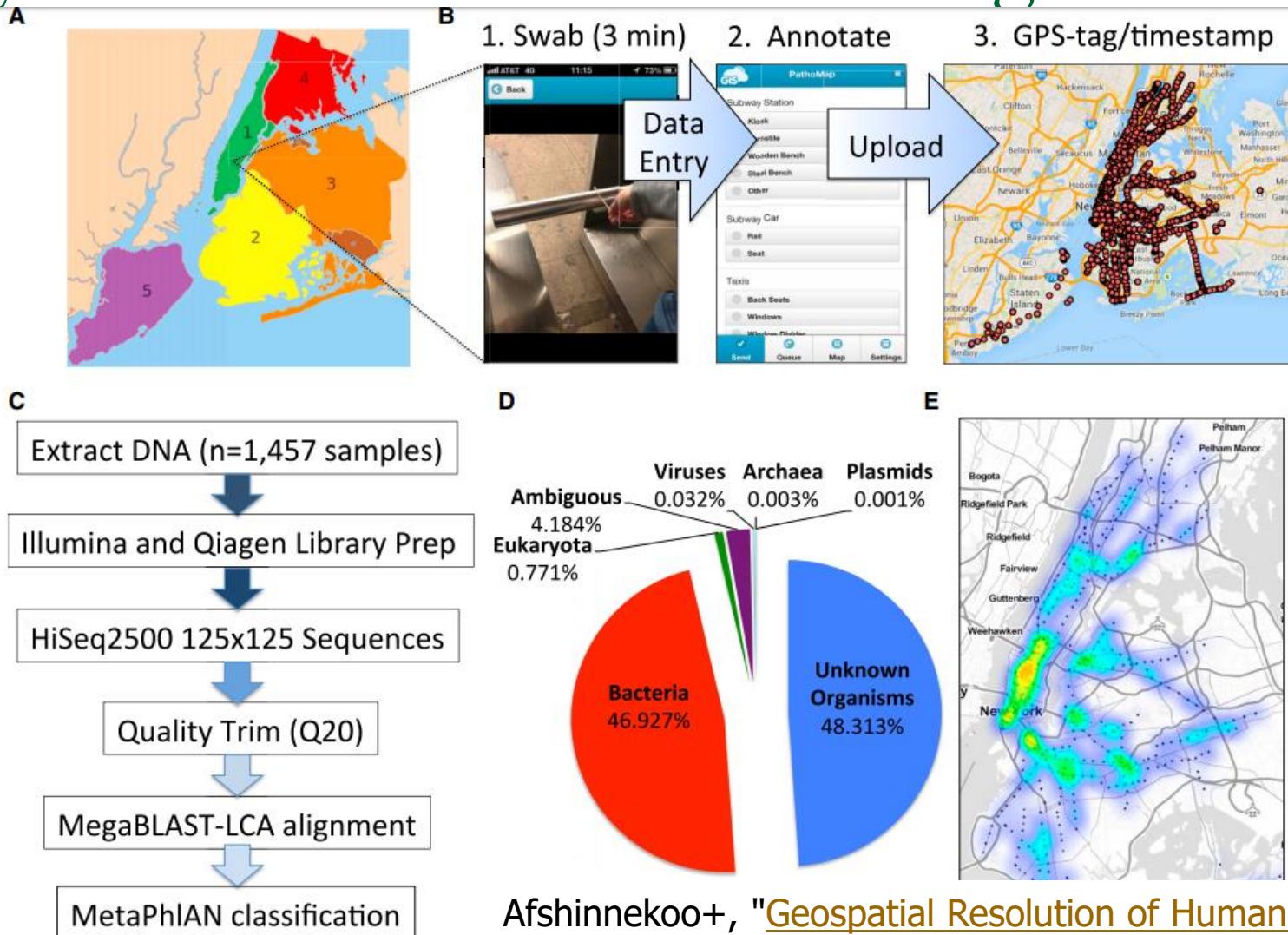


700Kg of Oxford Nanopore sequencers and consumables are on their way for use by Chinese scientists in understanding the current coronavirus outbreak.

Population-Scale Microbiome Profiling



City-Scale Microbiome Profiling



Afshinnekoo+, "[Geospatial Resolution of Human and Bacterial Diversity with City-Scale Metagenomics](#)", Cell Systems, 2015

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.

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
SAFARI 278

High-Throughput Genome Sequencers



Illumina MiSeq



Illumina NovaSeq 6000



Pacific
Biosciences
Sequel II



Pacific Biosciences RS II

Oxford
Nanopore
PromethION



Oxford Nanopore MinION



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

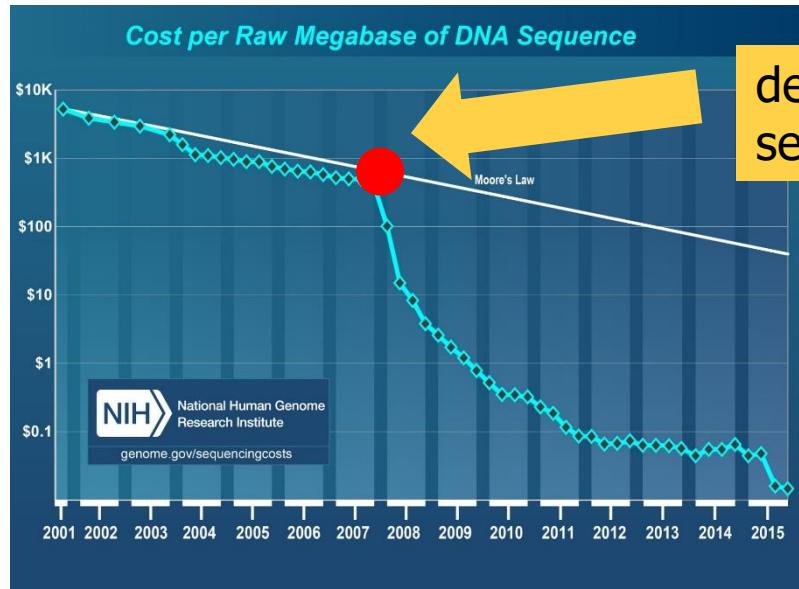
July-Aug. 2021, pp. 39-48, vol. 41

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



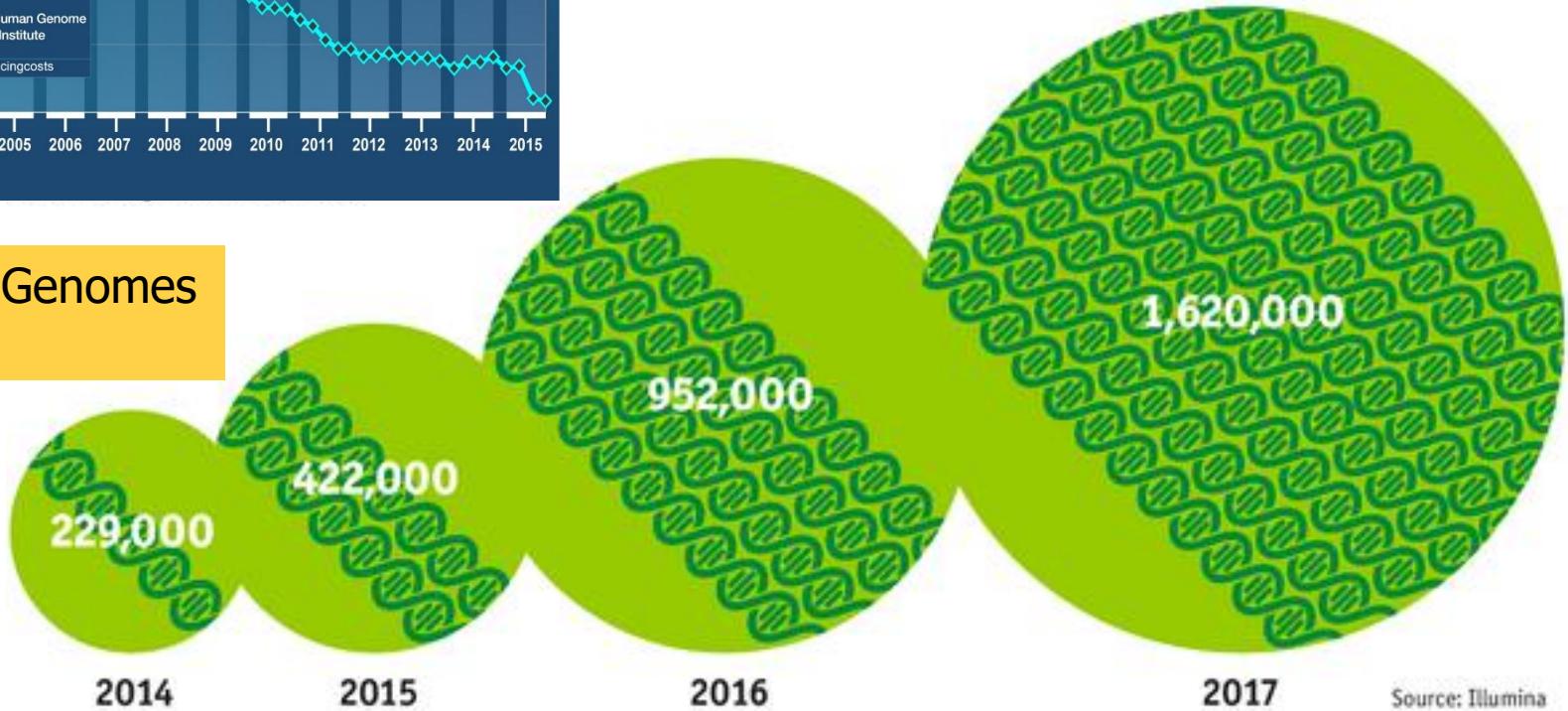
SmidgION from ONT

The Genomic Era



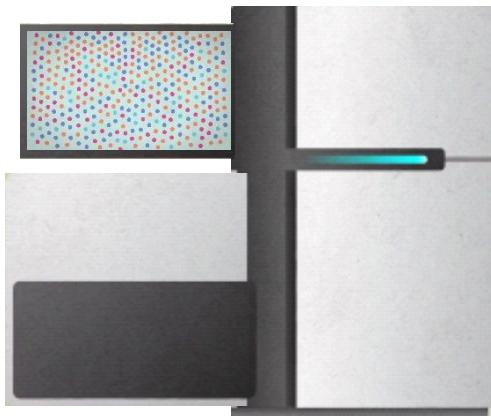
development of high-throughput sequencing (HTS) technologies

