

# Memory-Centric Computing

Onur Mutlu

[omutlu@gmail.com](mailto:omutlu@gmail.com)

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

31 January 2024

UT-Austin ECE Colloquium

**SAFARI**

**ETH** zürich

Computing

is Bottlenecked by Data



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

---

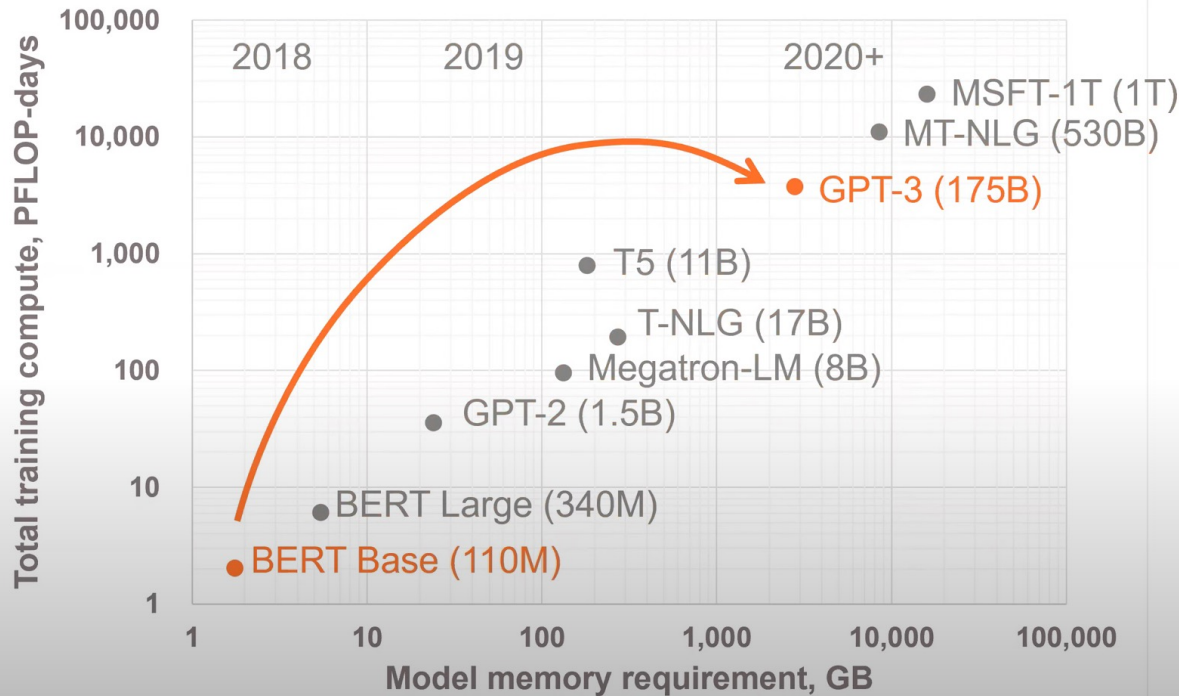
- Important workloads are all data intensive
- They require rapid and efficient processing of large amounts of data
- Data is increasing
  - We can generate more than we can process
  - We need to perform more sophisticated analyses on more data

# Huge Demand for Performance & Efficiency

## Exponential Growth of Neural Networks



Memory and compute requirements



**1800x more compute**  
In just **2 years**

Tomorrow, **multi-trillion** parameter models

# Data is Key for Future Workloads

---



## In-memory Databases

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



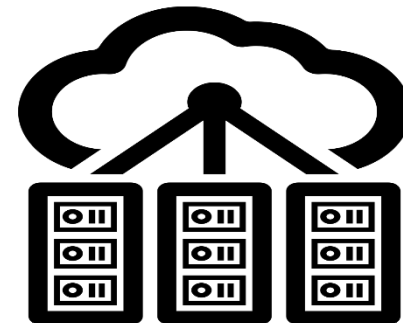
## In-Memory Data Analytics

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



## Graph/Tree Processing

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



## Datacenter Workloads

[Kanev+ (Google), ISCA'15]

# Data Overwhelms Modern Machines

---



**In-memory Databases**



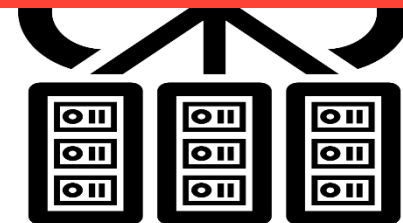
**Graph/Tree Processing**

**Data → performance & energy bottleneck**



**In-Memory Data Analytics**

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



**Datacenter Workloads**

[Kanev+ (Google), ISCA'15]

# Data is Key for Future Workloads



**Chrome**

Google's web browser



**TensorFlow Mobile**

Google's machine learning  
framework



**Video Playback**

Google's **video codec**



**Video Capture**

Google's **video codec**

# Data Overwhelms Modern Machines



**Chrome**



**TensorFlow Mobile**

Data → performance & energy bottleneck

**VP9**



**Video Playback**

Google's **video codec**

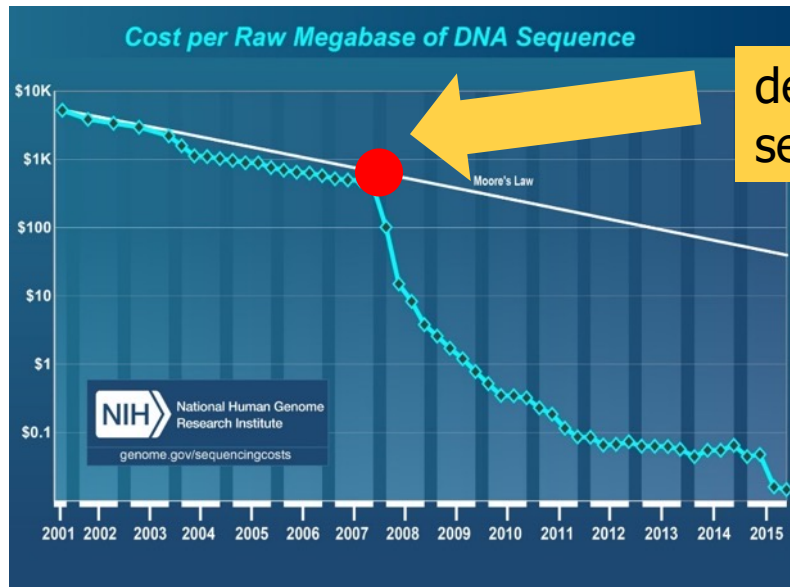
**VP9**



**Video Capture**

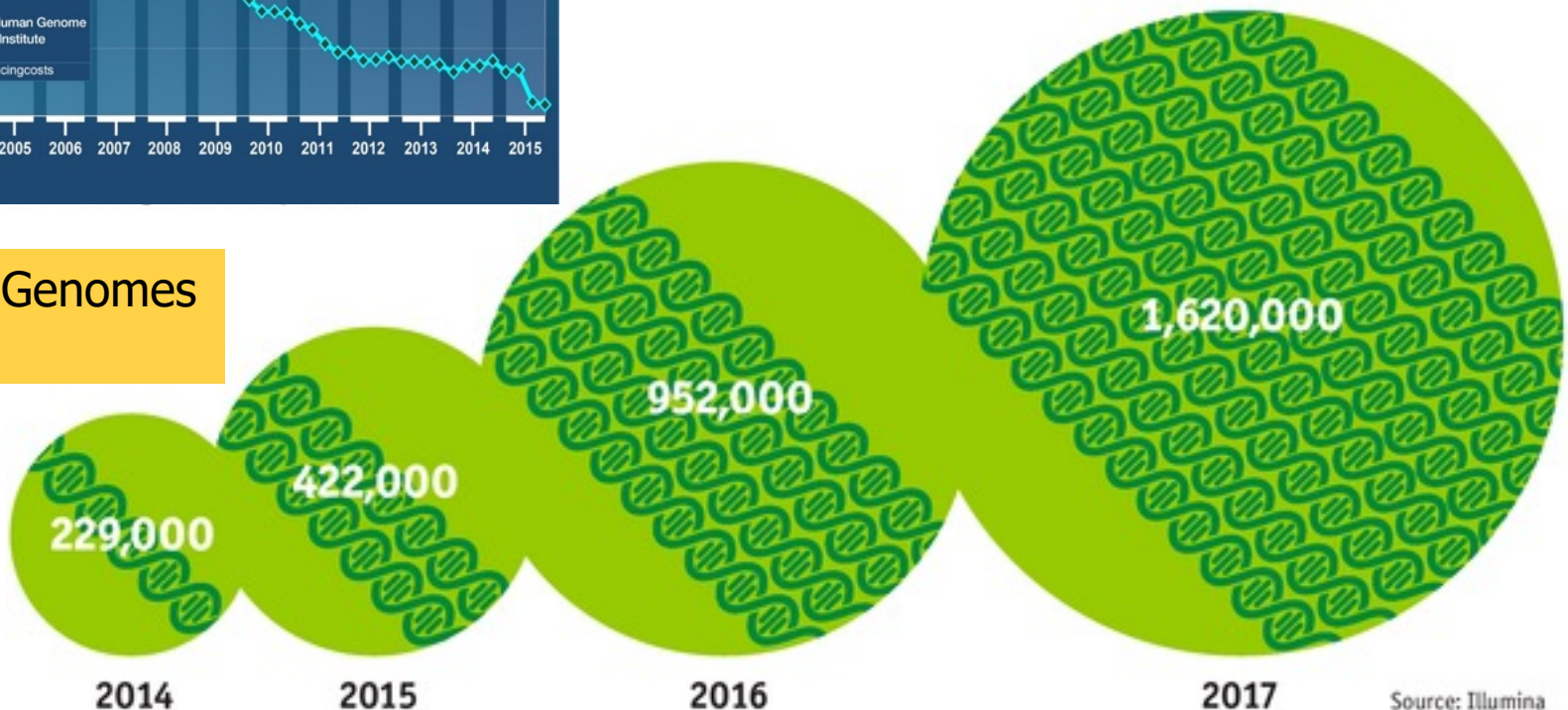
Google's **video codec**

# Data is Key for Future Workloads



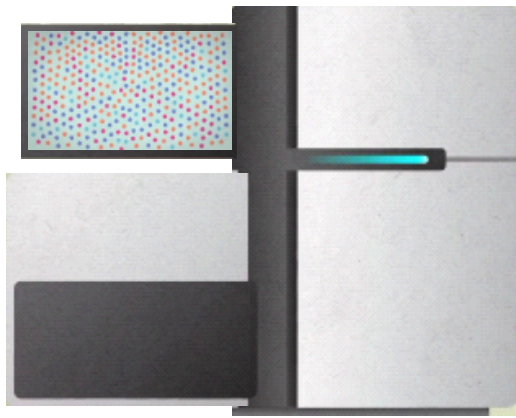
development of high-throughput sequencing (HTS) technologies

Number of Genomes Sequenced



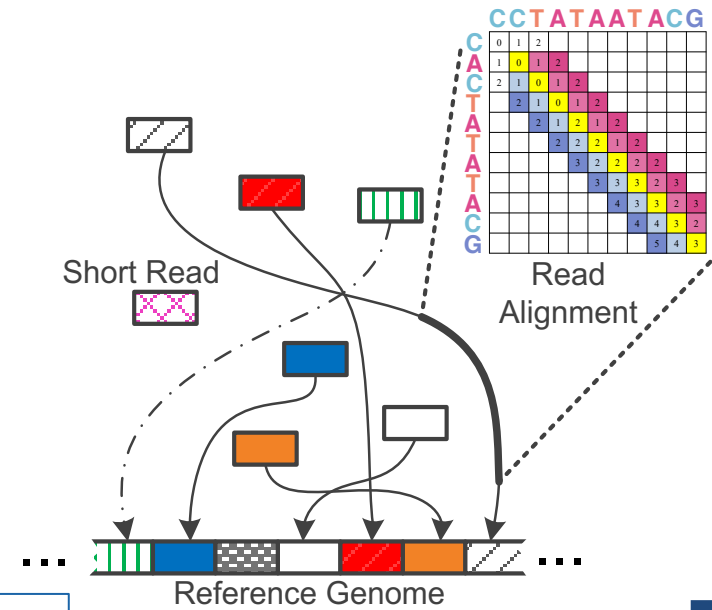
The Economist





Billions of Short Reads

ATATATACGTACTAGTACGT  
 TTTAGTACGTACGT  
 ATACGTACTAGTACGT  
 CGCCCCTACGTA  
 ACGTACTAGTACGT  
 TTAGTACGTACGT  
 TACGTACTAAAGTACGT  
 TACGTACTAGTACGT  
 TTTAAACGTA  
 CGTACTAGTACGT  
 GGGAGTACGTACGT



## 1 Sequencing

# Genome Analysis

## 2 Read Mapping

Data → performance & energy bottleneck

read4: CGCTTCCAT  
 read5: CCATGACGC  
 read6: TTCCATGAC



## 3 Variant Calling

## 4 Scientific Discovery



# We Need Faster & Scalable Genome Analysis



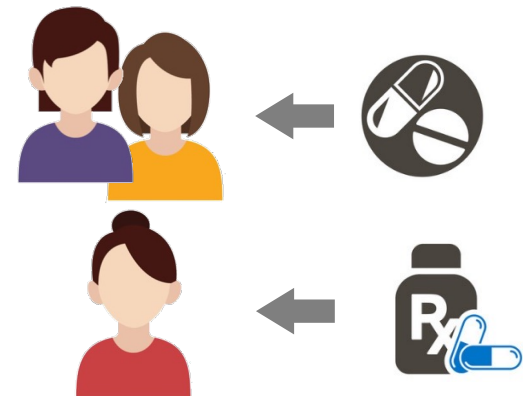
Understanding **genetic variations**,  
**species**, **evolution**, ...



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



Rapid surveillance of **disease outbreaks**



Developing **personalized medicine**

# 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

Senol Cali+, “**Nanopore Sequencing Technology and Tools for Genome Assembly: Computational Analysis of the Current State, Bottlenecks and Future Directions**,” *Briefings in Bioinformatics*, 2018.

[[Open arxiv.org version](#)]

# 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** ▼

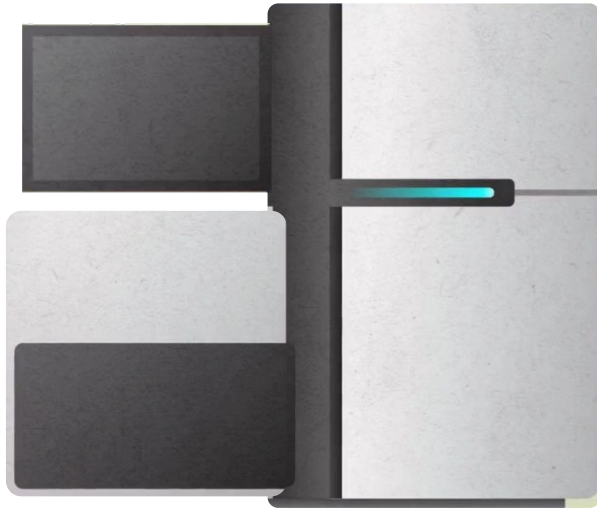


Oxford Nanopore MinION

Data → performance & energy bottleneck

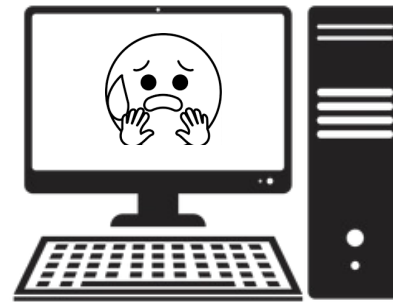
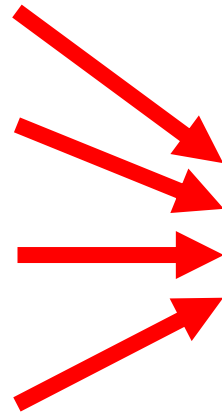
# Problems with (Genome) Analysis Today

---



**Special-Purpose** Machine  
for **Data Generation**

**FAST**



**General-Purpose** Machine  
for **Data Analysis**

**SLOW**

**Slow and inefficient processing capability**  
**Large amounts of data movement**

# Accelerating Genome Analysis [DAC 2023]

---

- Onur Mutlu and Can Firtina,  
**"Accelerating Genome Analysis via Algorithm-Architecture Co-Design"**  
*Invited Special Session Paper in Proceedings of the 60th Design Automation Conference (DAC), San Francisco, CA, USA, July 2023.*  
[\[arXiv version\]](#)

## Accelerating Genome Analysis via Algorithm-Architecture Co-Design

Onur Mutlu   Can Firtina  
*ETH Zürich*

# Accelerating Genome Analysis [IEEE MICRO 2020]

---

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

## Accelerating Genome Analysis: A Primer on an Ongoing Journey

**Mohammed Alser**

ETH Zürich

**Zülal Bingöl**

Bilkent University

**Damla Senol Cali**

Carnegie Mellon University

**Jeremie Kim**

ETH Zurich and Carnegie Mellon University

**Saugata Ghose**

University of Illinois at Urbana-Champaign and  
Carnegie Mellon University

**Can Alkan**

Bilkent University

**Onur Mutlu**

ETH Zurich, Carnegie Mellon University, and  
Bilkent University



# Beginner Reading on Genome Analysis

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

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

Computational and Structural Biotechnology Journal, 2022

[[Source code](#)]



ELSEVIER

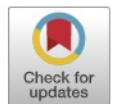


journal homepage: [www.elsevier.com/locate/csbj](http://www.elsevier.com/locate/csbj)



Review

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



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

ETH Zurich, Gloriastrasse 35, 8092 Zürich, Switzerland

**SAFARI**

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

# Data Overwhelms Modern Machines ...

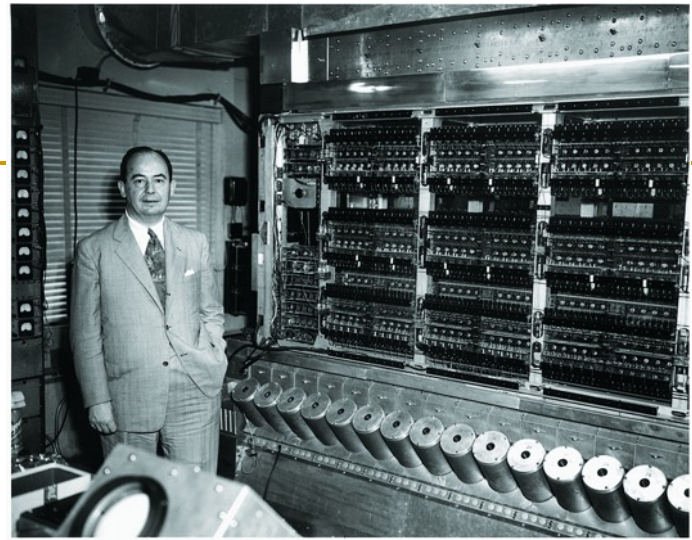
---

- Storage/memory capability
- Communication capability
- Computation capability
- Greatly impacts robustness, energy, performance, cost