Number of Genomes Sequenced



The Economist

Source: Illumina

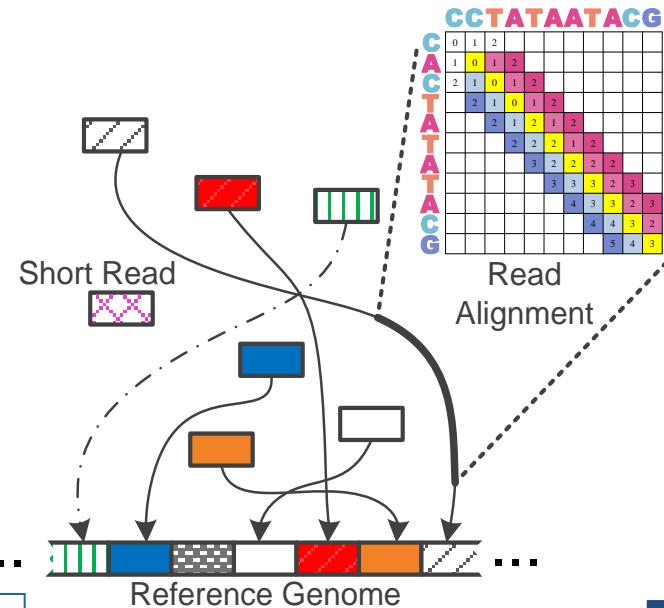


Billions of Short Reads

```
TATATATAACGTACGTACGT  
TTTAGTACGTACGTACGT  
ATACGTACGTACGTACGT  
ACG CCCCTACGTA  
ACGTACTAGTACGT  
TTAGTACGTACGT  
TACGTACTAAAGTACGT  
TACGTACTAGTACGT  
TTTAAAAACGTA  
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

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

Shifted Hamming Distance: SIMD Acceleration

<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

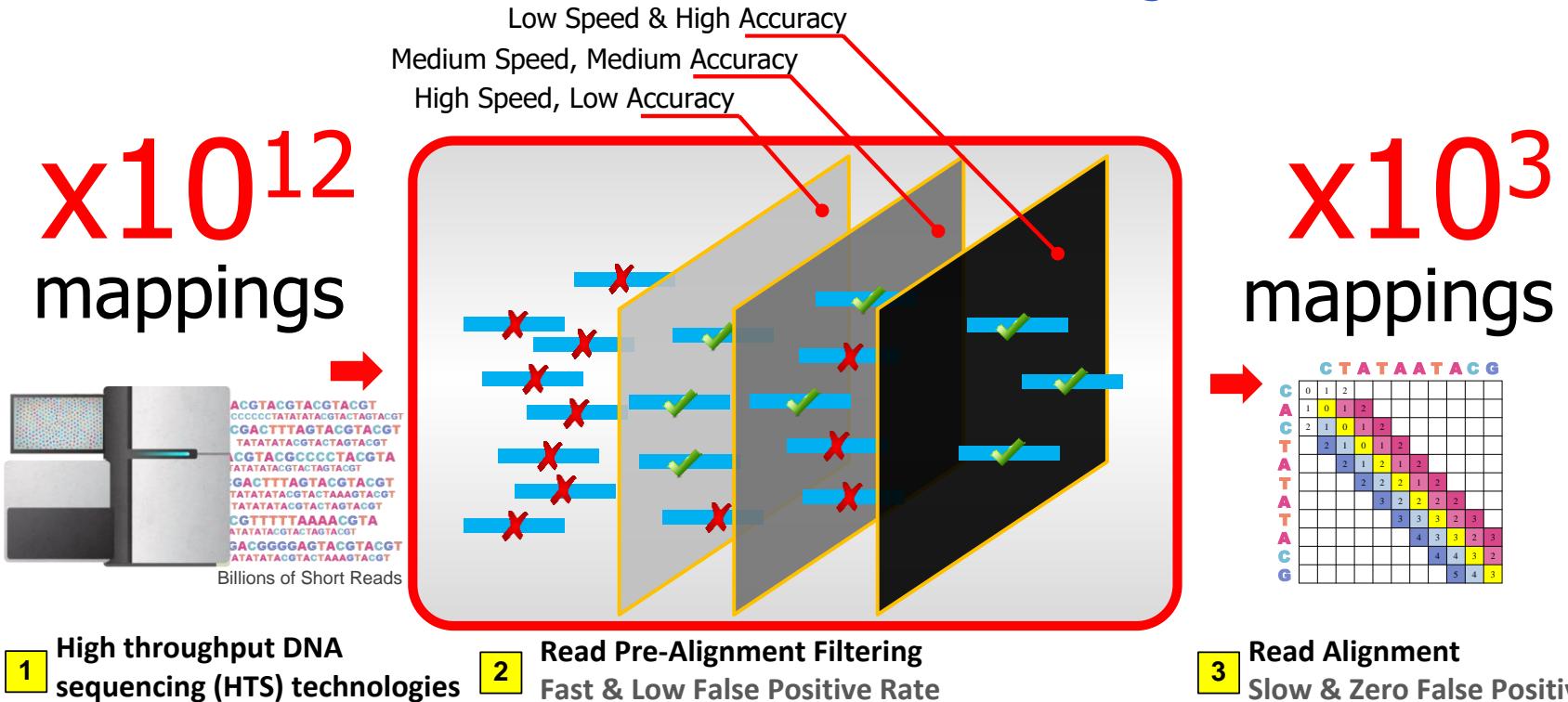
Sequence analysis

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 Alignment Filtering



GateKeeper: FPGA-Based Alignment Filtering

- 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



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,*}**

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

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

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

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

GenASM Framework [MICRO 2020]

- Damla Senol Cali, Gurpreet S. Kalsi, Zulal Bingol, 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*

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](#))]

[[Lightning Talk Video](#) (90 seconds)]

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

New Applications: Graph Genomes [ISCA 2022]

- Damla Senol Cali, Konstantinos Kanellopoulos, Joel Lindegger, Zulal Bingol, Gurpreet S. Kalsi, Ziyi Zuo, Can Firtina, Meryem Banu Cavlak, Jeremie Kim, Nika MansouriGhiasi, 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.

[[Slides \(pptx\)](#) ([pdf](#))]
[[arXiv version](#)]

SeGram: Accelerating Genomic Sequence-to-Graph Mapping via Algorithm/Hardware Co-Design

Damla Senol Cali Bionano Genomics USA	Konstantinos Kanellopoulos ETH Zurich Switzerland	Joel Lindegger ETH Zurich Switzerland
Zü'lal Bingöl Bilkent University Turkey	Gurpreet S. Kalsi Intel USA	Ziyi Zuo Carnegie Mellon University USA
Can Firtina ETH Zurich Switzerland	Gagandeep Singh ETH Zurich Switzerland	Nika Mansouri Ghiasi ETH Zurich Switzerland
Jeremie Kim ETH Zurich Switzerland	Meryem Banu Cavlak ETH Zurich Switzerland	Juan Gómez Luna ETH Zurich Switzerland
Nour Almadhoun Alserr ETH Zurich Switzerland	Mohammed Alser ETH Zurich Switzerland	Sreenivas Subramoney Intel Labs India
Can Alkan Bilkent University Turkey	Saugata Ghose University of Illinois Urbana-Champaign USA	Onur Mutlu ETH Zurich Switzerland

New Applications: Ref Genome Updates

RESEARCH

AirLift: A Fast and Comprehensive Technique for Remapping Alignments between Reference Genomes

Jeremie S. Kim¹, Can Firtina¹, Meryem Banu Cavlak², Damla Senol Cali³, Nastaran Hajinazar^{1,4},
Mohammed Alser¹, Can Alkan² and Onur Mutlu^{1,2,3*}

https://people.inf.ethz.ch/omutlu/pub/AirLift_genome-remapper_arxiv21.pdf

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

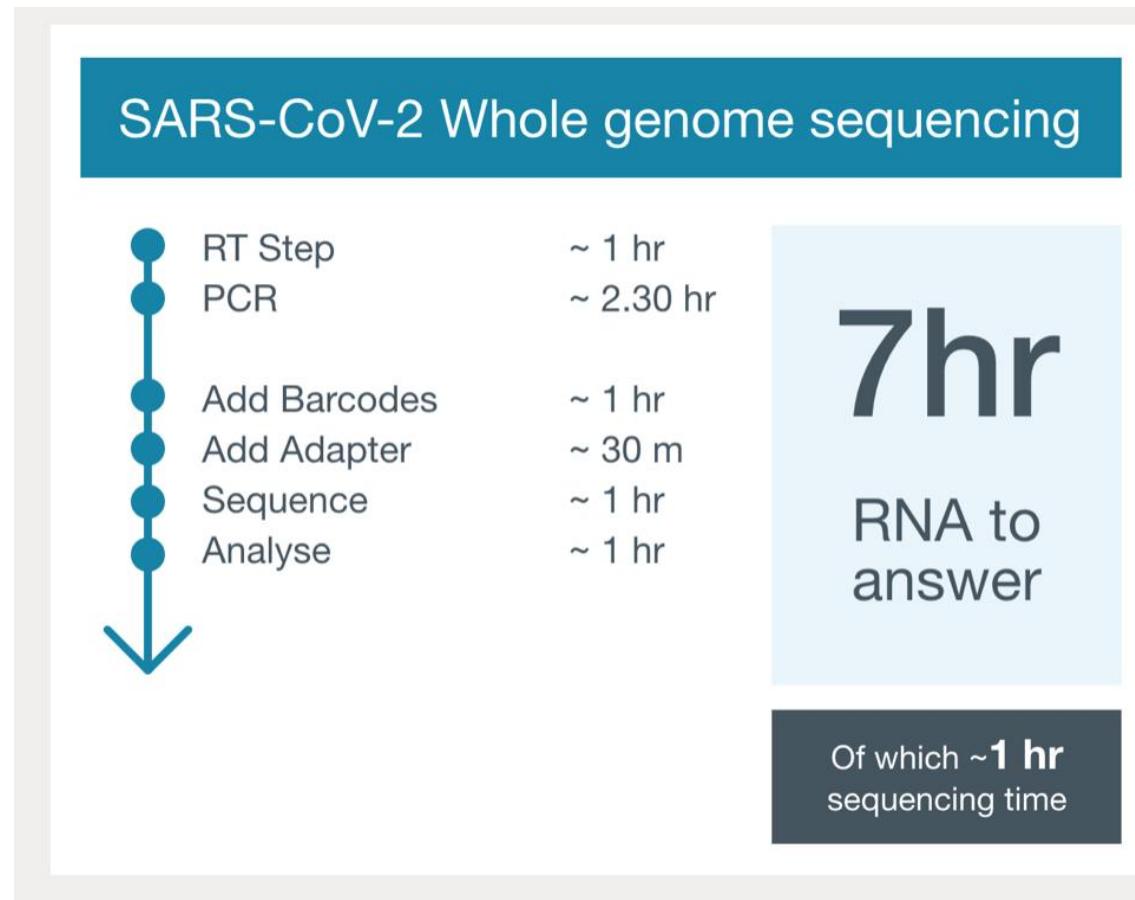
July-Aug. 2021, pp. 39-48, vol. 41

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



SmidgION from ONT

COVID-19 Nanopore Sequencing (I)



- From ONT (<https://nanoporetech.com/covid-19/overview>)

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 MinION 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

- From ONT (<https://nanoporetech.com/covid-19/overview>)

Accelerating Genome Analysis: Overview

- Mohammed Alser, Zulal Bingol, 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

More on Fast Genome Analysis ...

- Onur Mutlu,

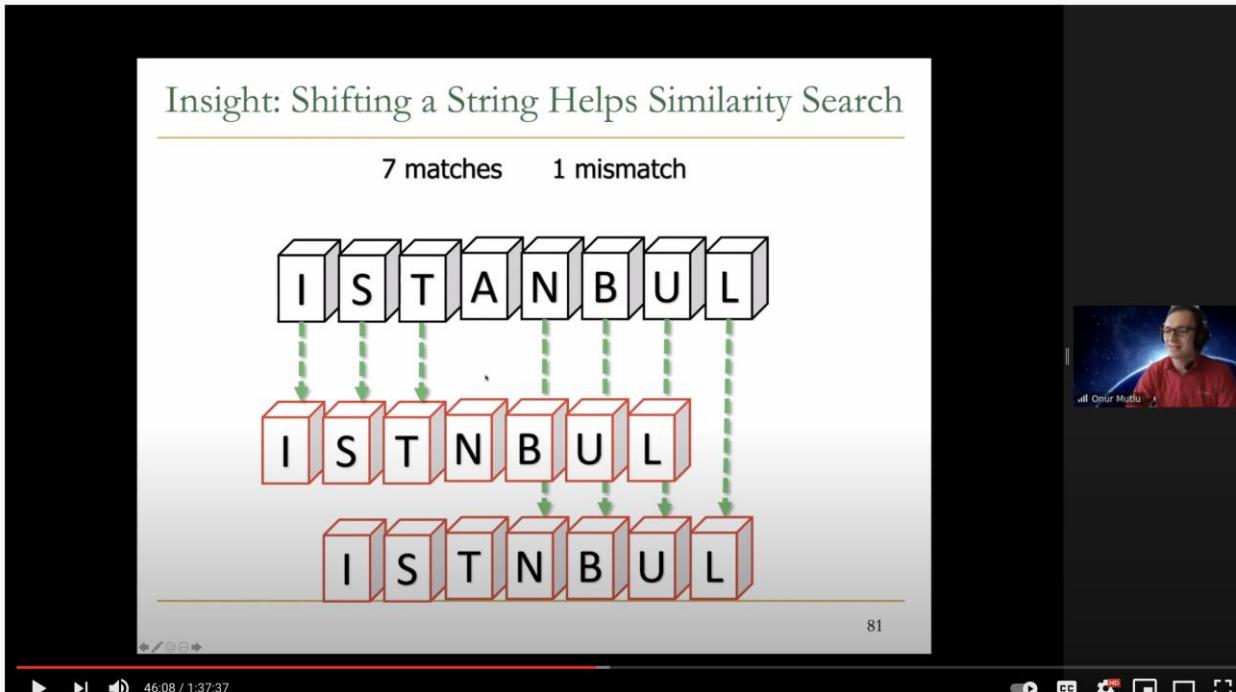
["Accelerating Genome Analysis: A Primer on an Ongoing Journey"](#)

Invited Lecture at [Technion](#), Virtual, 26 January 2021.

[[Slides \(pptx\)](#) ([pdf](#))]

[[Talk Video](#) (1 hour 37 minutes, including Q&A)]

[[Related Invited Paper \(at IEEE Micro, 2020\)](#)]



Onur Mutlu - Invited Lecture @Technion: Accelerating Genome Analysis: A Primer on an Ongoing Journey

566 views • Premiered Feb 6, 2021

31 0 SHARE SAVE ...

Detailed Lectures on Genome Analysis

- Computer Architecture, Fall 2020, Lecture 3a
 - **Introduction to Genome Sequence Analysis** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=CrRb32v7SJc&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=5>
- Computer Architecture, Fall 2020, Lecture 8
 - **Intelligent Genome Analysis** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=ygmQpdDTL7o&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=14>