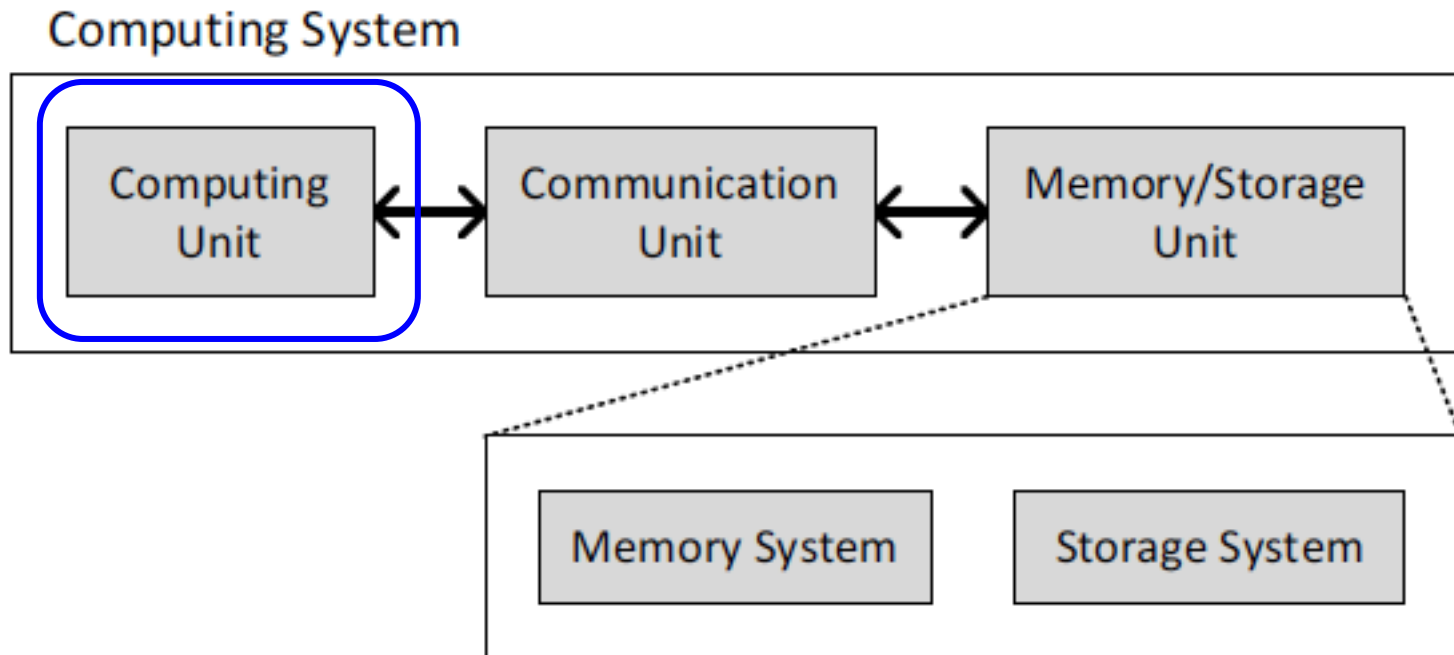


# A Computing System

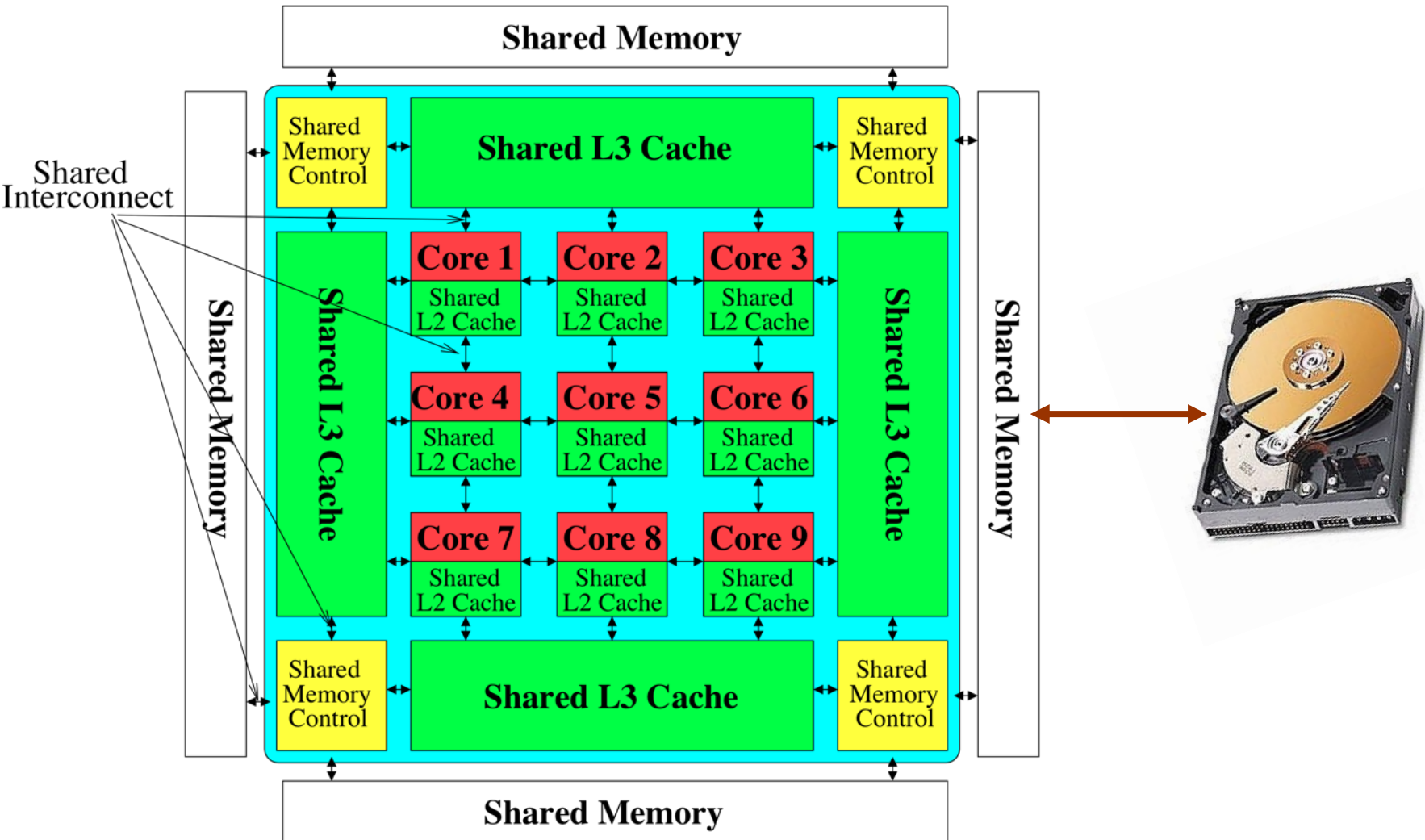
- Three key components
- Computation
- Communication
- Storage/memory



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



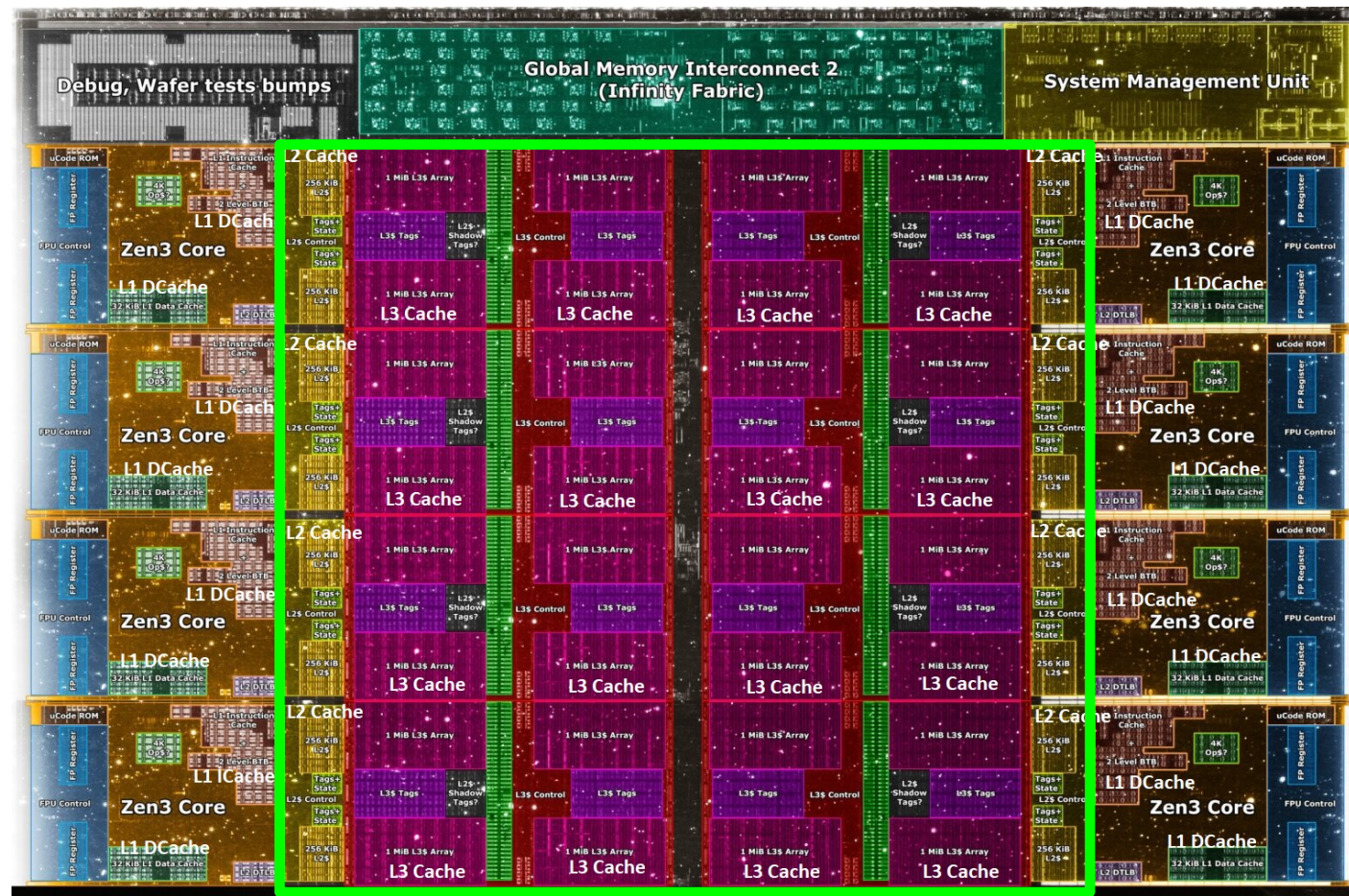
# Perils of Processor-Centric Design



**Most of the system is dedicated to storing and moving data**

**Yet, system is still bottlenecked by memory & storage**

# Deeper and Larger Memory Hierarchies



Core Count:  
8 cores/16 threads

L1 Caches:  
32 KB per core

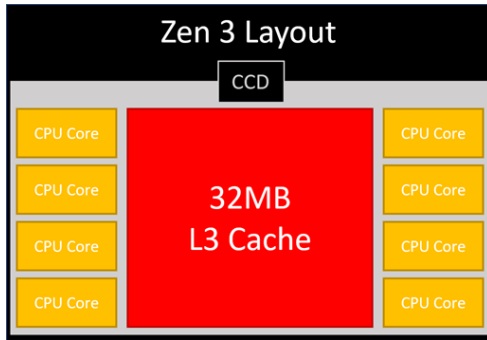
L2 Caches:  
512 KB per core

L3 Cache:  
32 MB shared

AMD Ryzen 5000, 2020



# AMD's 3D Last Level Cache (2021)

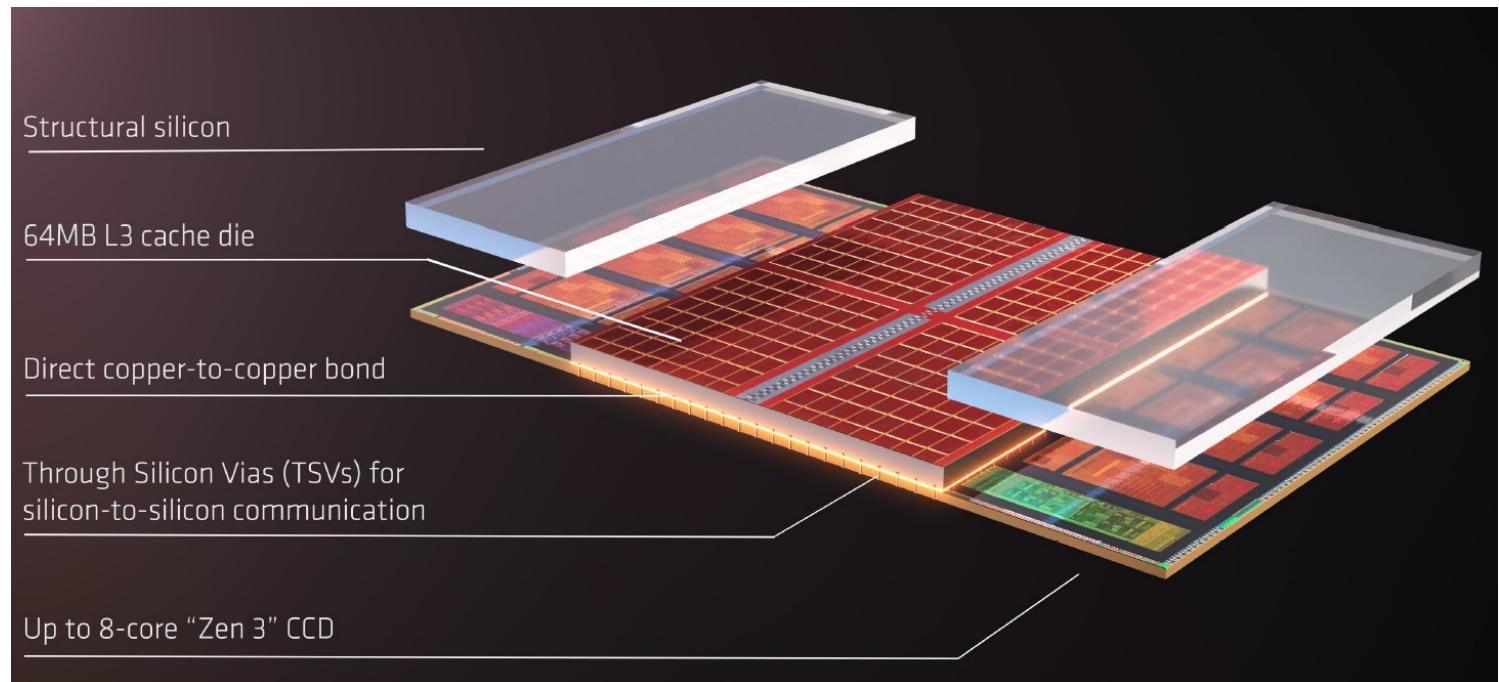


<https://community.microcenter.com/discussion/5134/comparing-zen-3-to-zen-2>

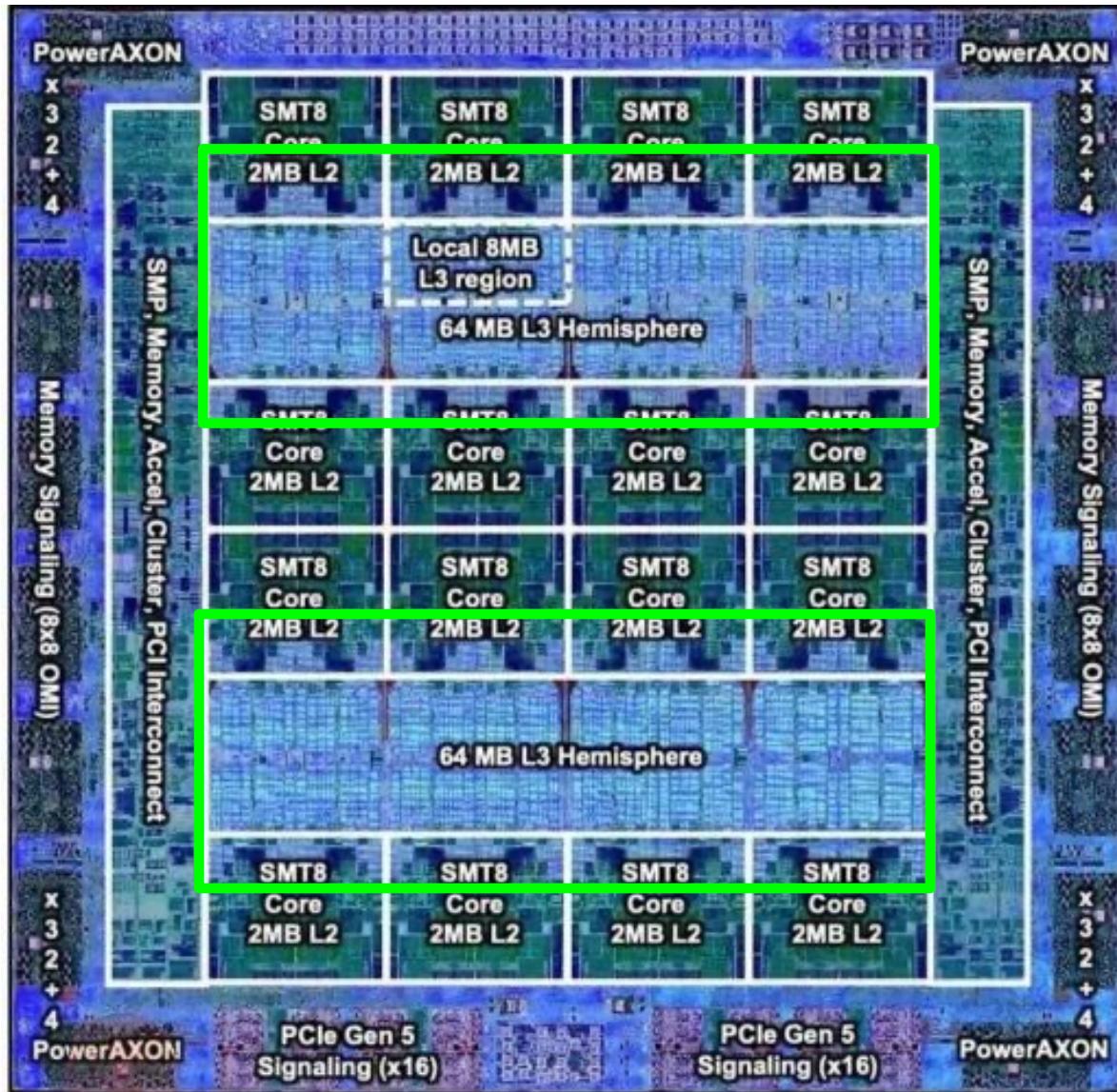
AMD increases the L3 size of their 8-core Zen 3 processors from 32 MB to 96 MB