- Computer Architecture, Fall 2020, Lecture 9a
 - **GenASM: Approx. String Matching Accelerator** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=XoLpzmN-Pas&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=15>
- Accelerating Genomics Project Course, Fall 2020, Lecture 1
 - **Accelerating Genomics** (ETH Zürich, Fall 2020)
 - <https://www.youtube.com/watch?v=rgjl8ZyLsAg&list=PL5Q2soXY2Zi9E2bBVAgCqLgwiDRQDTyId>

Many Interesting Things
Are Happening Today
in Computer Architecture

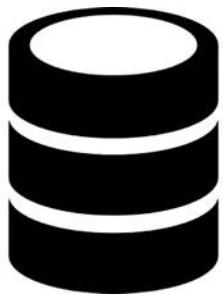
More Demanding Workloads

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]



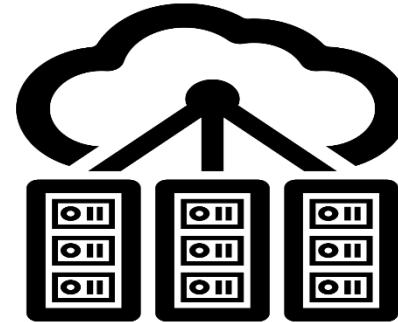
Graph/Tree Processing

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



In-Memory Data Analytics

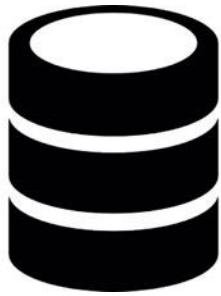
[Clapp+ (Intel), IISWC'15;
Awan+, BDCloud'15]



Datacenter Workloads

[Kanев+ (Google), ISCA'15]

Data Overwhelms Modern Machines



In-memory Databases



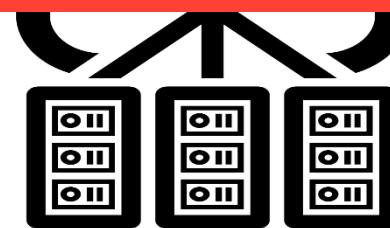
Graph/Tree Processing

Data → performance & energy bottleneck



In-Memory Data Analytics

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



Datacenter Workloads

[Kanев+ (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



Video Playback

Google's **video codec**



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

Rachata Ausavarungnirun¹

Aki Kuusela³

Saugata Ghose¹

Eric Shiu³

Allan Knies³

Youngsok Kim²

Rahul Thakur³

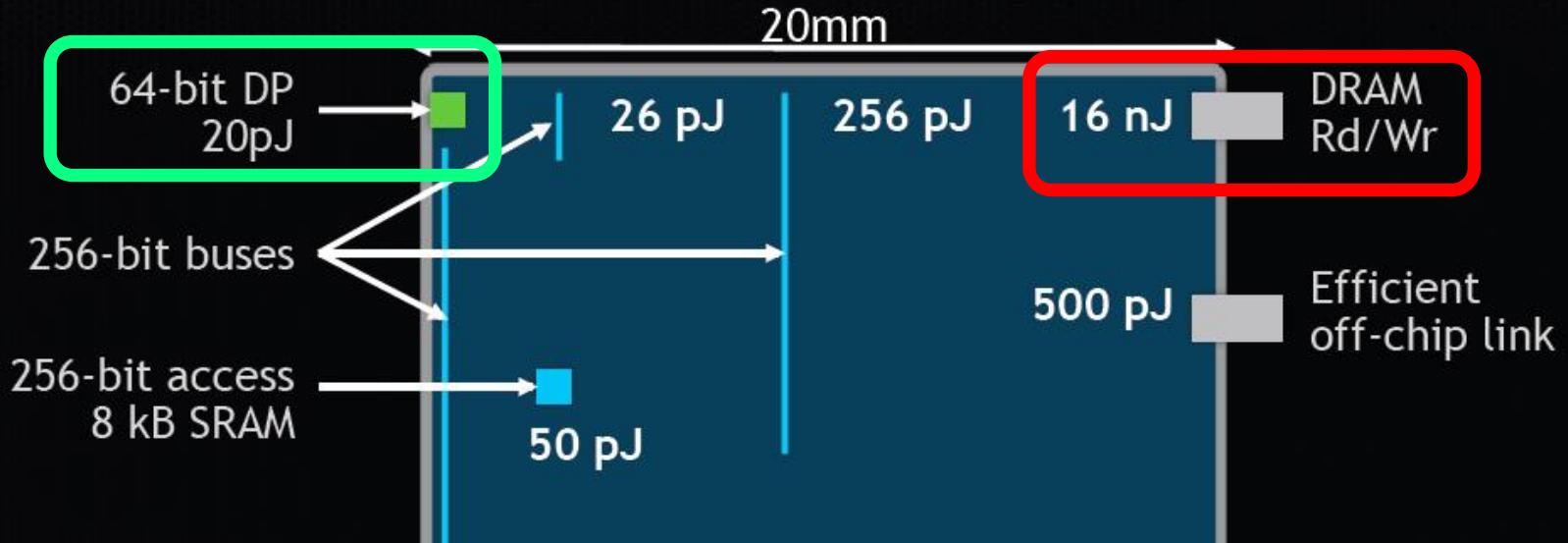
Daehyun Kim^{4,3}

Onur Mutlu^{5,1}

Data Movement vs. Computation Energy

Communication Dominates Arithmetic

Dally, HiPEAC 2015



A memory access consumes \sim 100-1000X
the energy of a complex 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
 - Fundamentally Secure and Dependable Computers
- New Accelerators (Algorithm-Hardware Co-Designs)
 - Artificial Intelligence & Machine Learning
 - Graph Analytics
 - Genome Analysis
- New Memories and Storage Systems
 - Non-Volatile Main Memory
 - Intelligent Memory

Increasingly Demanding Applications

Dream

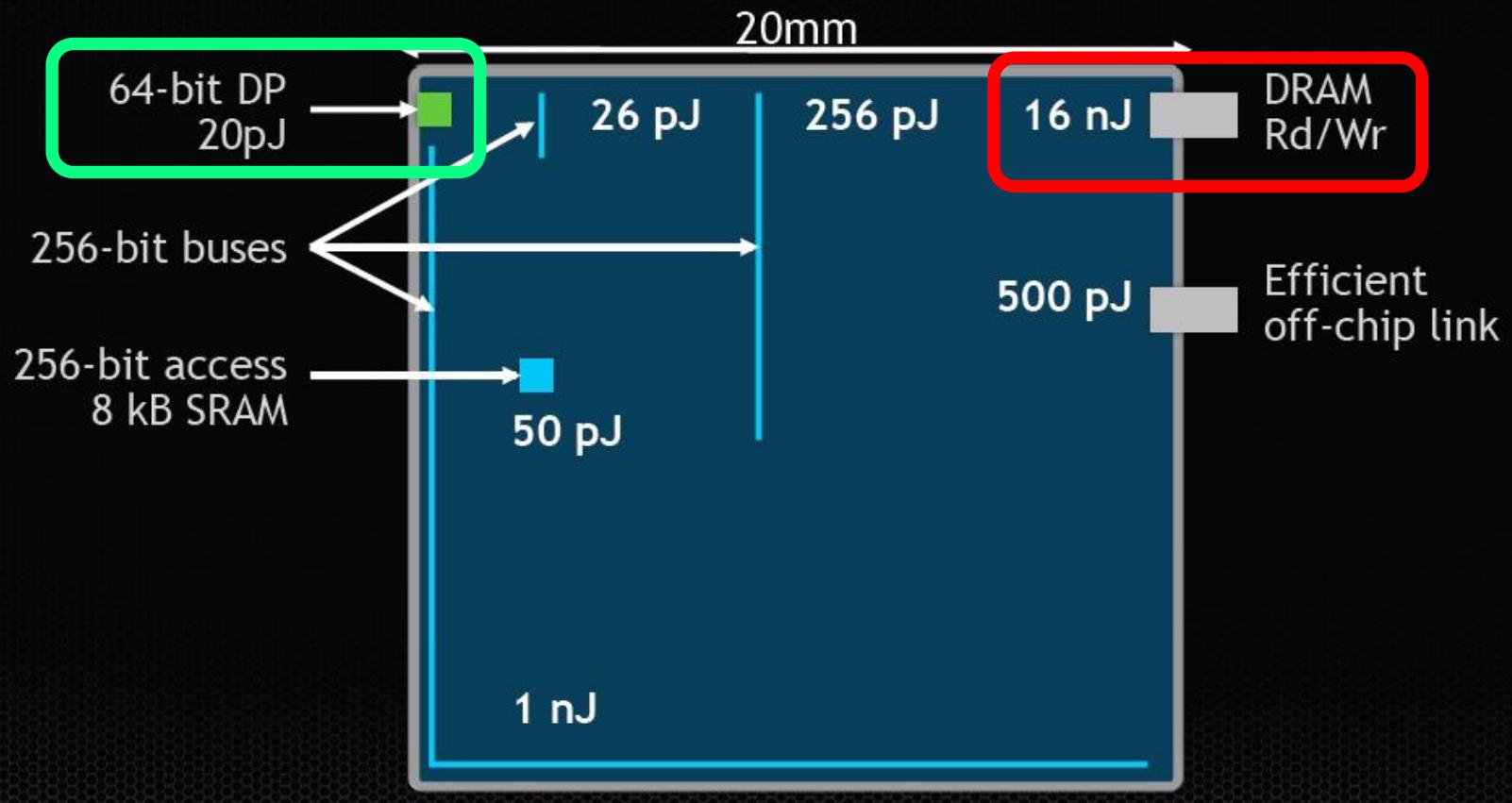