**Additional 64 MB L3 cache die**  
**stacked on top of the processor die**

- Connected using Through Silicon Vias (TSVs)
- Total of 96 MB L3 cache



# Deeper and Larger Memory Hierarchies



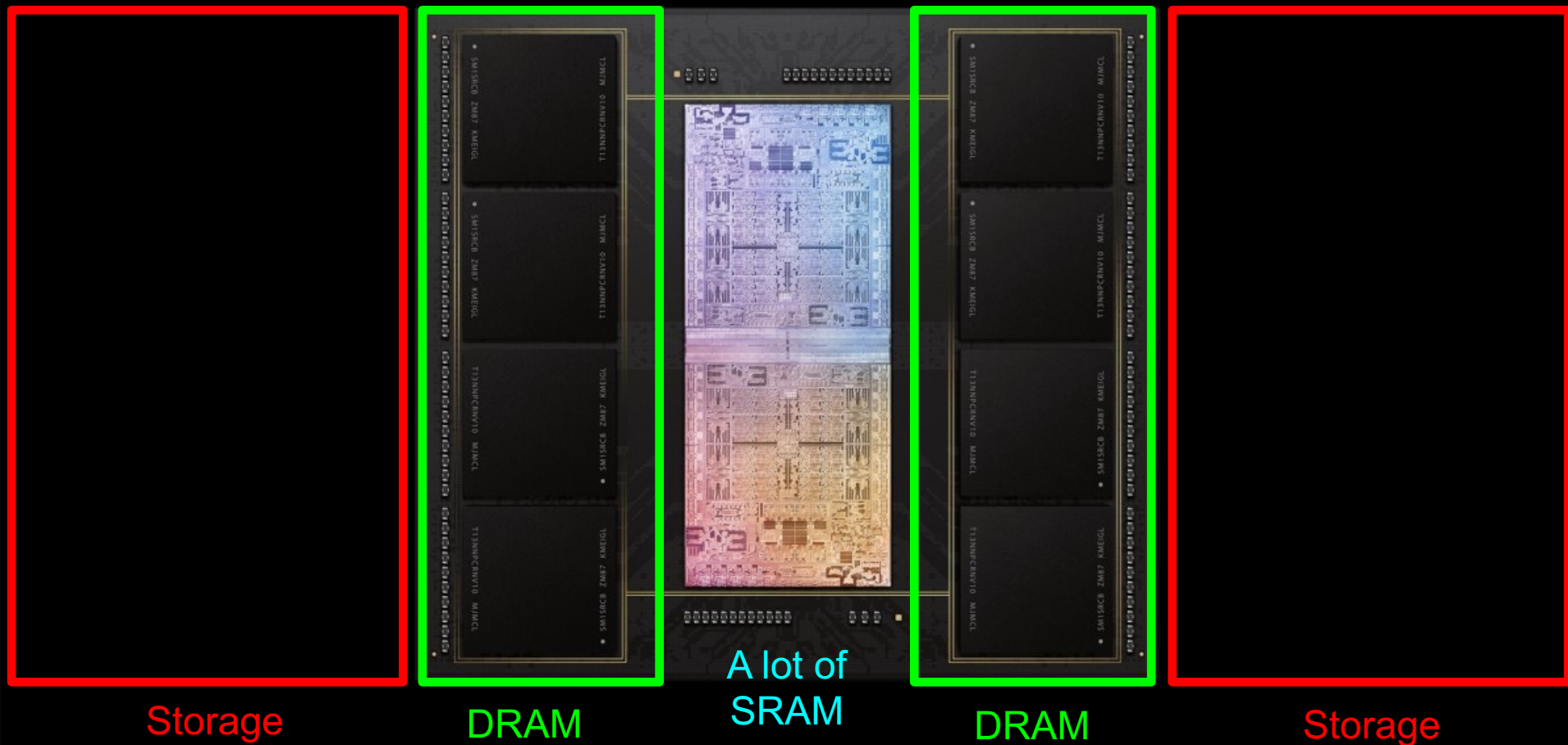
IBM POWER10,  
2020

Cores:  
15-16 cores,  
8 threads/core

L2 Caches:  
2 MB per core

L3 Cache:  
120 MB shared

# Deeper and Larger Memory Hierarchies



Apple M1 Ultra System (2022)



# Data Overwhelms Modern Machines



**Chrome**



**TensorFlow Mobile**

Data → performance & energy bottleneck

**VP9**



**Video Playback**

Google's **video codec**

**VP9**



**Video Capture**

Google's **video codec**



# Data Movement Overwhelms Modern Machines

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

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

## Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks

Amirali Boroumand<sup>1</sup>

Saugata Ghose<sup>1</sup>

Youngsok Kim<sup>2</sup>

Rachata Ausavarungnirun<sup>1</sup>

Eric Shiu<sup>3</sup>

Rahul Thakur<sup>3</sup>

Daehyun Kim<sup>4,3</sup>

Aki Kuusela<sup>3</sup>

Allan Knies<sup>3</sup>

Parthasarathy Ranganathan<sup>3</sup>

Onur Mutlu<sup>5,1</sup>

# Data Movement Overwhelms Accelerators

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

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

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

Amirali Boroumand<sup>†◇</sup>  
Geraldo F. Oliveira<sup>\*</sup>

Saugata Ghose<sup>‡</sup>  
Xiaoyu Ma<sup>§</sup>

Berkin Akin<sup>§</sup>  
Eric Shiu<sup>§</sup>

Ravi Narayanaswami<sup>§</sup>  
Onur Mutlu<sup>\*†</sup>

<sup>†</sup>Carnegie Mellon Univ.    <sup>◇</sup>Stanford Univ.    <sup>‡</sup>Univ. of Illinois Urbana-Champaign    <sup>§</sup>Google    <sup>\*</sup>ETH Zürich

## An Intelligent Architecture Handles Data Well

# How to Handle Data Well

---

- **Ensure data does not overwhelm** the components
  - via intelligent algorithms, architectures & system designs: algorithm-architecture-devices
- **Take advantage of** vast amounts of **data** and metadata
  - to improve architectural & system-level decisions
- **Understand and exploit** properties of (different) **data**
  - to improve algorithms & architectures in various metrics

# Corollaries: Computing Systems Today ...

---

- Are **processor-centric** vs. **data-centric**
- Make **designer-dictated** decisions vs. **data-driven**
- Make **component-based myopic** decisions vs. **data-aware**



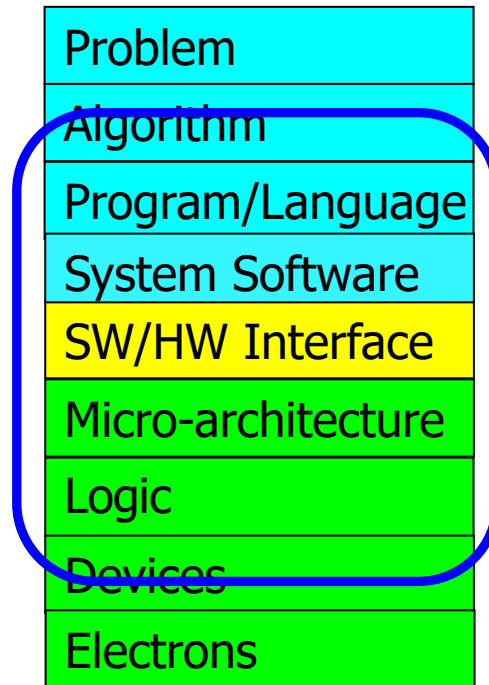
**Data-centric**

**Data-driven**

**Data-aware**

# We Need to Revisit the Entire Stack

---



**We can get there step by step**

# A Blueprint for Fundamentally Better Architectures

---

- Onur Mutlu,  
**"Intelligent Architectures for Intelligent Computing Systems"**  
*Invited Paper in Proceedings of the Design, Automation, and Test in Europe Conference (**DATE**), Virtual, February 2021.*  
[Slides (pptx) (pdf)]  
[IEDM Tutorial Slides (pptx) (pdf)]  
[Short DATE Talk Video (11 minutes)]  
[Longer IEDM Tutorial Video (1 hr 51 minutes)]

## Intelligent Architectures for Intelligent Computing Systems

Onur Mutlu  
ETH Zurich  
omutlu@gmail.com

# Data-Centric (Memory-Centric) Architectures

# Data-Centric Architectures: Properties

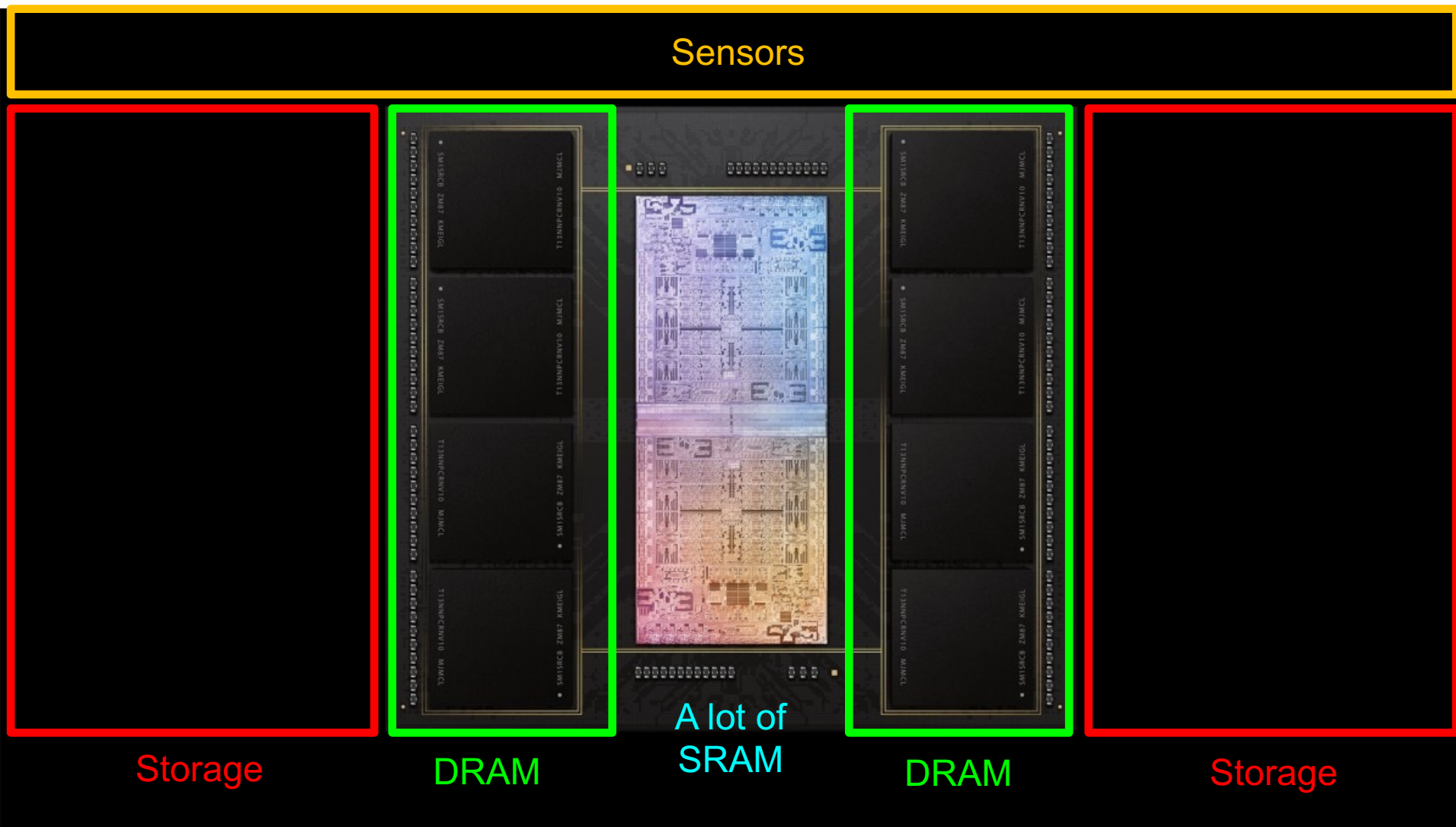
---

- **Process data where it resides** (where it makes sense)
  - Processing in and near memory & sensor structures
- **Low-latency & low-energy data access**
- **Low-cost data storage & processing**
  - High capacity memory at low cost: hybrid memory, compression
- **Intelligent data management**
  - Intelligent controllers handling robustness, security, cost, perf.

# Processing Data Where It Makes Sense



# Process Data Where It Makes Sense



Apple M1 Ultra System (2022)

# Processing in/near Memory: An Old Idea

- Kautz, "Cellular Logic-in-Memory Arrays", IEEE TC 1969.

IEEE TRANSACTIONS ON COMPUTERS, VOL. C-18, NO. 8, AUGUST 1969

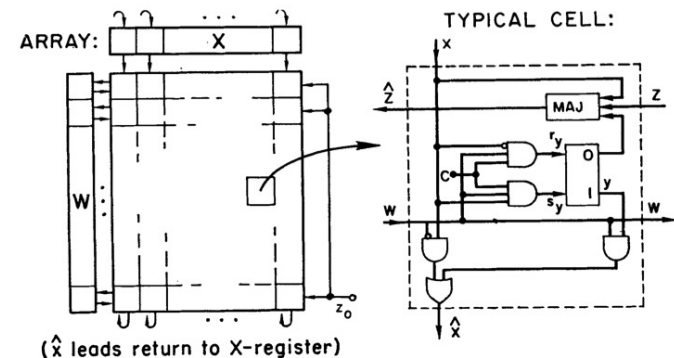
## Cellular Logic-in-Memory Arrays

WILLIAM H. KAUTZ, MEMBER, IEEE

**Abstract**—As a direct consequence of large-scale integration, many advantages in the design, fabrication, testing, and use of digital circuitry can be achieved if the circuits can be arranged in a two-dimensional iterative, or cellular, array of identical elementary networks, or cells. When a small amount of storage is included in each cell, the same array may be regarded either as a logically enhanced memory array, or as a logic array whose elementary gates and connections can be "programmed" to realize a desired logical behavior.

In this paper the specific engineering features of such cellular logic-in-memory (CLIM) arrays are discussed, and one such special-purpose array, a cellular sorting array, is described in detail to illustrate how these features may be achieved in a particular design. It is shown how the cellular sorting array can be employed as a single-address, multiword memory that keeps in order all words stored within it. It can also be used as a content-addressed memory, a pushdown memory, a buffer memory, and (with a lower logical efficiency) a programmable array for the realization of arbitrary switching functions. A second version of a sorting array, operating on a different sorting principle, is also described.

**Index Terms**—Cellular logic, large-scale integration, logic arrays logic in memory, push-down memory, sorting, switching functions.



CELL EQUATIONS:  $\hat{x} = \bar{w}x + wy$   
 $s_y = wcx, r_y = w\bar{c}$   
 $\hat{z} = M(x, \bar{y}, z) = x\bar{y} + z(x + \bar{y})$

Fig. 1. Cellular sorting array I.

# Processing in/near Memory: An Old Idea

---

- Stone, “A Logic-in-Memory Computer,” IEEE TC 1970.

## A Logic-in-Memory Computer

HAROLD S. STONE

*Abstract*—If, as presently projected, the cost of microelectronic arrays in the future will tend to reflect the number of pins on the array rather than the number of gates, the logic-in-memory array is an extremely attractive computer component. Such an array is essentially a microelectronic memory with some combinational logic associated with each storage element.

# Why In-Memory Computation Today?

---

## ■ **Huge problems with Memory Technology**

- ❑ Memory technology scaling is not going well (e.g., RowHammer)
- ❑ Many scaling issues demand intelligence in memory

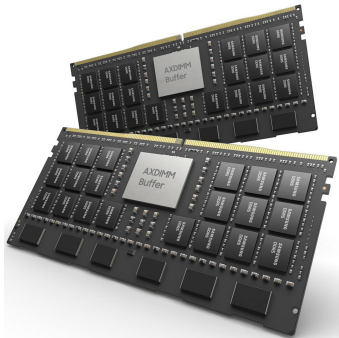
## ■ **Huge demand from Applications & Systems**

- ❑ Data access bottleneck
- ❑ Energy & power bottlenecks
- ❑ Data movement energy dominates computation energy
- ❑ Need all at the same time: performance, energy, sustainability
- ❑ We can improve all metrics by minimizing data movement

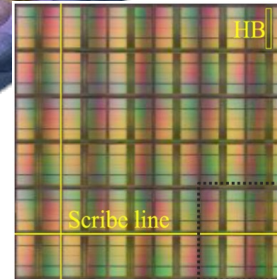
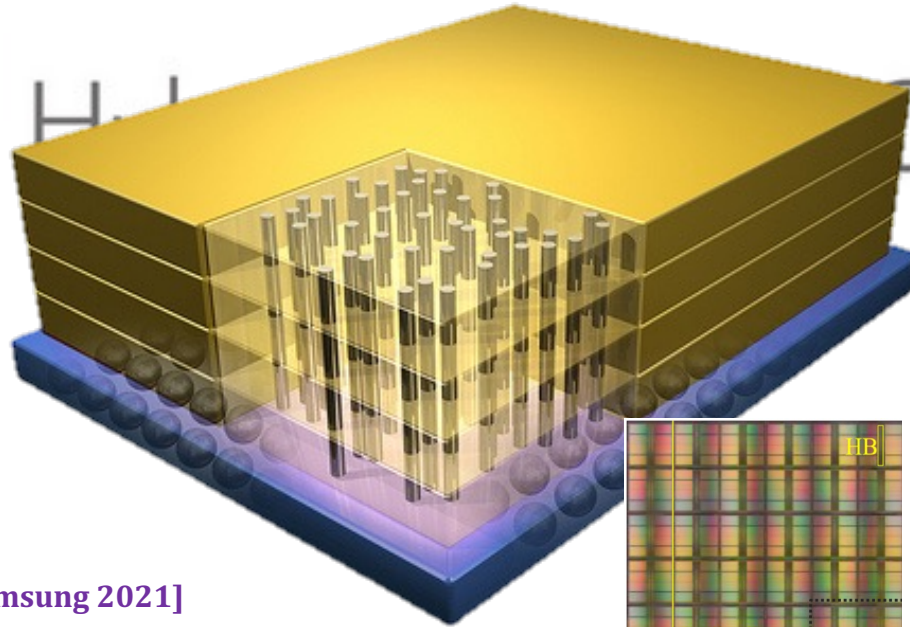
## ■ **Designs are squeezed in the middle**



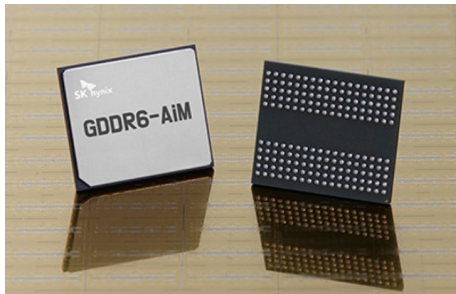
# Processing-in-Memory Landscape Today



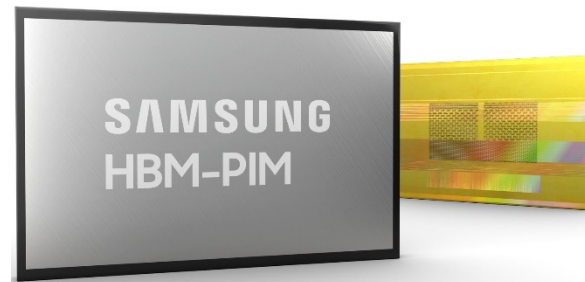
[Samsung 2021]



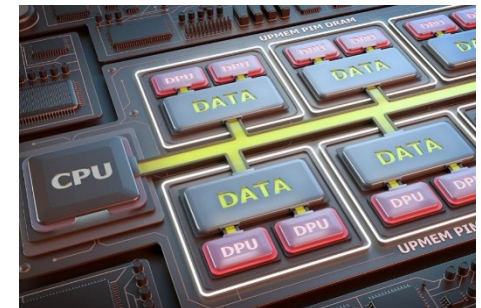
[Alibaba 2022]



[SK Hynix 2022]



[Samsung 2021]



[UPMEM 2019]

# Processing-in-Memory Landscape Today

IEEE COMPUTER ARCHITECTURE LETTERS, VOL. 22, NO. 1, JANUARY-JUNE

## Computational CXL-Memory Solution for Accelerating Memory-Intensive Applications

Joonseop Sim<sup>ID</sup>, Soohong Ahn<sup>ID</sup>, Taeyoung Ahn<sup>ID</sup>,  
Seungyong Lee<sup>ID</sup>, Myunghyun Rhee, Jooyoung Kim<sup>ID</sup>,  
Kwangsik Shin, Donguk Moon<sup>ID</sup>,  
Euseok Kim, and Kyoung Park<sup>ID</sup>

**Abstract**—CXL interface is the up-to-date technology that enables effective memory expansion by providing a memory-sharing protocol in configuring heterogeneous devices. However, its limited physical bandwidth can be a significant bottleneck for emerging data-intensive applications. In this work, we propose a novel CXL-based memory disaggregation architecture with a real-world prototype demonstration, which overcomes the bandwidth limitation of the CXL interface using near-data processing. The experimental results demonstrate that our design achieves up to  $1.9\times$  better performance/power efficiency than the existing CPU system.

**Index Terms**—Compute express link (CXL), near-data-processing (NDP)

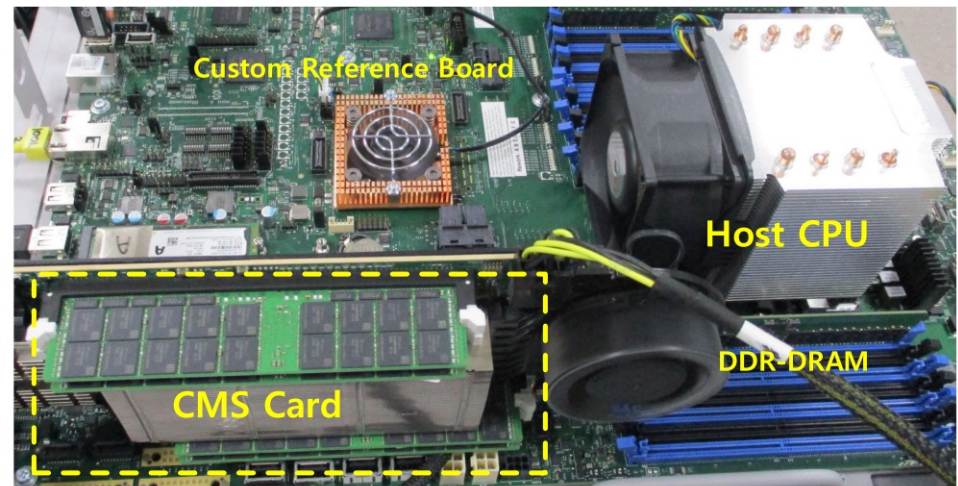


Fig. 6. FPGA prototype of proposed CMS card.

# Processing-in-Memory Landscape Today

## Samsung Processing in Memory Technology at Hot Chips 2023

By **Patrick Kennedy** - August 28, 2023



*Samsung PIM PNM For Transformer Based AI HC35\_Page\_24*



# Memory Scaling Issues **Are** Real

---

- Onur Mutlu,  
**"Memory Scaling: A Systems Architecture Perspective"**  
*Proceedings of the 5th International Memory Workshop (IMW)*, Monterey, CA, May 2013. Slides  
(pptx) (pdf)  
EETimes Reprint

## Memory Scaling: A Systems Architecture Perspective

Onur Mutlu  
Carnegie Mellon University  
onur@cmu.edu  
<http://users.ece.cmu.edu/~omutlu/>

# A Curious Phenomenon [Kim et al., ISCA 2014]

---

One can  
predictably induce errors  
in most DRAM memory chips

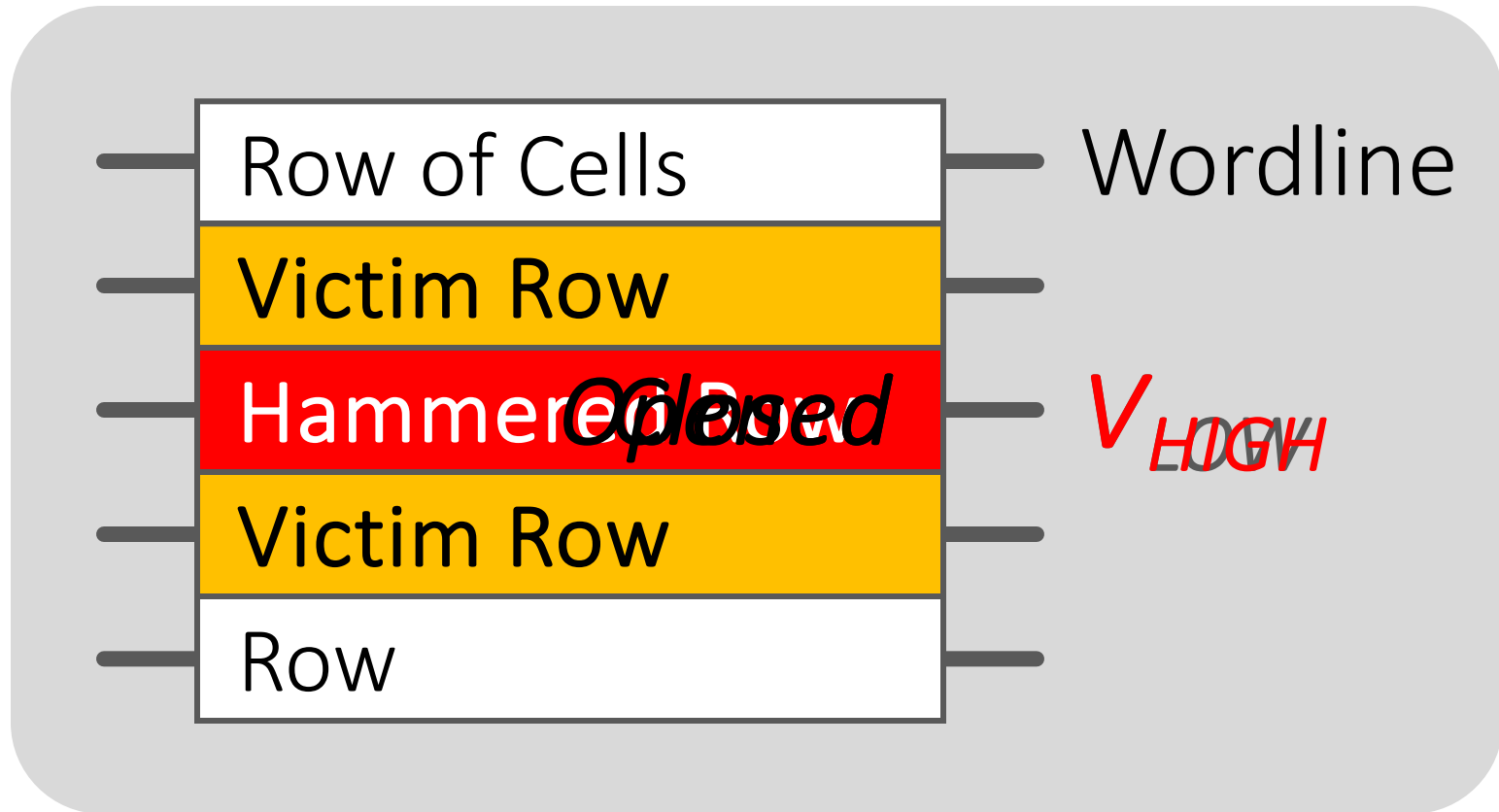
Kim+, "[Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors](#)," ISCA 2014.



Rowhammer

---

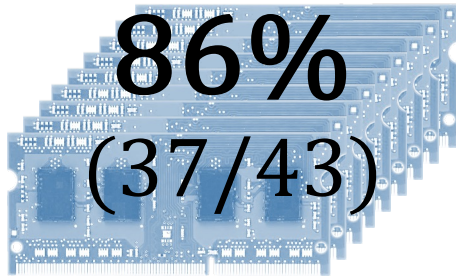
# Modern Memory is Prone to Disturbance Errors



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

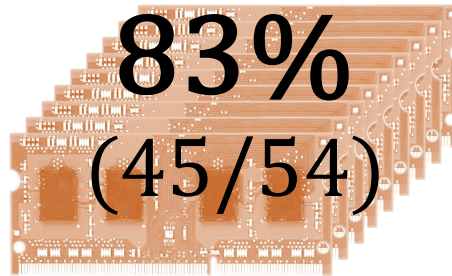
# Most DRAM Modules Are Vulnerable

A company



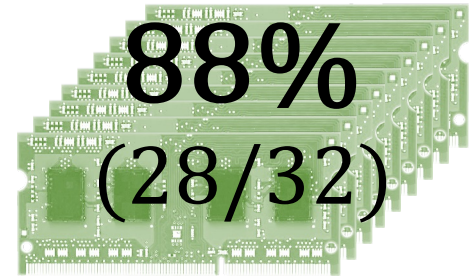
Up to  
 $1.0 \times 10^7$   
errors

B company



Up to  
 $2.7 \times 10^6$   
errors

C company



Up to  
 $3.3 \times 10^5$   
errors

# The RowHammer Vulnerability

---

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



# RowHammer [ISCA 2014]

- 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 \(pptx\) \(pdf\)\]](#) [\[Source Code and Data\]](#) [\[Lecture Video\]](#) (1 hr 49 mins), 25 September 2020]  
***One of the 7 papers of 2012-2017 selected as Top Picks in Hardware and Embedded Security for IEEE TCAD ([link](#)).***  
***Selected to the ISCA-50 25-Year Retrospective Issue covering 1996-2020 in 2023 ([Retrospective \(pdf\)](#) [Full Issue](#)).***

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

Yoongu Kim<sup>1</sup>   Ross Daly\*   Jeremie Kim<sup>1</sup>   Chris Fallin\*   Ji Hye Lee<sup>1</sup>  
Donghyuk Lee<sup>1</sup>   Chris Wilkerson<sup>2</sup>   Konrad Lai   Onur Mutlu<sup>1</sup>

<sup>1</sup>Carnegie Mellon University   <sup>2</sup>Intel Labs

# Memory Scaling Issues **Are** Real

---

- 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<sup>§‡</sup>      Jeremie S. Kim<sup>‡§</sup>  
<sup>§</sup>ETH Zürich      <sup>‡</sup>Carnegie Mellon University

# Memory Scaling Issues **Are** Real

---

- Onur Mutlu, Ataberk Olgun, and A. Giray Yaglikci,  
**"Fundamentally Understanding and Solving RowHammer"**  
*Invited Special Session Paper at the 28th Asia and South Pacific Design Automation Conference (ASP-DAC), Tokyo, Japan, January 2023.*  
[arXiv version]  
[Slides (pptx) (pdf)]  
[Talk Video (26 minutes)]

## Fundamentally Understanding and Solving RowHammer

Onur Mutlu  
onur.mutlu@safari.ethz.ch  
ETH Zürich  
Zürich, Switzerland

Ataberk Olgun  
ataberk.olgund@safari.ethz.ch  
ETH Zürich  
Zürich, Switzerland

A. Giray Yağlıkçı  
giray.yaglikci@safari.ethz.ch  
ETH Zürich  
Zürich, Switzerland

## Main Memory Needs Intelligent Controllers

# Industry's Intelligent DRAM Controllers (I)

## ISSCC 2023 / SESSION 28 / HIGH-DENSITY MEMORIES /

### **28.8 A 1.1V 16Gb DDR5 DRAM with Probabilistic-Aggressor Tracking, Refresh-Management Functionality, Per-Row Hammer Tracking, a Multi-Step Precharge, and Core-Bias Modulation for Security and Reliability Enhancement**

Woongrae Kim, Chulmoon Jung, Seongnyuh Yoo, Duckhwa Hong, Jeongjin Hwang, Jungmin Yoon, Oh Yong Jung, Joonwoo Choi, Sanga Hyun, Mankeun Kang, Sangho Lee, Dohong Kim, Sanghyun Ku, Donhyun Choi, Nogeun Joo, Sangwoo Yoon, Junseok Noh, Byeongyong Go, Cheolhoe Kim, Sunil Hwang, Mihyun Hwang, Seol-Min Yi, Hyungmin Kim, Sanghyuk Heo, Yeonsu Jang, Kyoungchul Jang, Shinho Chu, Yoonna Oh, Kwidong Kim, Junghyun Kim, Soohwan Kim, Jeongtae Hwang, Sangil Park, Junphyo Lee, Inchul Jeong, Joohwan Cho, Jonghwan Kim

SK hynix Semiconductor, Icheon, Korea





# Industry's Intelligent DRAM Controllers (II)

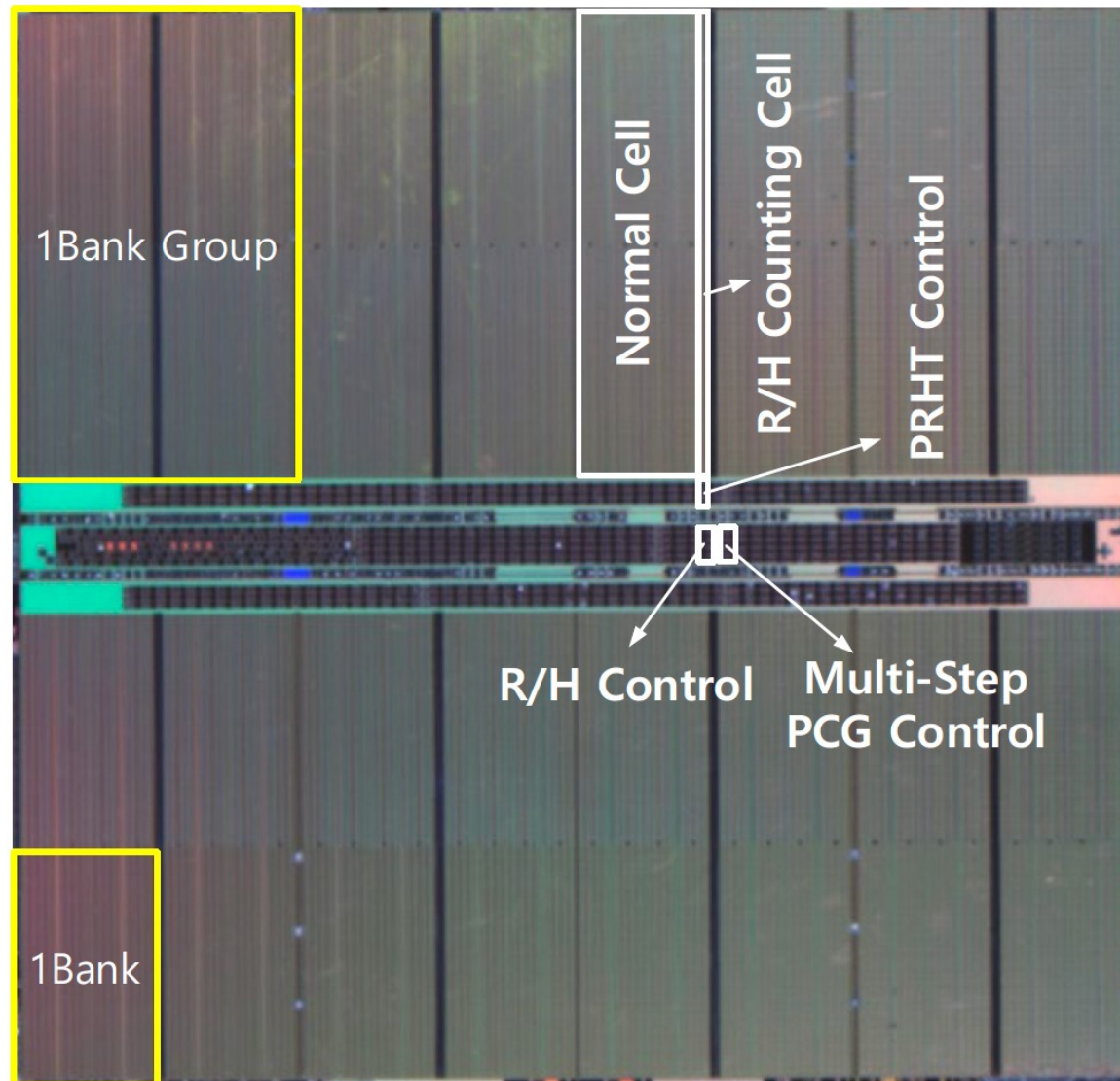
---

SK hynix Semiconductor, Icheon, Korea

DRAM products have been recently adopted in a wide range of high-performance computing applications: such as in cloud computing, in big data systems, and IoT devices. This demand creates larger memory capacity requirements, thereby requiring aggressive DRAM technology node scaling to reduce the cost per bit [1,2]. However, DRAM manufacturers are facing technology scaling challenges due to row hammer and refresh retention time beyond 1a-nm [2]. Row hammer is a failure mechanism, where repeatedly activating a DRAM row disturbs data in adjacent rows. Scaling down severely threatens reliability since a reduction of DRAM cell size leads to a reduction in the intrinsic row hammer tolerance [2,3]. To improve row hammer tolerance, there is a need to probabilistically activate adjacent rows with carefully sampled active addresses and to improve intrinsic row hammer tolerance [2]. In this paper, row-hammer-protection and refresh-management schemes are presented to guarantee DRAM security and reliability despite the aggressive scaling from 1a-nm to sub 10-nm nodes. The probabilistic-aggressor-tracking scheme with a refresh-management function (RFM) and per-row hammer tracking (PRHT) improve DRAM resilience. A multi-step precharge reinforces intrinsic row-hammer tolerance and a core-bias modulation improves retention time: even in the face of cell-transistor degradation due to technology scaling. This comprehensive scheme leads to a reduced probability of failure, due to row hammer attacks, by 93.1% and an improvement in retention time by 17%.