and, they will come

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

Increasingly Diverging/Complex Tradeoffs

Communication Dominates Arithmetic

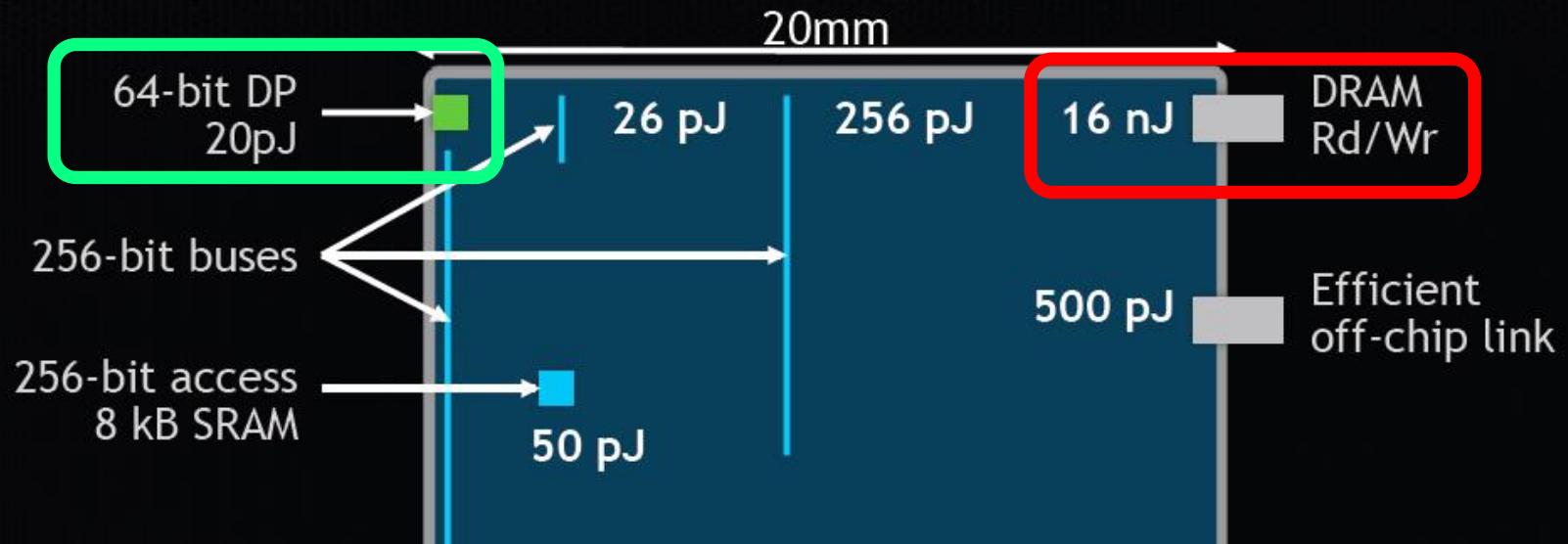
Dally, HiPEAC 2015



Increasingly Diverging/Complex Tradeoffs

Communication Dominates Arithmetic

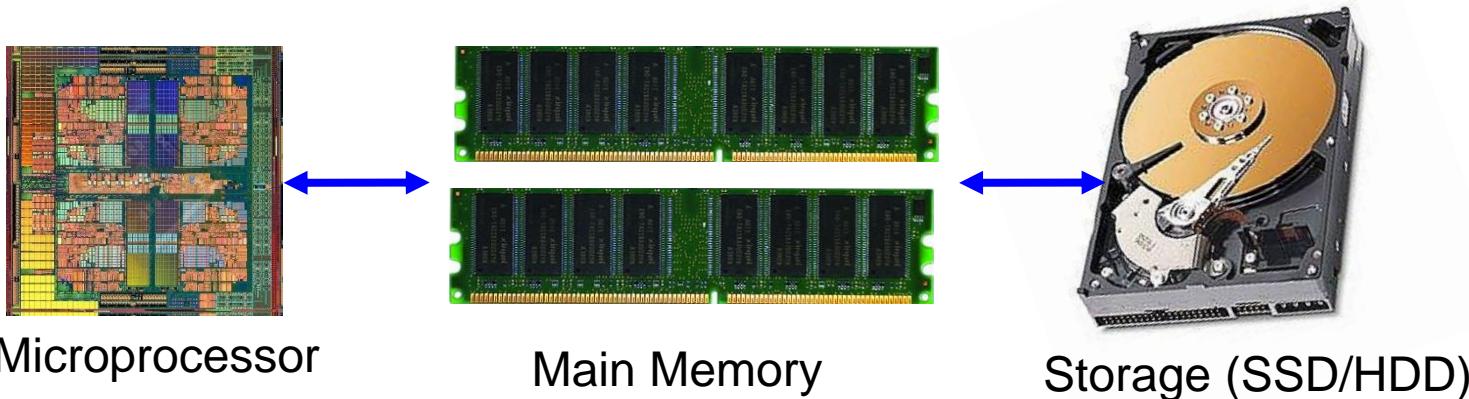
Dally, HiPEAC 2015



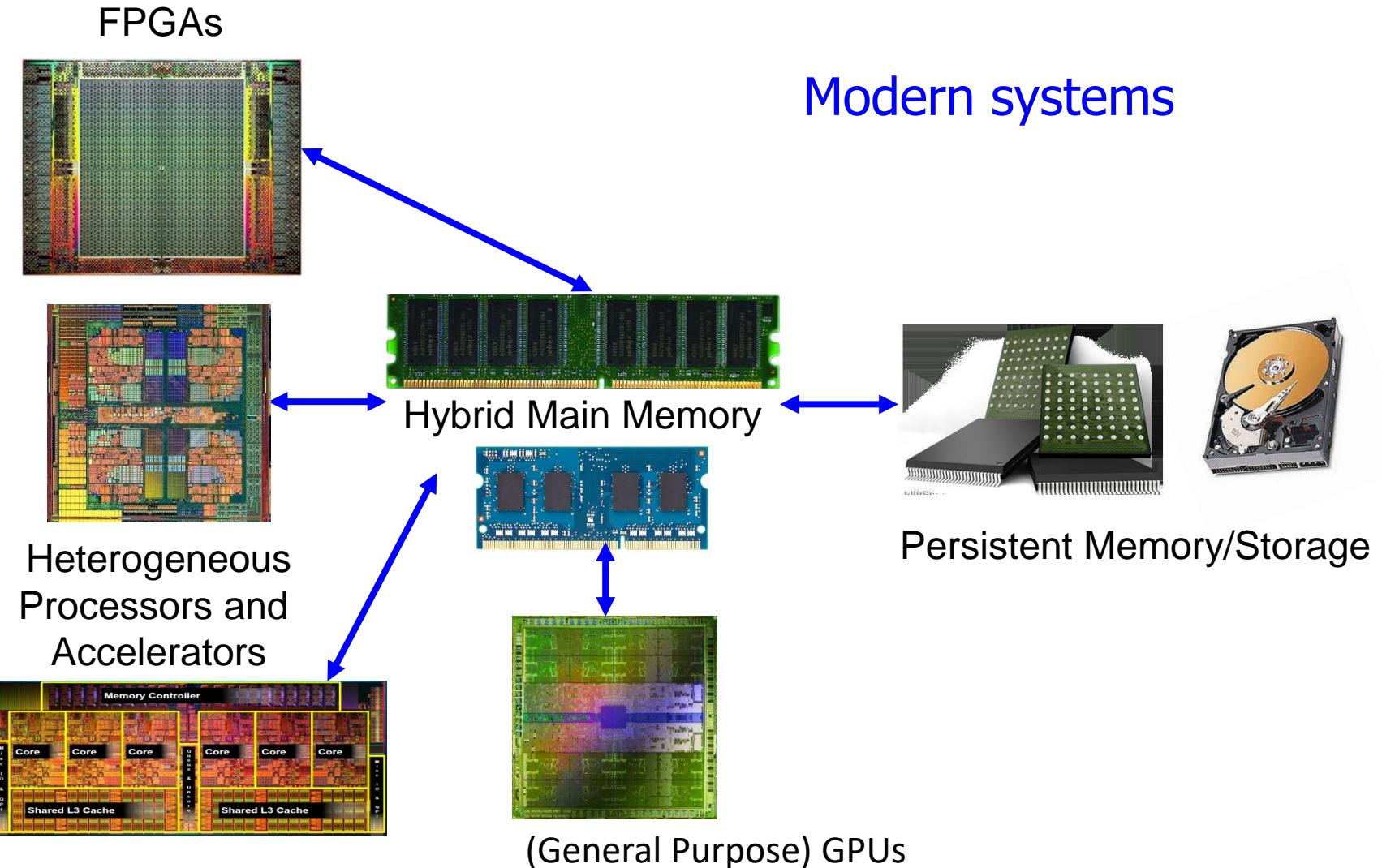
A memory access consumes $\sim 1000\times$ the energy of a complex addition

Increasingly Complex Systems

Past systems

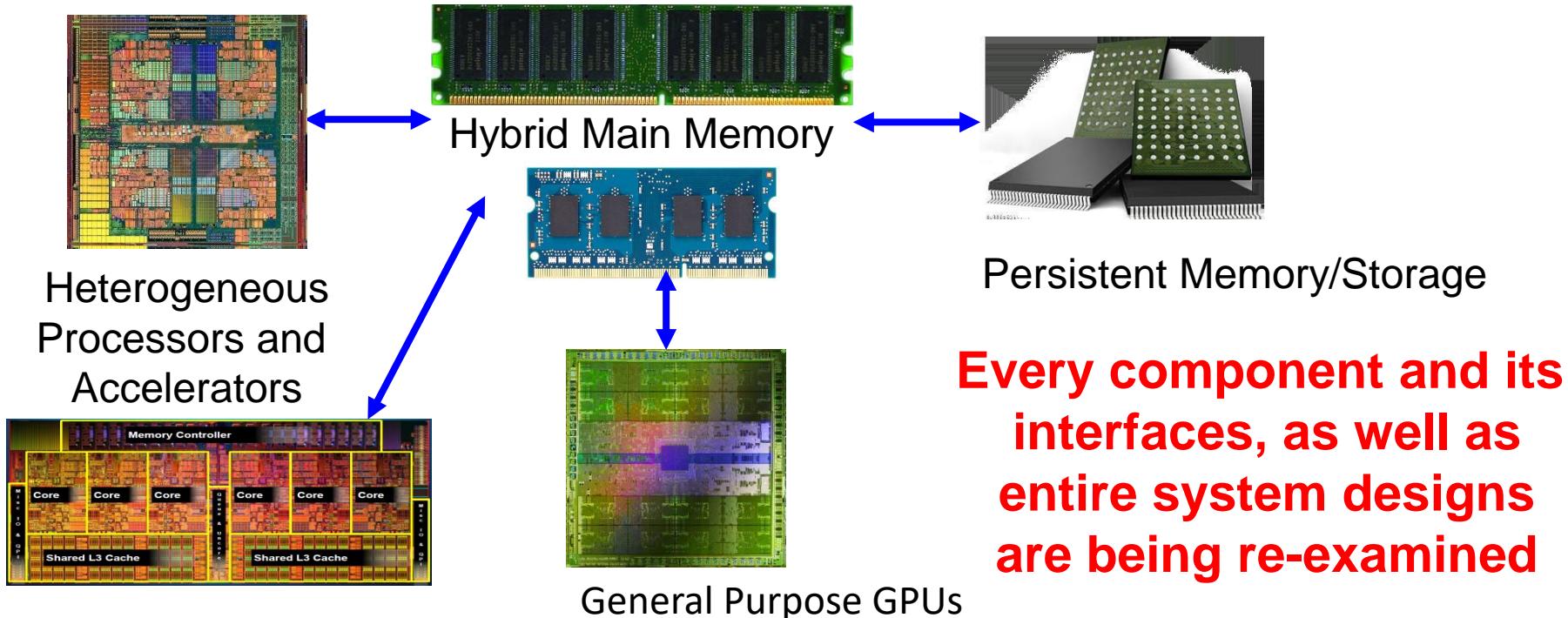


Increasingly Complex Systems



Computer Architecture Today

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



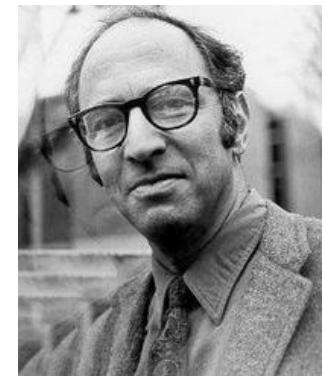
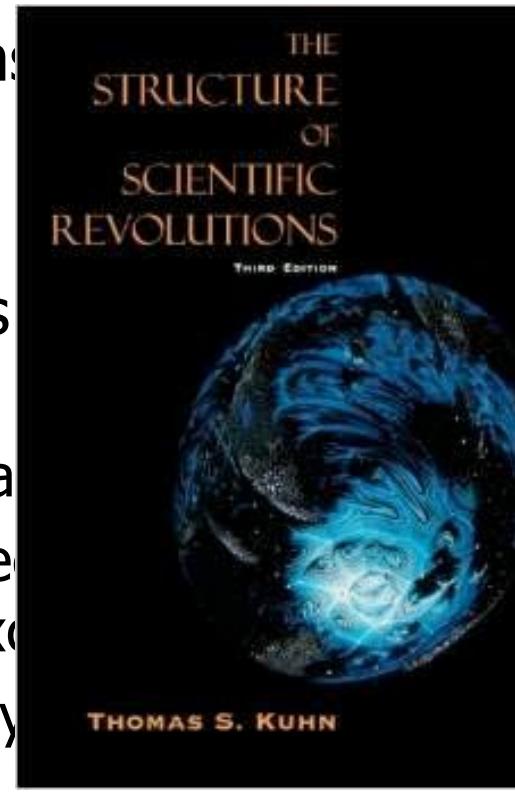
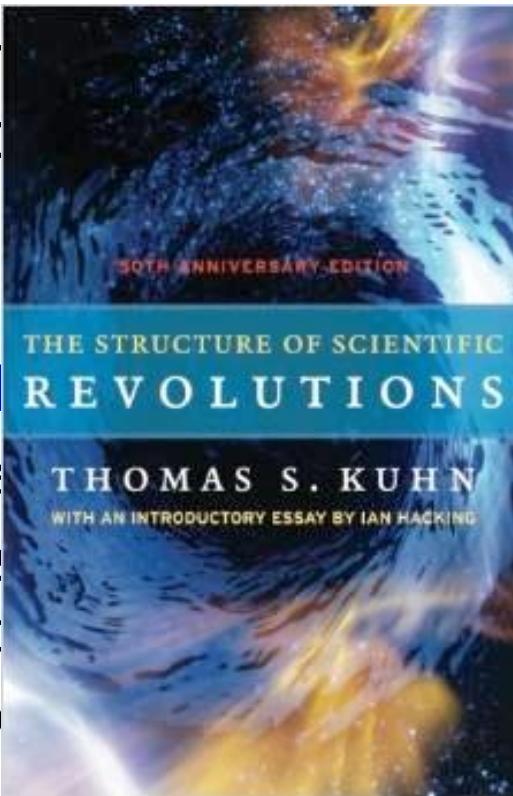
Computer Architecture Today (II)

- 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-paradigmatic science
 - Normal science
 - Things that don't fit
 - Revolution

ture of
field
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 Fundamentals

Question: What Is This?



Answer: The First Major Piece of a Famous Architect

- **Bahnhof Stadelhofen:** "The train station has several of the features that became signatures of his work; straight lines and right angles are rare."
- ETH Alumnus, PhD in Civil Engineering



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 Malmö, Sweden, the Margaret Hunt Hill Bridge in Dallas, Texas, and the Museum of Tomorrow in Rio de Janeiro,

Compare To This



Question 2: What Is This?



Answer: Masterpiece of a Famous Architect

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:

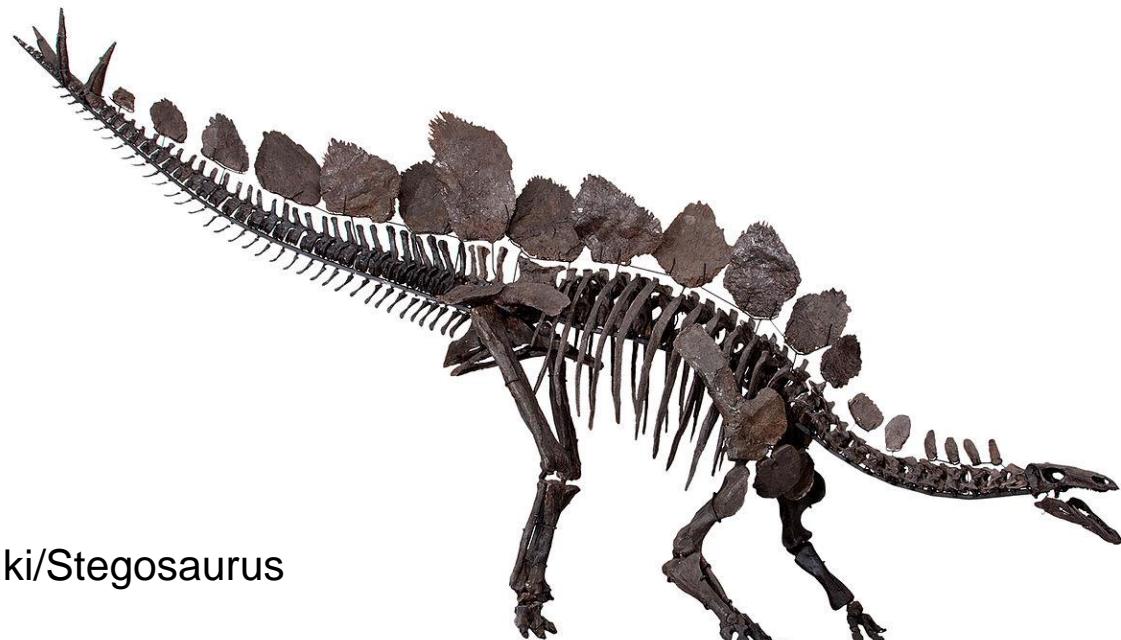
“ In the name of security, Santiago Calatrava's bird has grown a beak. Its ribs have doubled in number and its wings have lost their interstices of glass.... [T]he main transit hall, between Church and Greenwich Streets, will almost certainly lose some of its delicate quality, while gaining structural 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 (/stɛgə'sɔːrəs/[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>

Design Constraints: No one 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