# Industry's Intelligent DRAM Controllers (III)



ISSCC 2023 / SESSION 28 / HIGH-DENSITY MEMORIES

**28.8 A 1.1V 16Gb DDR5 DRAM with Probabilistic-Aggressor Tracking, Refresh-Management Functionality, Per-Row Hammer Tracking, a Multi-Step Precharge, and Core-Bias Modulation for Security and Reliability Enhancement**

Woongrae Kim, Chulmoon Jung, Seongnyuh Yoo, Duckhwa Hong, Jeongjin Hwang, Jungmin Yoon, Ohyoung Jung, Joonwoo Choi, Sanga Hyun, Mankeun Kang, Sangho Lee, Dohong Kim, Sanghyun Ku, Donhyun Choi, Nogeun Joo, Sangwoo Yoon, Junseok Noh, Byeongyong Go, Cheolhoe Kim, Sunil Hwang, Mihyun Hwang, Seol-Min Yi, Hyungmin Kim, Sanghyuk Heo, Yeonsu Jang, Kyoungchul Jang, Shinho Chu, Yoonna Oh, Kwidong Kim, Junghyun Kim, Soohwan Kim, Jeongtae Hwang, Sangil Park, Junphyo Lee, Inchul Jeong, Joohwan Cho, Jonghwan Kim

SK hynix Semiconductor, Icheon, Korea

## DSAC: Low-Cost Rowhammer Mitigation Using In-DRAM Stochastic and Approximate Counting Algorithm

Seungki Hong   Dongha Kim   Jaehyung Lee   Reum Oh  
Changsik Yoo   Sangjoon Hwang   Jooyoung Lee

DRAM Design Team, Memory Division, Samsung Electronics

<https://arxiv.org/pdf/2302.03591v1.pdf>

# Are We Now BitFlip Free?

---

- **Appears 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*



- 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*

# A Recent RowHammer Lecture

**What Is 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

**FORGET SOFTWARE—NOW HACKERS ARE EXPLOITING PHYSICS**

Stanford

Stanford Online 529K subscribers

Subscribed

35

Share

Download

Save

1.2K views 1 month ago



# Emerging Memories Also Need Intelligent Controllers

---

- 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<sup>†</sup> Engin Ipek<sup>†</sup> Onur Mutlu<sup>‡</sup> Doug Burger<sup>†</sup>

<sup>†</sup>Computer Architecture Group  
Microsoft Research  
Redmond, WA  
{blee, ipek, dburger}@microsoft.com

<sup>‡</sup>Computer Architecture Laboratory  
Carnegie Mellon University  
Pittsburgh, PA  
onur@cmu.edu



Intelligent  
Memory Controllers  
Can Avoid Many Failures  
& Enable Better Scaling

# Three Key Systems & Application Trends

---

## 1. **Data access** is the major bottleneck

- ▣ Applications are increasingly data hungry

## 2. **Energy** consumption is a key limiter

## 3. **Data movement energy** dominates compute

- ▣ Especially true for off-chip to on-chip movement

# Do We Want This?

---





# Or This?

---





High Performance,

Energy Efficient,

Sustainable

(All at the Same Time)

# The Problem

---

Data access is the major performance and energy bottleneck

Our current  
design principles  
cause great energy waste  
(and great performance loss)



# The Problem

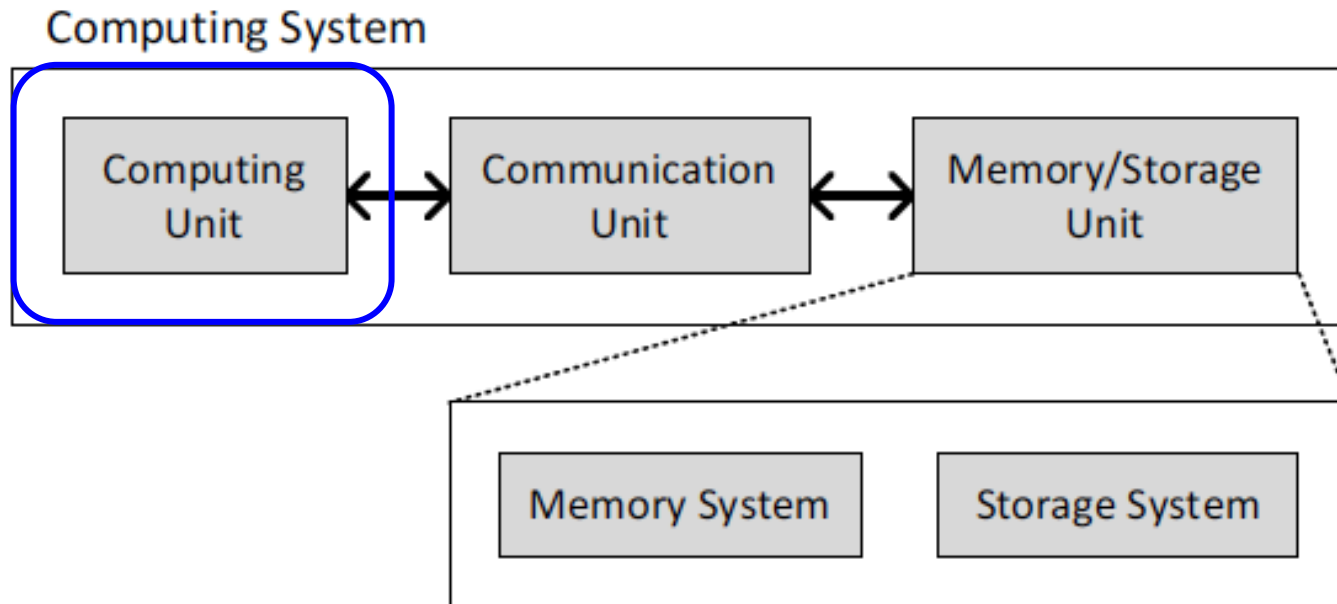
---

Processing of data  
is performed  
far away from the data

# Today's Computing Systems

---

- Processor centric
- All data processed in the processor → at great system cost



# It's the Memory, Stupid!

---

- **“It's the Memory, Stupid!”** (Richard Sites, MPR, 1996)

**RICHARD SITES**

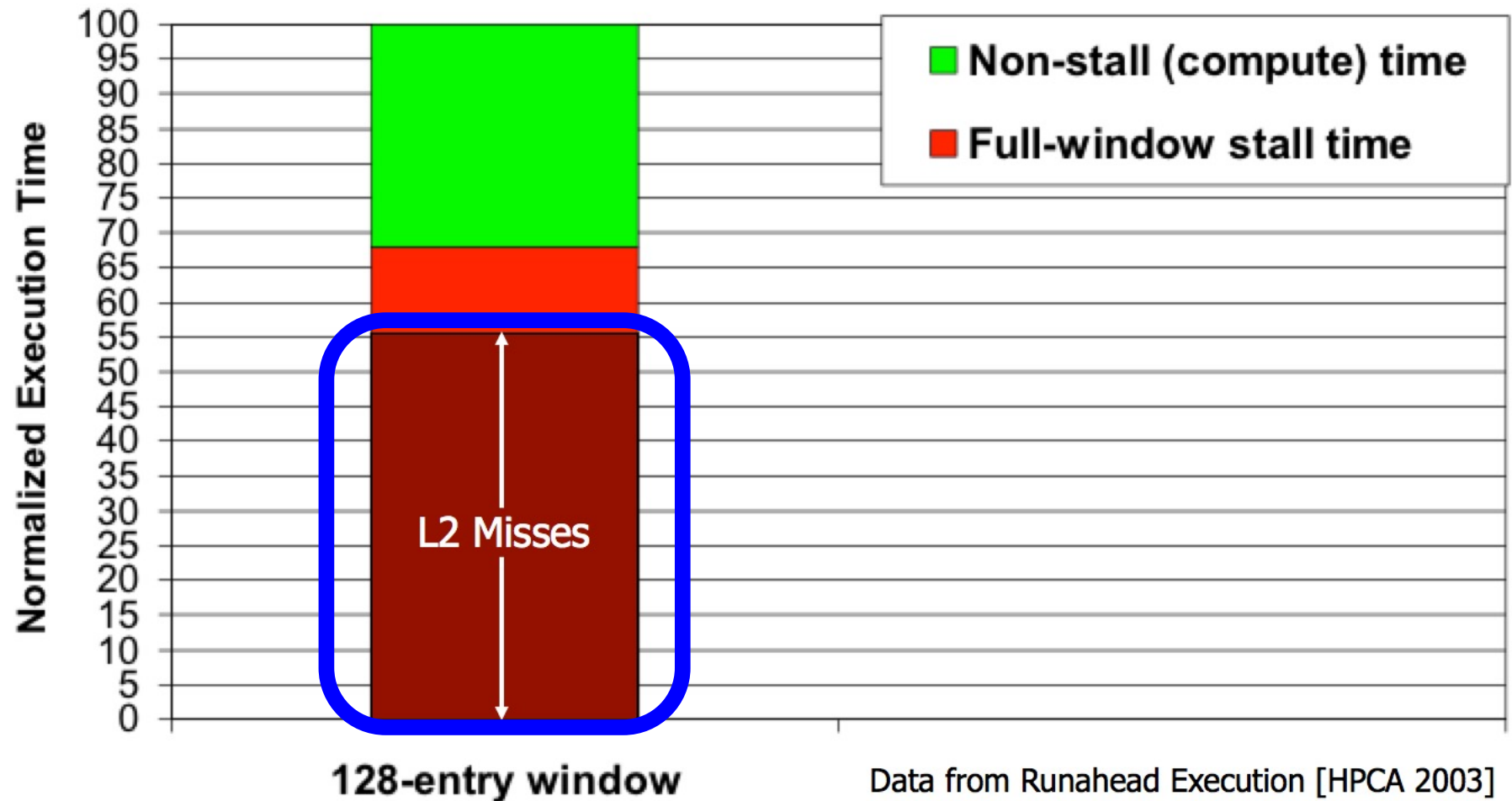
## **It's the Memory, Stupid!**

When we started the Alpha architecture design in 1988, we estimated a 25-year lifetime and a relatively modest 32% per year compounded performance improvement of implementations over that lifetime (1,000× total). We guestimated about 10× would come from CPU clock improvement, 10× from multiple instruction issue, and 10× from multiple processors.

5, 1996  MICROPROCESSOR REPORT

I expect that over the coming decade memory subsystem design will be the *only* important design issue for microprocessors.

# The Performance Perspective



# The Performance Perspective

---

- Onur Mutlu, Jared Stark, Chris Wilkerson, and Yale N. Patt,  
**"Runahead Execution: An Alternative to Very Large Instruction Windows for Out-of-order Processors"**  
*Proceedings of the 9th International Symposium on High-Performance Computer Architecture (HPCA)*, pages 129-140, Anaheim, CA, February 2003. [Slides \(pdf\)](#)  
***One of the 15 computer arch. papers of 2003 selected as Top Picks by IEEE Micro. HPCA Test of Time Award (awarded in 2021).***

## Runahead Execution: An Alternative to Very Large Instruction Windows for Out-of-order Processors

Onur Mutlu §    Jared Stark †    Chris Wilkerson ‡    Yale N. Patt §

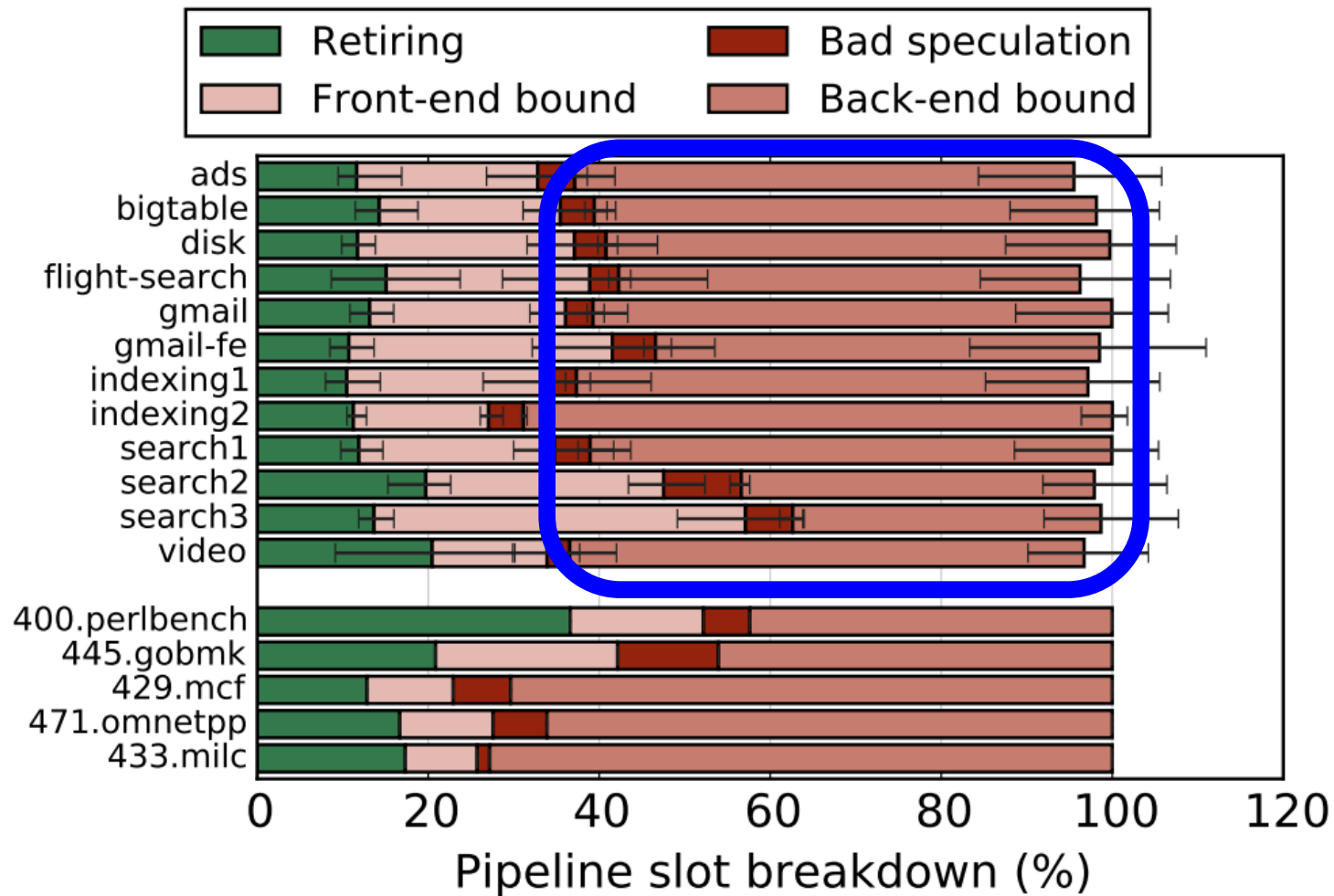
§ECE Department  
The University of Texas at Austin  
{onur,patt}@ece.utexas.edu

†Microprocessor Research  
Intel Labs  
jared.w.stark@intel.com

‡Desktop Platforms Group  
Intel Corporation  
chris.wilkerson@intel.com

# The Performance Perspective (Today)

- All of Google's Data Center Workloads (2015):

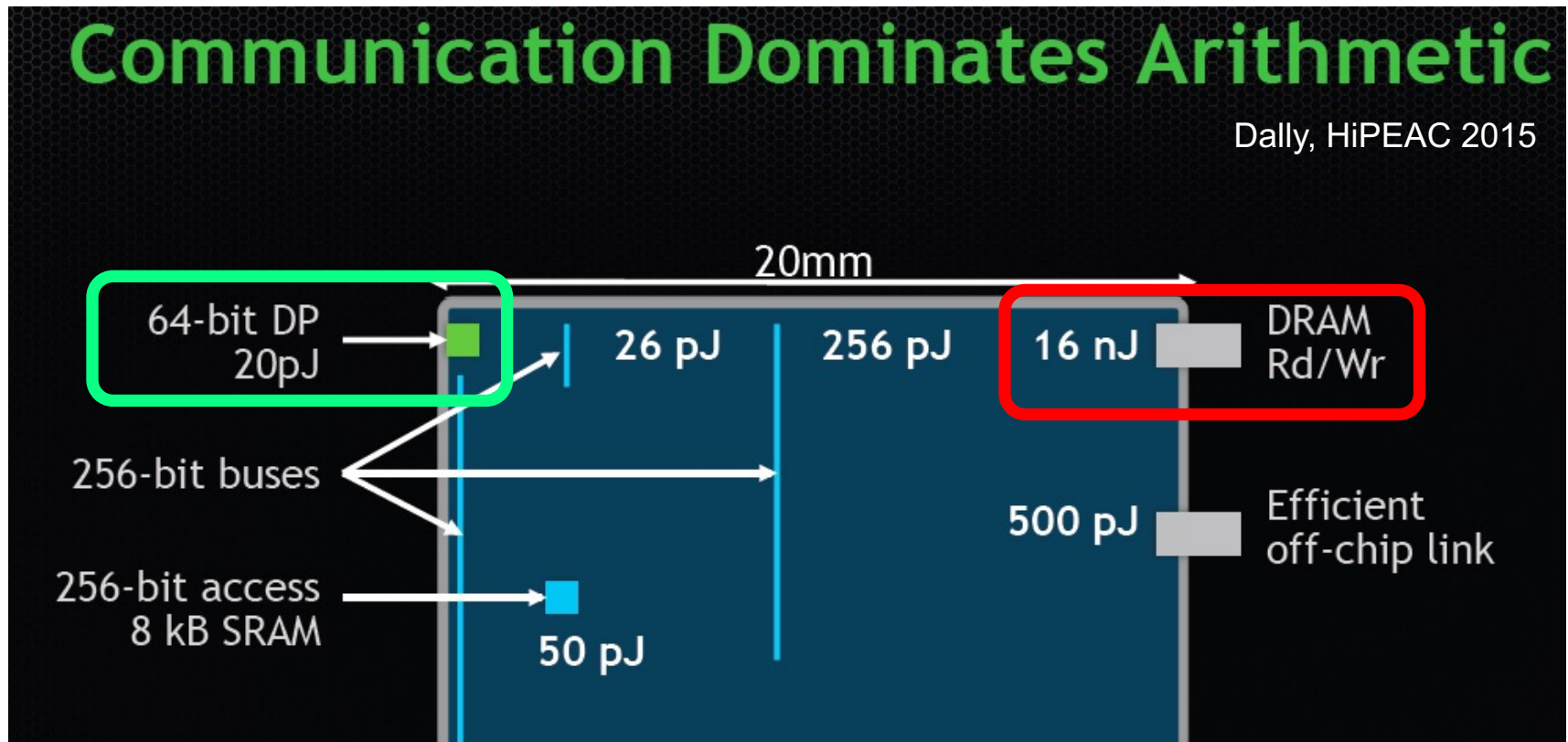




# Data Movement vs. Computation Energy

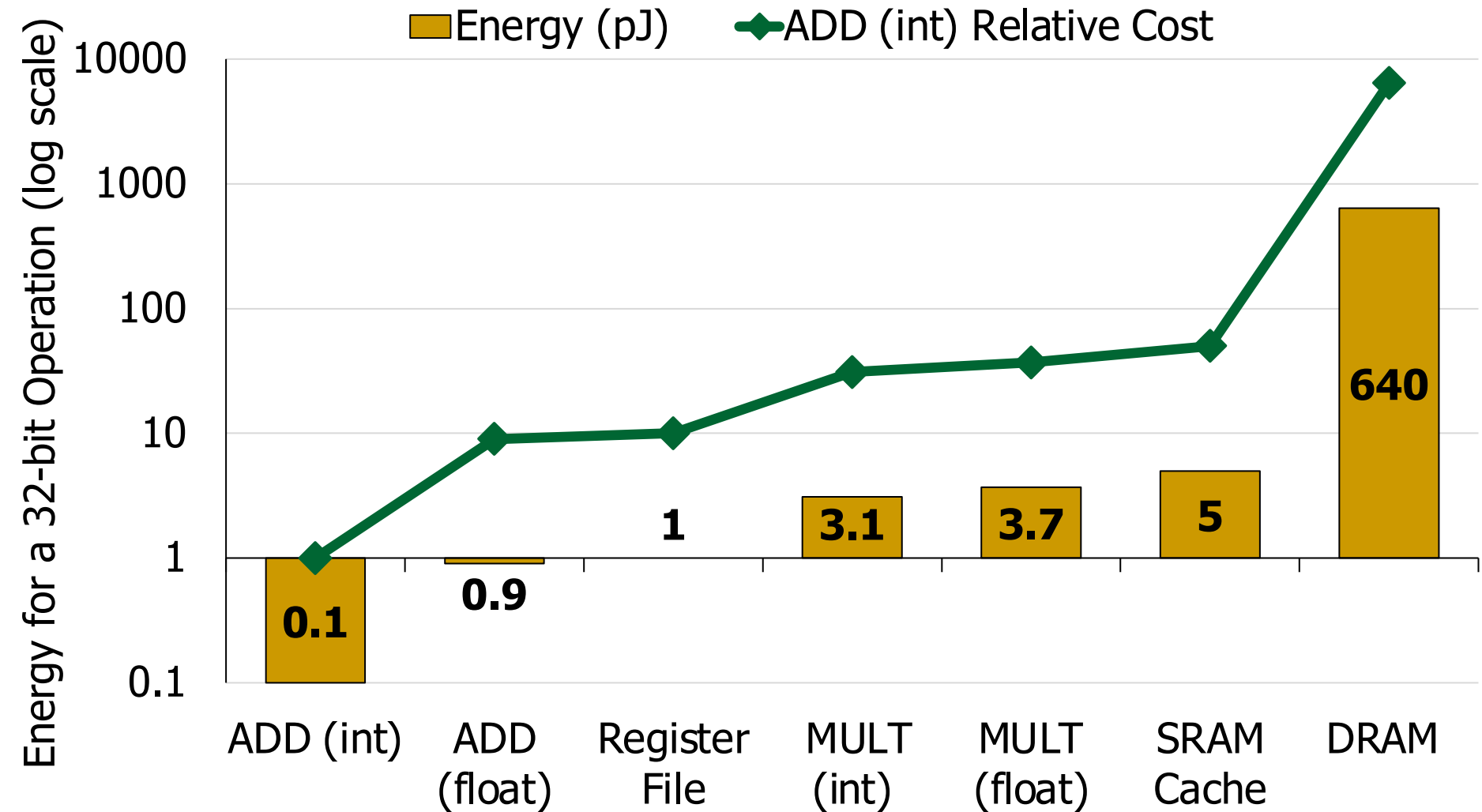
## Communication Dominates Arithmetic

Dally, HiPEAC 2015

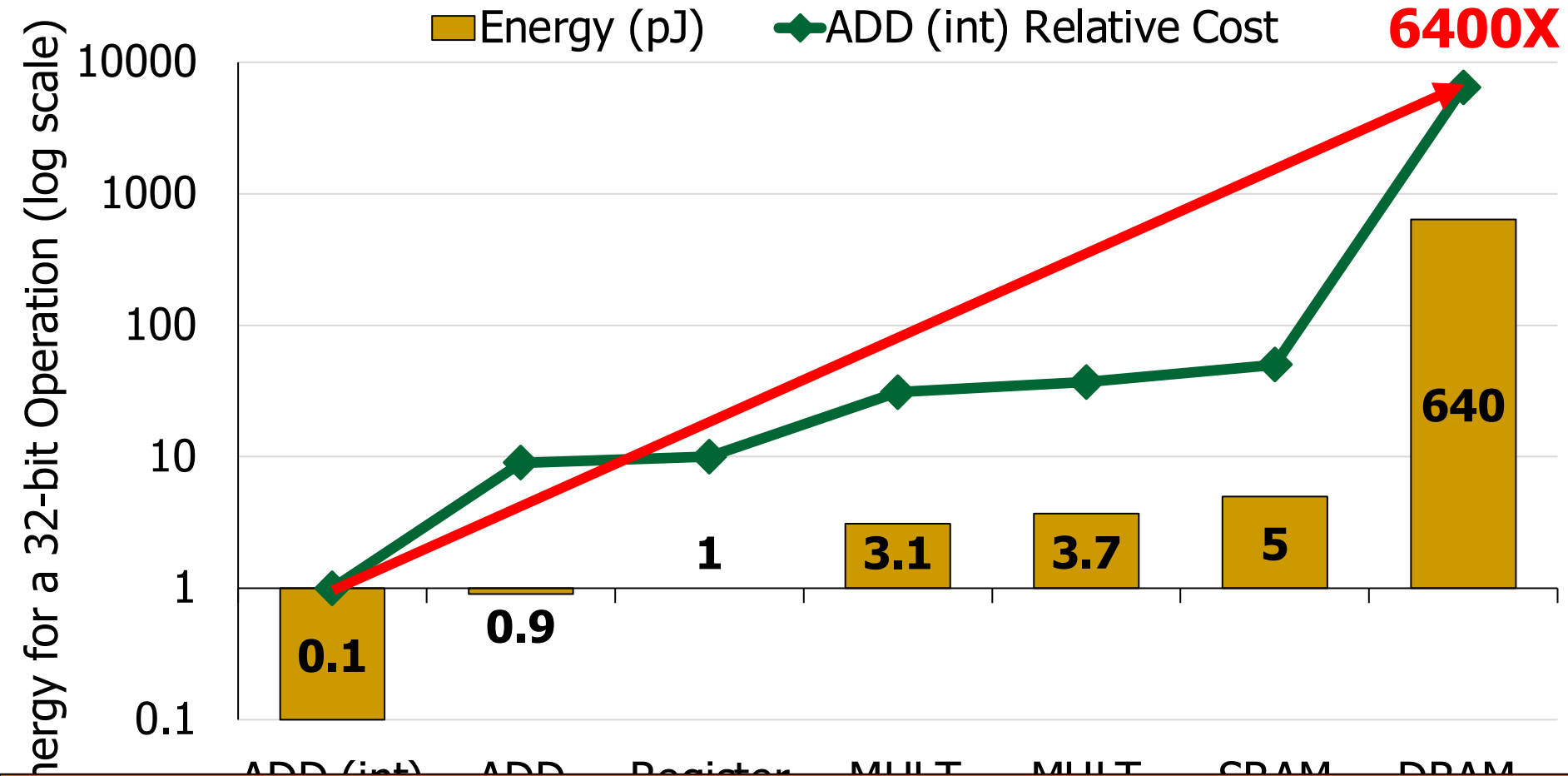


A memory access consumes  $\sim 100\text{-}1000\times$  the energy of a complex addition

# Data Movement vs. Computation Energy



# Data Movement vs. Computation Energy



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

# Energy Waste in 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.

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

## Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks

Amirali Boroumand<sup>1</sup>

Saugata Ghose<sup>1</sup>

Youngsok Kim<sup>2</sup>

Rachata Ausavarungnirun<sup>1</sup>

Eric Shiu<sup>3</sup>

Rahul Thakur<sup>3</sup>

Daehyun Kim<sup>4,3</sup>

Aki Kuusela<sup>3</sup>

Allan Knies<sup>3</sup>

Parthasarathy Ranganathan<sup>3</sup>

Onur Mutlu<sup>5,1</sup>

# Energy Waste in Accelerators

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

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

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

Amirali Boroumand<sup>†◇</sup>  
Geraldo F. Oliveira<sup>\*</sup>

Saugata Ghose<sup>‡</sup>  
Xiaoyu Ma<sup>§</sup>

Berkin Akin<sup>§</sup>  
Eric Shiu<sup>§</sup>

Ravi Narayanaswami<sup>§</sup>  
Onur Mutlu<sup>\*†</sup>

<sup>†</sup>Carnegie Mellon Univ.

<sup>◇</sup>Stanford Univ.

<sup>‡</sup>Univ. of Illinois Urbana-Champaign

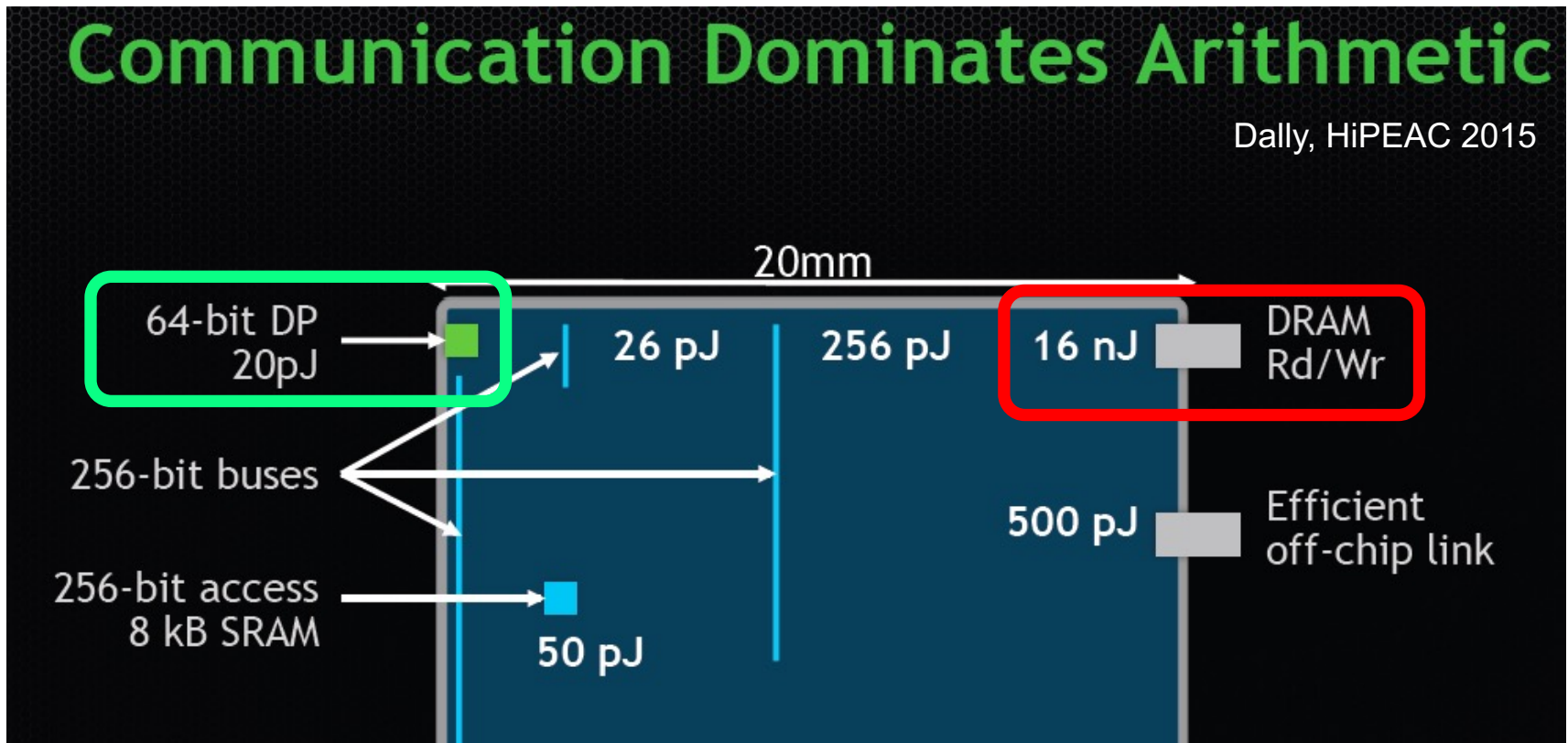
<sup>§</sup>Google

<sup>\*</sup>ETH Zürich

# We Do Not Want to Move Data!

## Communication Dominates Arithmetic

Dally, HiPEAC 2015



A memory access consumes  $\sim 100\text{-}1000\times$  the energy of a complex addition

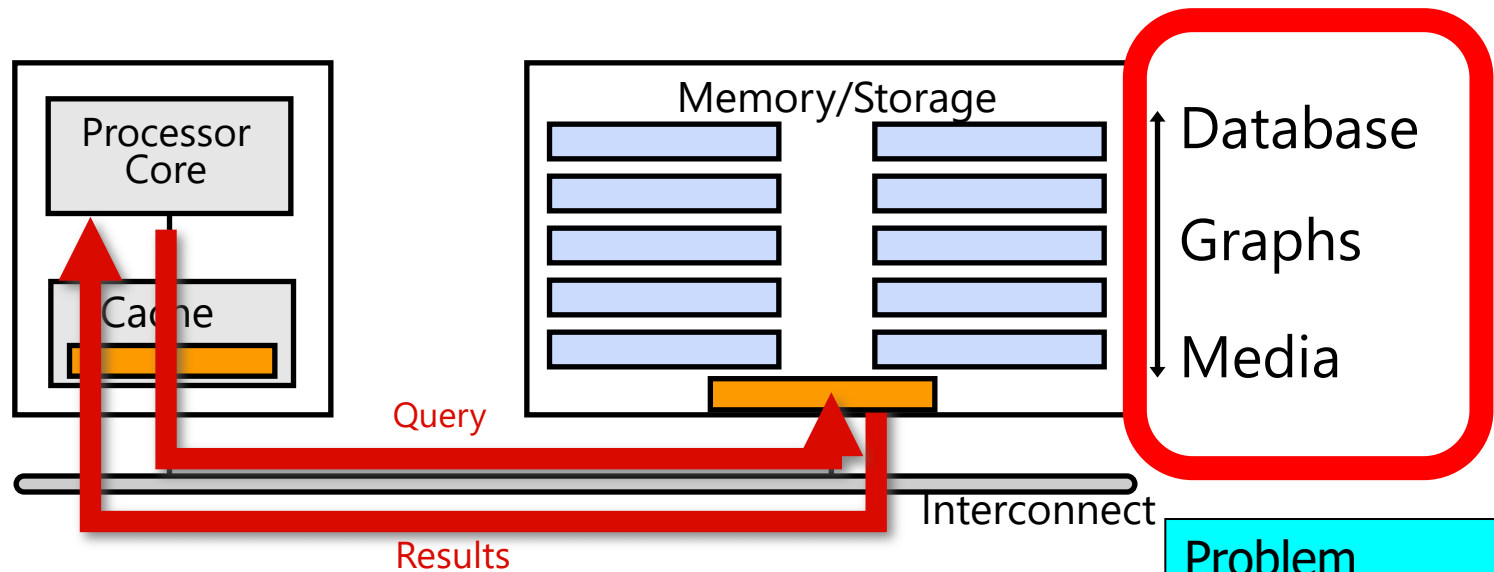


# We Need A Paradigm Shift To ...

---

- Enable computation with minimal data movement
- Compute where it makes sense (where data resides)
- Make computing architectures more data-centric