Question: What Is This?





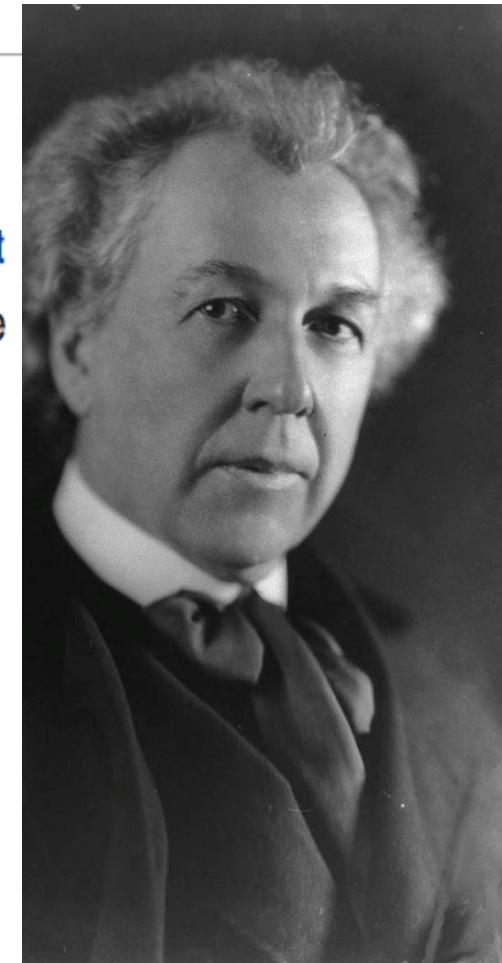
Answer: Masterpiece of Another Famous Architect

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](#).



Your First Comp Arch Assignment

- Go and visit Bahnhof Stadelhofen
 - Extra credit: Repeat for Oculus
 - Extra+ credit: Repeat for Fallingwater
- Appreciate the beauty & out-of-the-box and creative thinking
- Think about tradeoffs in the design of the Bahnhof
 - Strengths, weaknesses, goals of design
- Derive principles on your own for good design and innovation
- Due date: **Any time during 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

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
 - ...
-
- How to evaluate goodness of design is always a critical question.
-

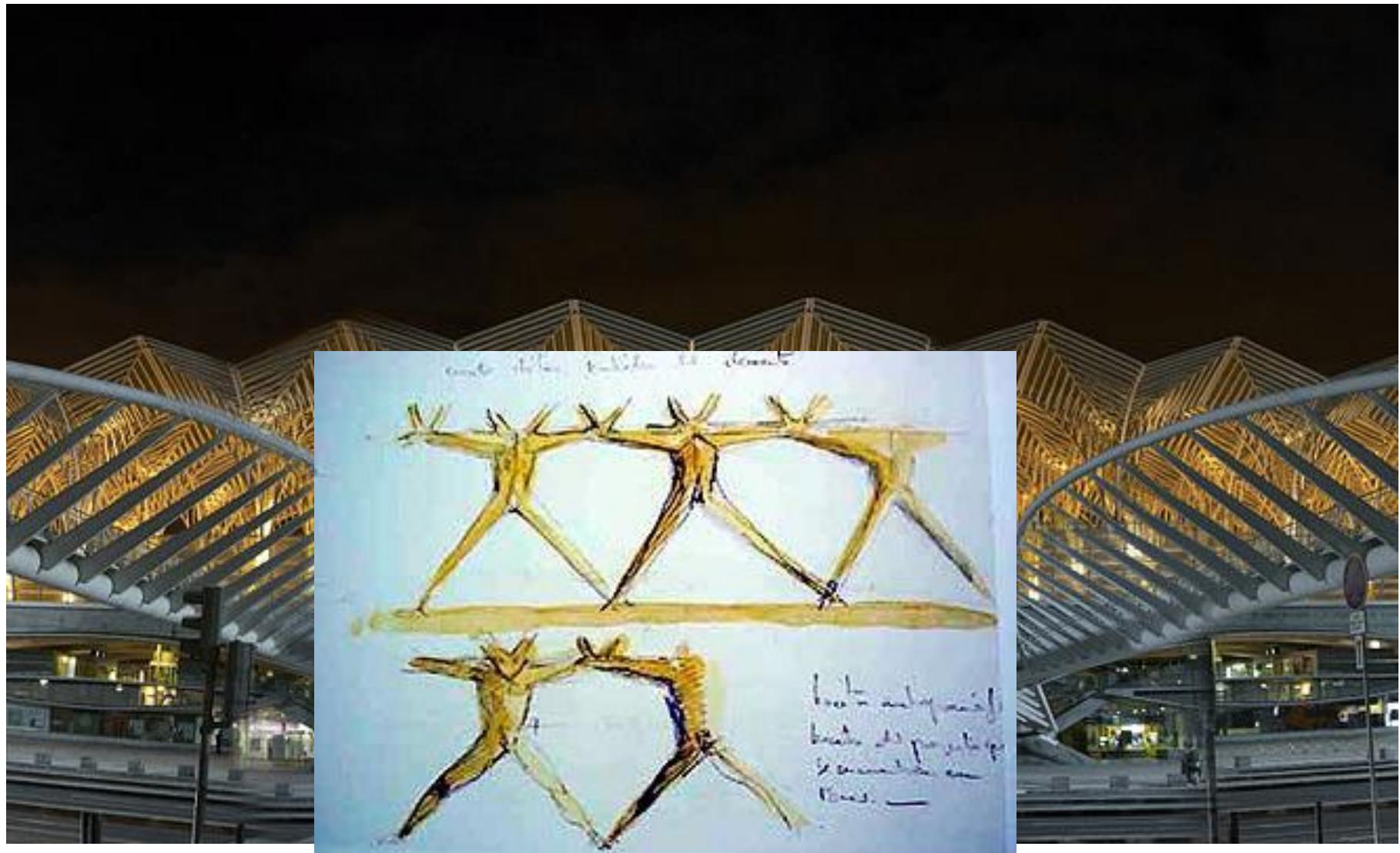
A Key Question

- How was Calavatra able to design especially his key buildings?
 - Can have many guesses
 - (Ultra) 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?



What About This?



Milwaukee Art Museum



Athens Olympic Stadium



City of Arts and Sciences, Valencia



Florida Polytechnic University (I)



Oculus, New York City



A Quote from The Other Famous Architect

- “architecture [...] based upon principle, and not upon precedent” (Frank Lloyd Wright)



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.

Another View



Yet Another View





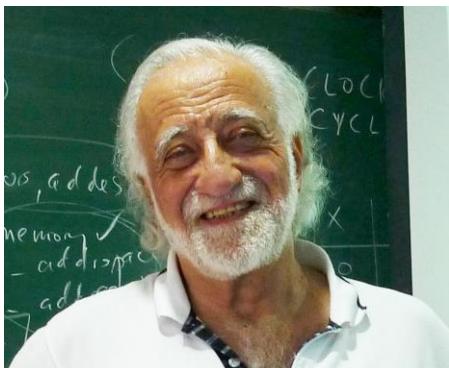
Major High-Level Goals of This Course

- Understand the principles
- Understand the precedents
- Based on such understanding:
 - Enable you to evaluate tradeoffs of different designs and ideas
 - Enable you to develop principled designs
 - Enable you to develop novel, out-of-the-box designs
- The focus is on:
 - Principles, precedents, and how to use them for new designs
- In Computer Architecture

Role of the (Computer) Architect

Role of the Architect

- ***Look Backward (Examine old code)***
- ***Look forward (Listen to the dreamers)***
- ***Look Up (Nature of the problems)***
- ***Look Down (Predict the future of technology)***



from Yale Patt's lecture notes

Role of The (Computer) Architect

- Look backward (to the past)