# Goal: Processing Inside Memory/Storage



- Many questions ... How do we design the:
  - ❑ compute-capable memory & controllers?
  - ❑ processors & communication units?
  - ❑ software & hardware interfaces?
  - ❑ system software, compilers, languages?
  - ❑ algorithms & theoretical foundations?

|                    |
|--------------------|
| Problem            |
| Algorithm          |
| Program/Language   |
| System Software    |
| SW/HW Interface    |
| Micro-architecture |
| Logic              |
| Devices            |
| Electrons          |

# PIM Review and Open Problems

---

## A Modern Primer on Processing in Memory

Onur Mutlu<sup>a,b</sup>, Saugata Ghose<sup>b,c</sup>, Juan Gómez-Luna<sup>a</sup>, Rachata Ausavarungnirun<sup>d</sup>

*SAFARI Research Group*

<sup>a</sup>*ETH Zürich*

<sup>b</sup>*Carnegie Mellon University*

<sup>c</sup>*University of Illinois at Urbana-Champaign*

<sup>d</sup>*King Mongkut's University of Technology North Bangkok*

Onur Mutlu, Saugata Ghose, Juan Gomez-Luna, and Rachata Ausavarungnirun,  
**"A Modern Primer on Processing in Memory"**  
*Invited Book Chapter in **Emerging Computing: From Devices to Systems - Looking Beyond Moore and Von Neumann**, Springer, to be published in 2021.*

# PIM Course (Fall 2022)

## ■ Fall 2022 Edition:

- [https://safari.ethz.ch/projects\\_and\\_seminars/fall2022/doku.php?id=processing\\_in\\_memory](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](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):

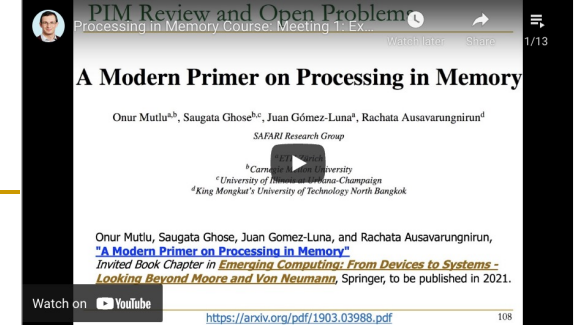
- <https://www.youtube.com/watch?v=9e4Chnwdovo&list=PL5Q2soXY2Zi-841fUYUK9EsXKhQKRPyX>

## ■ 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>

**SAFARI**

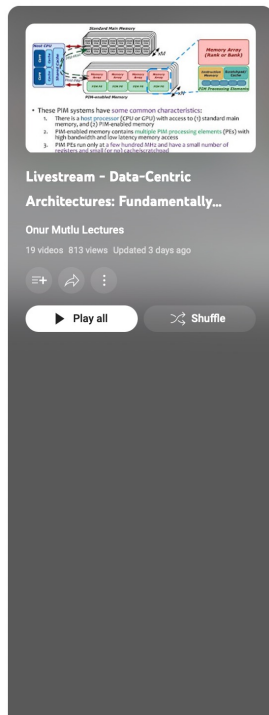


### Spring 2022 Meetings/Schedule

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

# Processing-in-Memory Course (Spring 2023)

- Short weekly lectures
- Hands-on projects



- PIM Course: Lecture 1: Data-Centric Architectures: Improving Performance & Energy (Spring 2023)**  
Onur Mutlu Lectures • 1.1K views • Streamed 3 months ago  
1:14:16
- PIM Course: Lecture 2: How to Evaluate Data Movement Bottlenecks (Spring 2023)**  
Onur Mutlu Lectures • 332 views • 2 months ago  
16:37
- ASPLOS 2023 Tutorial: Real-world Processing-in-Memory Systems for Modern Workloads**  
Onur Mutlu Lectures • 1.5K views • Streamed 2 months ago  
6:27:39
- PIM Course: Lecture 3: Real-world PIM: UPMEM PIM (Spring 2023)**  
Onur Mutlu Lectures • 411 views • 2 months ago  
19:43
- PIM Course: Lecture 4: Real-world PIM: Microbenchmarking of UPMEM PIM (Spring 2023)**  
Onur Mutlu Lectures • 188 views • 2 months ago  
24:10
- Análisis Experimental de una Arquitectura PIM - Juan Gómez Luna - Lecture in Spanish @ U. de Córdoba**  
Onur Mutlu Lectures • 169 views • 2 months ago  
2:27:12
- PIM Course: Lecture 5: Real-world PIM: Samsung HBM-PIM (Spring 2023)**  
Onur Mutlu Lectures • 483 views • 2 months ago  
24:08
- PIM Course: Lecture 6: Real-world PIM: SK Hynix AIM (Spring 2023)**  
Onur Mutlu Lectures • 573 views • 1 month ago  
35:50
- PIM Course: Lecture 7: Real-world PIM: Samsung AxDIMM (Spring 2023)**  
Onur Mutlu Lectures • 325 views • 1 month ago  
21:32

[https://www.youtube.com/playlist?list=PL5Q2soXY2zi\\_EObuoAZVSq\\_o6UySWQHvz](https://www.youtube.com/playlist?list=PL5Q2soXY2zi_EObuoAZVSq_o6UySWQHvz)

**SAFARI Project & Seminars Courses**  
(Spring 2023)

Search

Recent Changes Media Manager Sitemap

Trace: • heterogeneous\_systems • **processing\_in\_memory**

Home

**Courses**

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

**processing\_in\_memory**

**Table of Contents**

- Data-Centric Architectures: Fundamentally Improving Performance and Energy (227-0085-37L)
- Course Description
- Mentors
- Lecture Video Playlist on YouTube
- Spring 2023 Meetings/Schedule
- Past Lecture Video Playlists on YouTube
- Learning Materials
- Assignments

**Data-Centric Architectures: Fundamentally Improving Performance and Energy (227-0085-37L)**

**Course Description**

Data movement between the memory units and the compute units of current computing systems is a major performance and energy bottleneck. From large-scale servers to mobile devices, data movement costs dominate computation costs in terms of both performance and energy consumption. For example, data movement between the main memory and the processing cores accounts for 62% of the total system energy in consumer applications. As a result, the data movement bottleneck is a huge burden that greatly limits the energy efficiency and performance of modern computing systems. This phenomenon is an undesired effect of the dichotomy between memory and the processor, which leads to the data movement bottleneck.

Many modern and important workloads such as machine learning, computational biology, graph processing, databases, video analytics, and real-time data analytics suffer greatly from the data movement bottleneck. These workloads are exemplified by irregular memory accesses, relatively low data reuse, low cache line utilization, low arithmetic intensity (i.e., ratio of operations per accessed byte), and large datasets that greatly exceed the main memory size. The computation in these workloads cannot usually compensate for the data movement costs. In order to alleviate this data movement bottleneck, we need a paradigm shift from the traditional processor-centric design, where all computation takes place in the compute units, to a more data-centric design where processing elements are placed closer to or inside where the data resides. This paradigm of computing is known as Processing-in-Memory (PIM).

This is your perfect P&S if you want to become familiar with the main PIM technologies, which represent “the next big thing” in Computer Architecture. You will work hands-on with the first real-world PIM architecture, will explore different PIM architecture designs for important workloads, and will develop tools to enable research of future PIM systems. Projects in this course span software and hardware as well as the software/hardware interface. You can potentially work on developing and optimizing new workloads for the first real-world PIM hardware or explore new PIM designs in simulators, or do something else that can forward our understanding of the PIM paradigm.

**Prerequisites of the course:**

- Digital Design and Computer Architecture (or equivalent course).
- Familiarity with C/C++ programming.
- Interest in future computer architectures and computing paradigms.
- Interest in discovering why things do or do not work and solving problems
- Interest in making systems efficient and usable

[https://safari.ethz.ch/projects\\_and\\_seminars/spring2023/doku.php?id=processing\\_in\\_memory](https://safari.ethz.ch/projects_and_seminars/spring2023/doku.php?id=processing_in_memory)

# SSD Course (Spring 2023)

## Spring 2023 Edition:

- [https://safari.ethz.ch/projects\\_and\\_seminars/spring2023/doku.php?id=modern\\_ssd](https://safari.ethz.ch/projects_and_seminars/spring2023/doku.php?id=modern_ssd)

## Fall 2022 Edition:

- [https://safari.ethz.ch/projects\\_and\\_seminars/fall2022/doku.php?id=modern\\_ssd](https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=modern_ssd)

## Youtube Livestream (Spring 2023):

- [https://www.youtube.com/watch?v=4VTwOMmsnJY&list=PL5Q2soXY2Zi\\_8qOM5Icpp8hB2SHtm4z57&pp=iAQB](https://www.youtube.com/watch?v=4VTwOMmsnJY&list=PL5Q2soXY2Zi_8qOM5Icpp8hB2SHtm4z57&pp=iAQB)

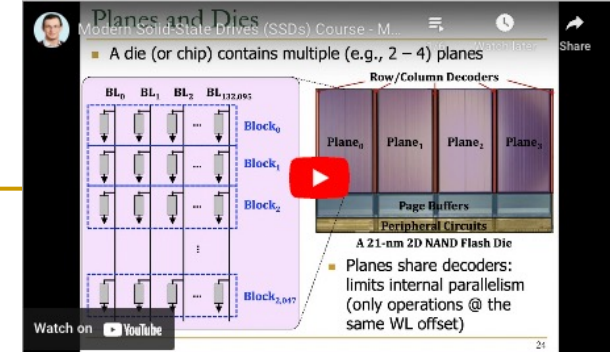
## Youtube Livestream (Fall 2022):

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

## Project course

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

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



## Fall 2022 Meetings/Schedule

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



# Real PIM Tutorials [MICRO'23, ISCA'23, ASPLOS'23, HPCA'23]

- June, March, Feb : Lectures + Hands-on labs + Invited talks



## ISCA 2023 Real-World PIM Tutorial

Search

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

Trace: • [start](#)

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

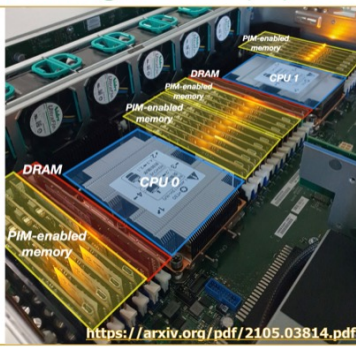
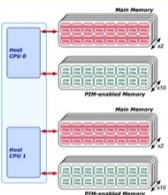
#### Tutorial Description

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.

#### Table of Contents

- [Real-world Processing-in-Memory Systems for Modern Workloads](#)
- [Tutorial Description](#)
- [Organizers](#)
- [Agenda \(June 18, 2023\)](#)
- [Lectures \(tentative\)](#)
- [Hands-on Labs \(tentative\)](#)
- [Learning Materials](#)

<https://events.safari.ethz.ch/isca-pim-tutorial/>

We Need to Think Differently  
from the Past Approaches

# Processing in Memory: Two Approaches

1. Processing **using** Memory
2. Processing **near** Memory

# A PIM Taxonomy

---

- **Nature** (of computation)

- **Using**: Use operational properties of memory structures
- **Near**: Add logic close to memory structures

- **Technology**

- Flash, DRAM, SRAM, RRAM, MRAM, FeRAM, PCM, 3D, ...

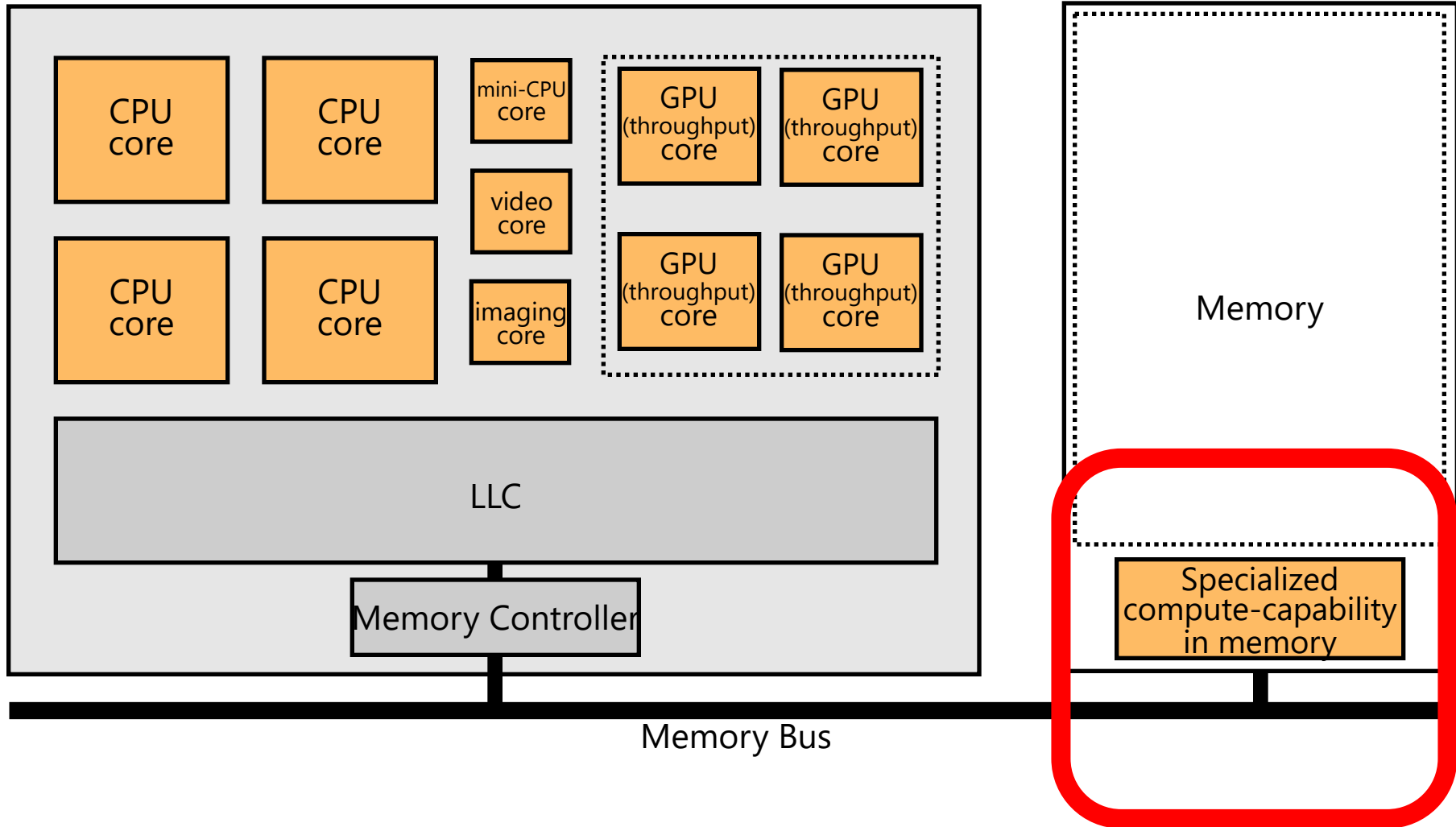
- **Location**

- Sensor, Cold Storage, Hard Disk, SSD, Main Memory, Cache, Register File, Memory Controller, Interconnect, ...

- A tuple of the three determines “PIM type”

- One can combine multiple “PIM types” in a system

# Mindset: Memory as an Accelerator



**Memory similar to a "conventional" accelerator**

# Example PIM Type: Processing using DRAM

---

- Nature: Using
- Technology: DRAM
- Location: Main Memory
  
- Processing using DRAM in Main Memory
  
  
  
  
  
  
  
  
  
- Seshadri+, “Fast Bulk Bitwise AND and OR in DRAM”, IEEE CAL 2015.
- Seshadri+, “Ambit: In-Memory Accelerator for Bulk Bitwise Operations Using Commodity DRAM Technology,” MICRO 2017.



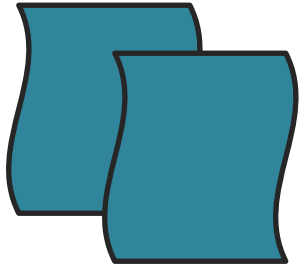
# Processing using DRAM

---

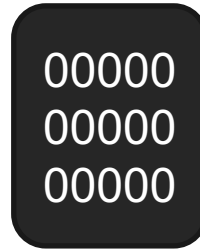
- We can support
  - Bulk bitwise AND, OR, NOT, MAJ
  - Bulk bitwise COPY and INIT/ZERO
  - True Random Number Generation; Physical Unclonable Functions
  - Lookup Table based more complex computation
- At low cost
- Using analog computation capability of DRAM
  - Idea: activating (multiple) rows performs computation
- 30-77X performance and energy improvement
  - Seshadri+, "Ambit: In-Memory Accelerator for Bulk Bitwise Operations Using Commodity DRAM Technology," MICRO 2017.
  - Seshadri+"RowClone: Fast and Efficient In-DRAM Copy and Initialization of Bulk Data," MICRO 2013.