 - Understand tradeoffs and designs, upsides/downsides, past workloads. Analyze and evaluate the past.
- Look forward (to the future)
 - Be the dreamer and create new designs. Listen to dreamers.
 - Push the state of the art. Evaluate new design choices.
- Look up (towards problems in the computing stack)
 - Understand important problems and their nature.
 - Develop architectures and ideas to solve important problems.
- Look down (towards device/circuit technology)
 - Understand the capabilities of the underlying technology.
 - Predict and adapt to the future of technology (you are designing for N years ahead). Enable the future technology.

Takeaways

- Being an architect is not easy
 - You need to consider **many** things in designing a new system + have good intuition/insight into ideas/tradeoffs
 - But, it is fun and can be very rewarding
 - And, enables a great future
 - E.g., many scientific and everyday-life innovations would not have been possible without architectural innovation that enabled very high performance systems
 - E.g., your mobile phones
 - E.g., self-driving vehicles
 - This course will enable you to become a good computer architect
-

So, I Hope You Are Here for This

Comp. Systems

- 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?

“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*

Digital Design

HW designer’s view of how a computer system works
Digital logic as a model of computation

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

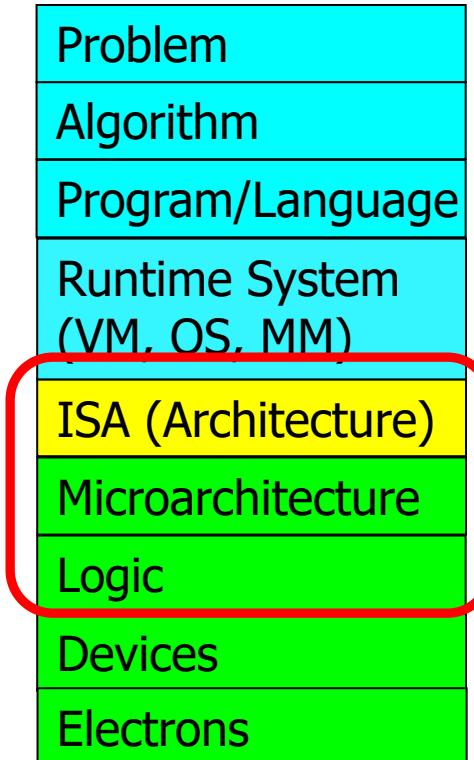
Building blocks of micro-arch (e.g., gates)



ISA
(Instruction Set Architecture)

Interface/contract between SW and HW.

What the programmer assumes hardware will satisfy.

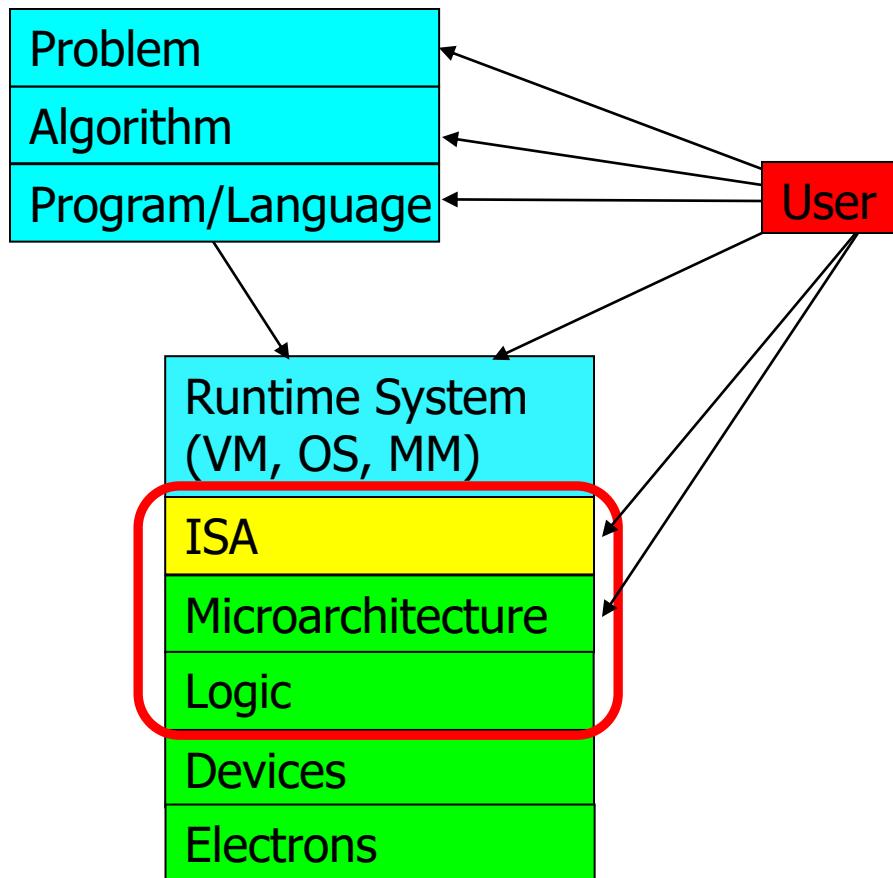


Aside: An Important Work By Hamming

- Hamming, "Error Detecting and Error Correcting Codes," Bell System Technical Journal 1950.
- Introduced the concept of Hamming distance
 - number of locations in which the corresponding symbols of two equal-length strings is different
- Developed a theory of codes used for error detection and correction
- Also see:
 - Hamming, "You and Your Research," Talk at Bell Labs, 1986.
 - <http://www.cs.virginia.edu/~robins/YouAndYourResearch.html>

Levels of Transformation, Revisited

- A user-centric view: computer designed for users



- The entire stack should be optimized for user

The Power of Abstraction

- **Levels of transformation create abstractions**
 - Abstraction: A higher level only needs to know about the interface to the lower level, not how the lower level is implemented
 - E.g., high-level language programmer does not really need to know what the ISA is and how a computer executes instructions
- **Abstraction improves productivity**
 - No need to worry about decisions made in underlying levels
 - E.g., programming in Java vs. C vs. assembly vs. binary vs. by specifying control signals of each transistor every cycle
- Then, why would you want to know what goes on underneath or above?

Crossing the Abstraction Layers

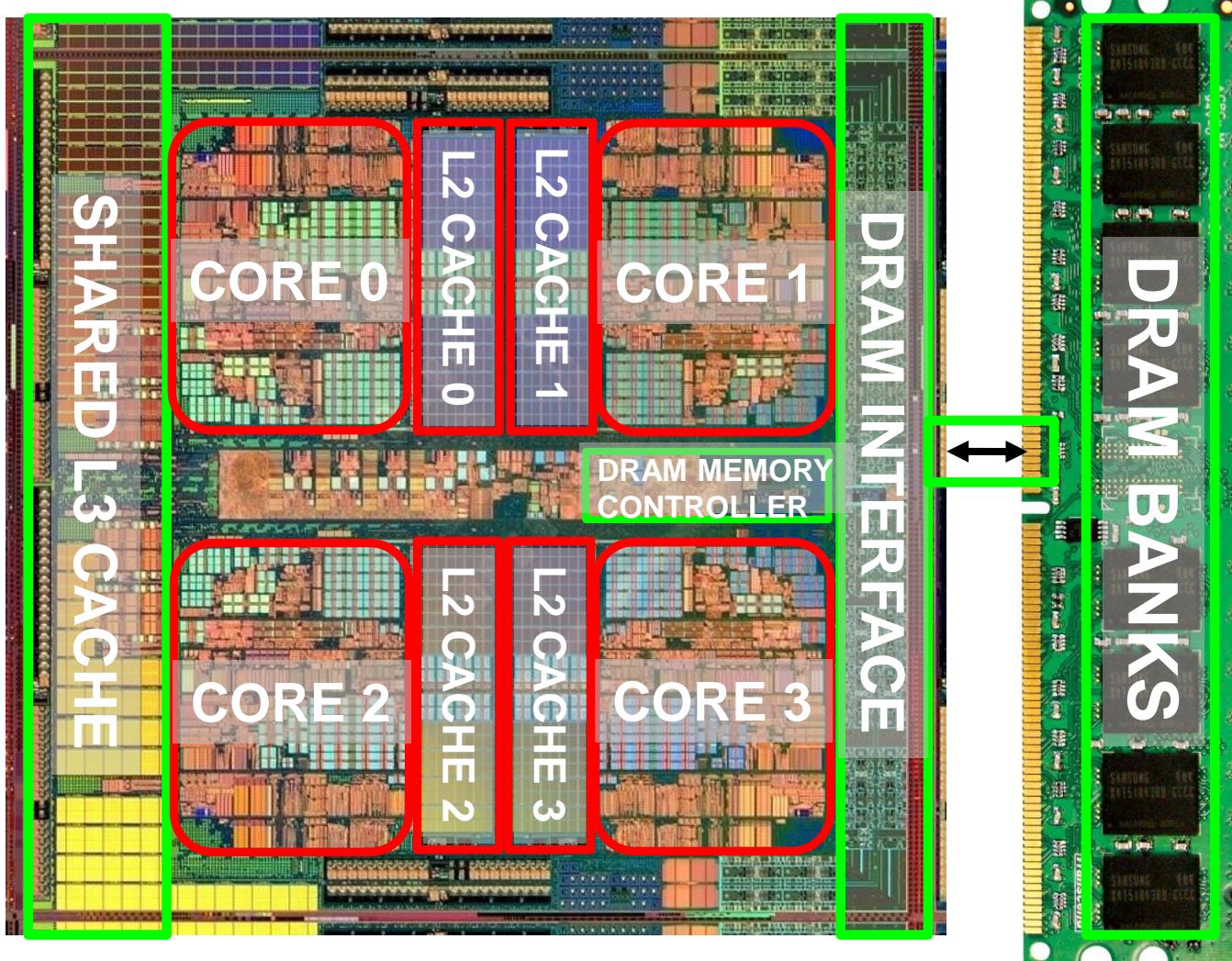
- As long as everything goes well, not knowing what happens underneath (or above) is not a problem.
- What if
 - ❑ The program you wrote is running slow?
 - ❑ The program you wrote does not run correctly?
 - ❑ The program you wrote consumes too much energy?
 - ❑ Your system just shut down and you have no idea why?
 - ❑ Someone just compromised your system and you have no idea how?
- What if
 - ❑ The hardware you designed is too hard to program?
 - ❑ The hardware you designed is too slow because it does not provide the right primitives to the software?
- What if
 - ❑ You want to design a much more efficient and higher performance system?

Crossing the Abstraction Layers

- Two key goals of this course are
 - to understand how a processor works underneath the software layer and how decisions made in hardware affect the software/programmer
 - to enable you to be comfortable in making design and optimization decisions that cross the boundaries of different layers and system components

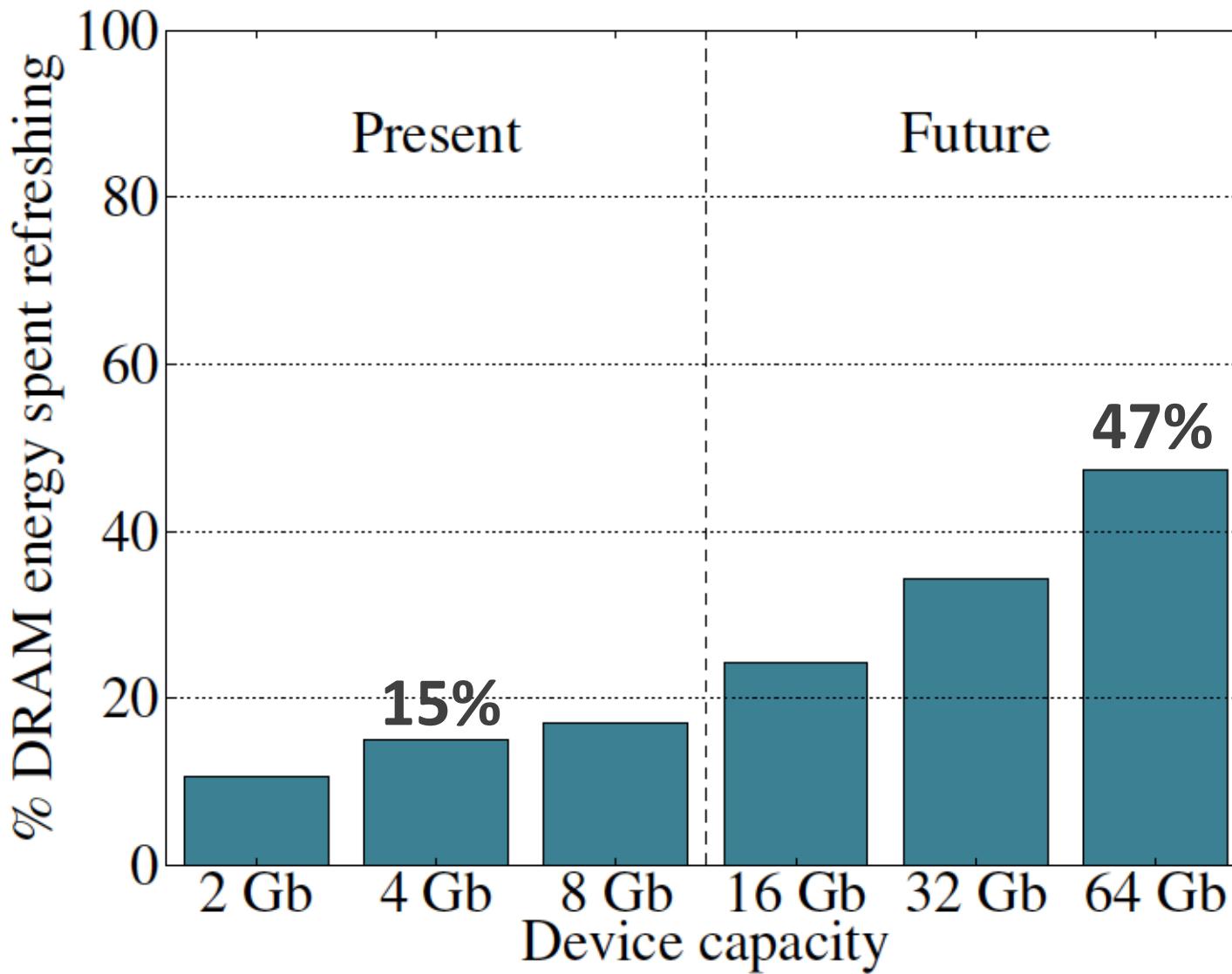
An Example: Multi-Core Systems

Multi-Core
Chip



*Die photo credit: AMD Barcelona

Another Example: Memory Refresh



Computer Architecture

Lecture 1: Introduction and Basics

Prof. Onur Mutlu

ETH Zürich

Fall 2023

28 September 2023