# Starting Simple: Data Copy and Initialization

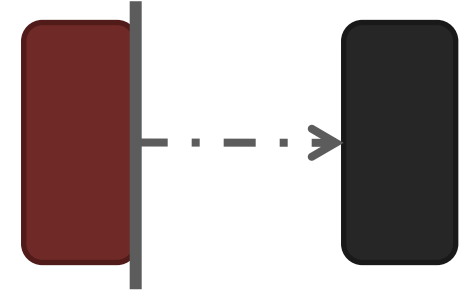
*memmove & memcpy: 5% cycles in Google's datacenter [Kanev+ ISCA'15]*



**Forking**



**Zero initialization  
(e.g., security)**



**Checkpointing**



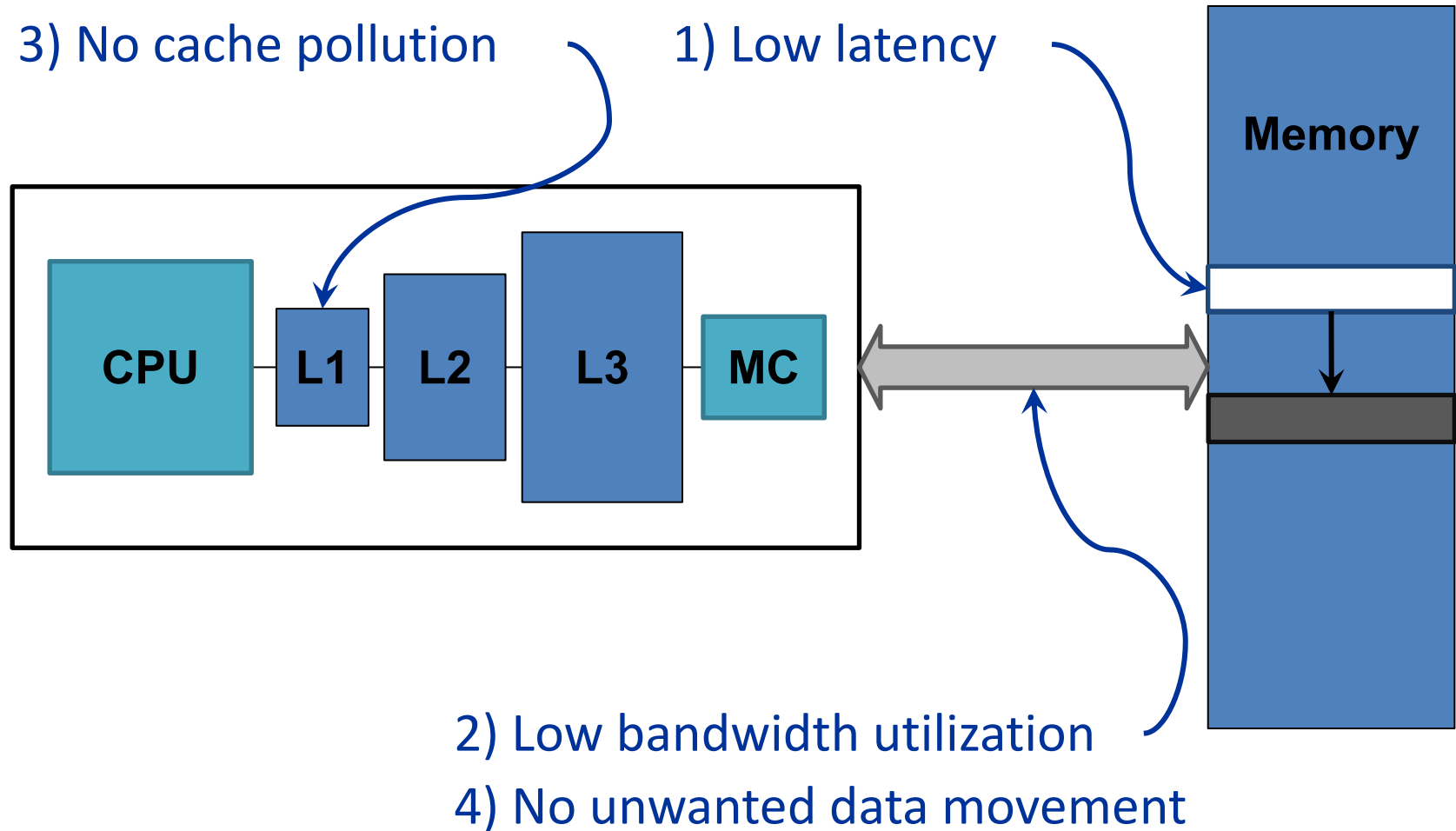
**VM Cloning  
Deduplication**



**Page Migration**

...  
Many more

# Future Systems: In-Memory Copy

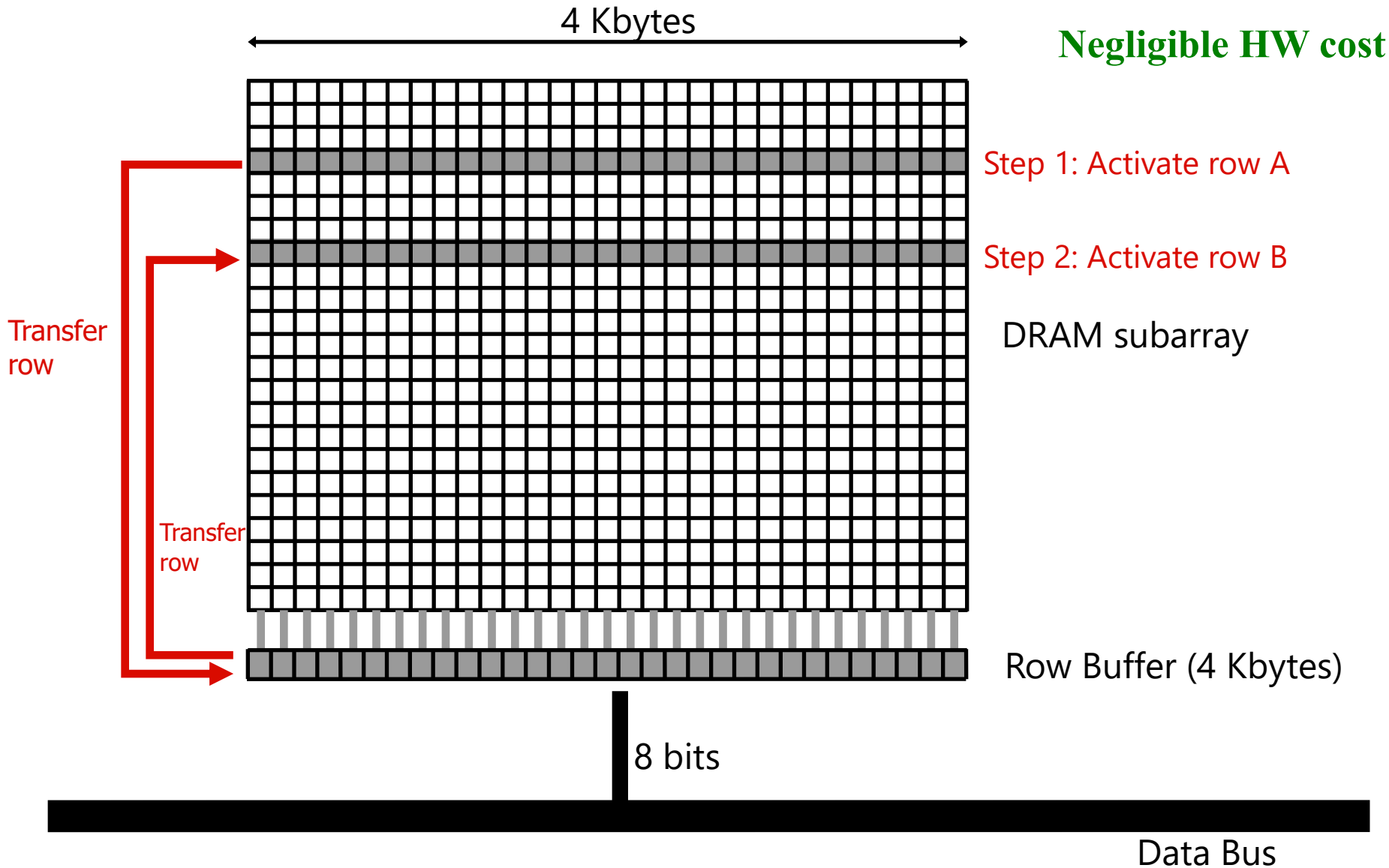


1046ns, 3.6uJ → 90ns, 0.04uJ

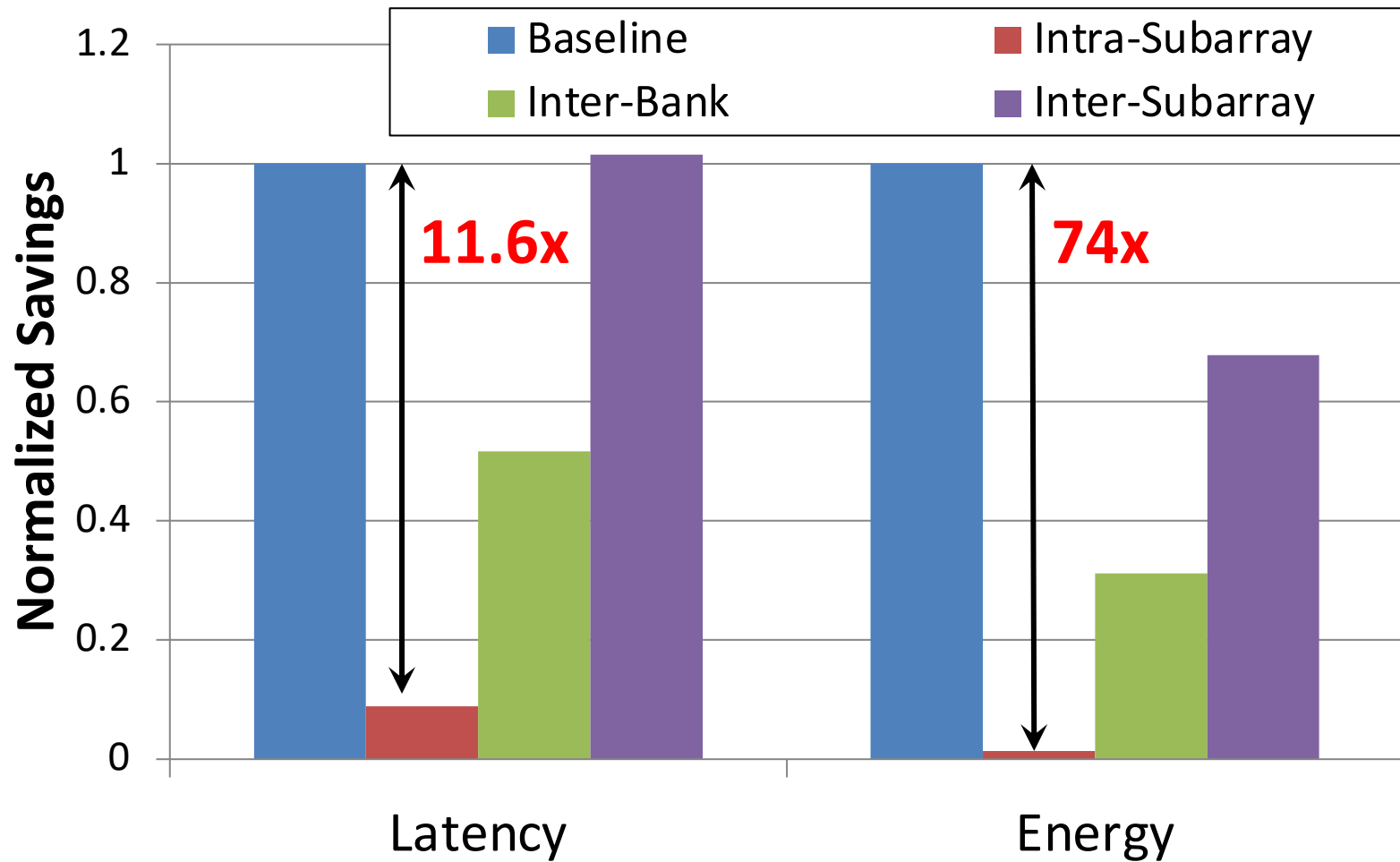
# RowClone: In-DRAM Row Copy

**Idea: Two consecutive ACTivates**

**Negligible HW cost**



# RowClone: Latency and Energy Savings



Seshadri et al., "RowClone: Fast and Efficient In-DRAM Copy and Initialization of Bulk Data," MICRO 2013.

# More on RowClone

---

- 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

Carnegie Mellon University    †Intel Pittsburgh



# RowClone in Off-the-Shelf DRAM Chips

---

- Idea: Violate DRAM timing parameters to mimic RowClone

## **ComputeDRAM: In-Memory Compute Using Off-the-Shelf DRAMs**

Fei Gao

feig@princeton.edu

Department of Electrical Engineering  
Princeton University

Georgios Tziantzioulis

georgios.tziantzioulis@princeton.edu

Department of Electrical Engineering  
Princeton University

David Wentzlaff

wentzlaf@princeton.edu

Department of Electrical Engineering  
Princeton University

# Real Processing Using Memory Prototype

---

- End-to-end RowClone & TRNG using off-the-shelf DRAM chips
- Idea: Violate DRAM timing parameters to mimic RowClone

## **PiDRAM: A Holistic End-to-end FPGA-based Framework for Processing-in-DRAM**

Ataberk Olgun<sup>§†</sup>

Juan Gómez Luna<sup>§</sup>

Konstantinos Kanellopoulos<sup>§</sup>

Behzad Salami<sup>§\*</sup>

Hasan Hassan<sup>§</sup>

Oğuz Ergin<sup>†</sup>

Onur Mutlu<sup>§</sup>

<sup>§</sup>ETH Zürich

<sup>†</sup>TOBB ETÜ

<sup>\*</sup>BSC

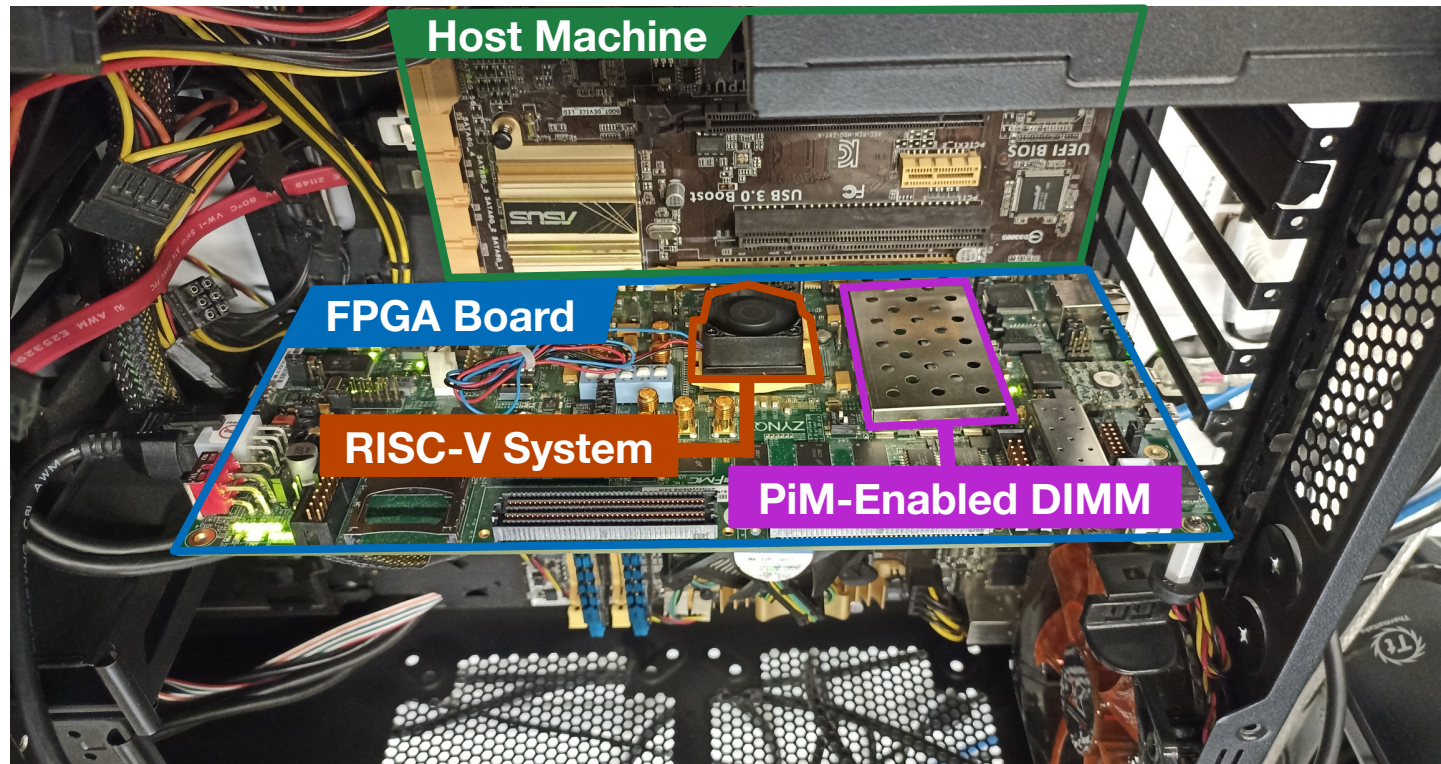
<https://arxiv.org/pdf/2111.00082.pdf>

<https://github.com/cmu-safari/pidram>

<https://www.youtube.com/watch?v=qeukNs5XI3g&t=4192s>

# Real Processing-using-Memory Prototype

---



<https://arxiv.org/pdf/2111.00082.pdf>

<https://github.com/cmu-safari/pidram>

<https://www.youtube.com/watch?v=qeukNs5XI3g&t=4192s>

# Real Processing-using-Memory Prototype

☰ README.md

## Building a PiDRAM Prototype

To build PiDRAM's prototype on Xilinx ZC706 boards, developers need to use the two sub-projects in this directory. `fpga-zynq` is a repository branched off of [UCB-BAR's fpga-zynq](#) repository. We use `fpga-zynq` to generate rocket chip designs that support end-to-end DRAM PuM execution. `controller-hardware` is where we keep the main Vivado project and Verilog sources for PiDRAM's memory controller and the top level system design.

## Rebuilding Steps

1. Navigate into `fpga-zynq` and read the README file to understand the overall workflow of the repository
  - Follow the readme in `fpga-zynq/rocket-chip/riscv-tools` to install dependencies
2. Create the Verilog source of the rocket chip design using the `ZynqCopyFPGAConfig`
  - Navigate into `zc706`, then run `make rocket CONFIG=ZynqCopyFPGAConfig -j<number of cores>`
3. Copy the generated Verilog file (should be under `zc706/src`) and overwrite the same file in `controller-hardware/source/hdl/impl/rocket-chip`
4. Open the Vivado project in `controller-hardware/Vivado_Project` using Vivado 2016.2
5. Generate a bitstream
6. Copy the bitstream (`system_top.bit`) to `fpga-zynq/zc706`
7. Use the `./build_script.sh` to generate the new `boot.bin` under `fpga-images-zc706`, you can use this file to program the FPGA using the SD-Card
  - For details, follow the relevant instructions in `fpga-zynq/README.md`

You can run programs compiled with the RISC-V Toolchain supplied within the `fpga-zynq` repository. To install the toolchain, follow the instructions under `fpga-zynq/rocket-chip/riscv-tools`.

## Generating DDR3 Controller IP sources

We cannot provide the sources for the Xilinx PHY IP we use in PiDRAM's memory controller due to licensing issues. We describe here how to regenerate them using Vivado 2016.2. First, you need to generate the IP RTL files:

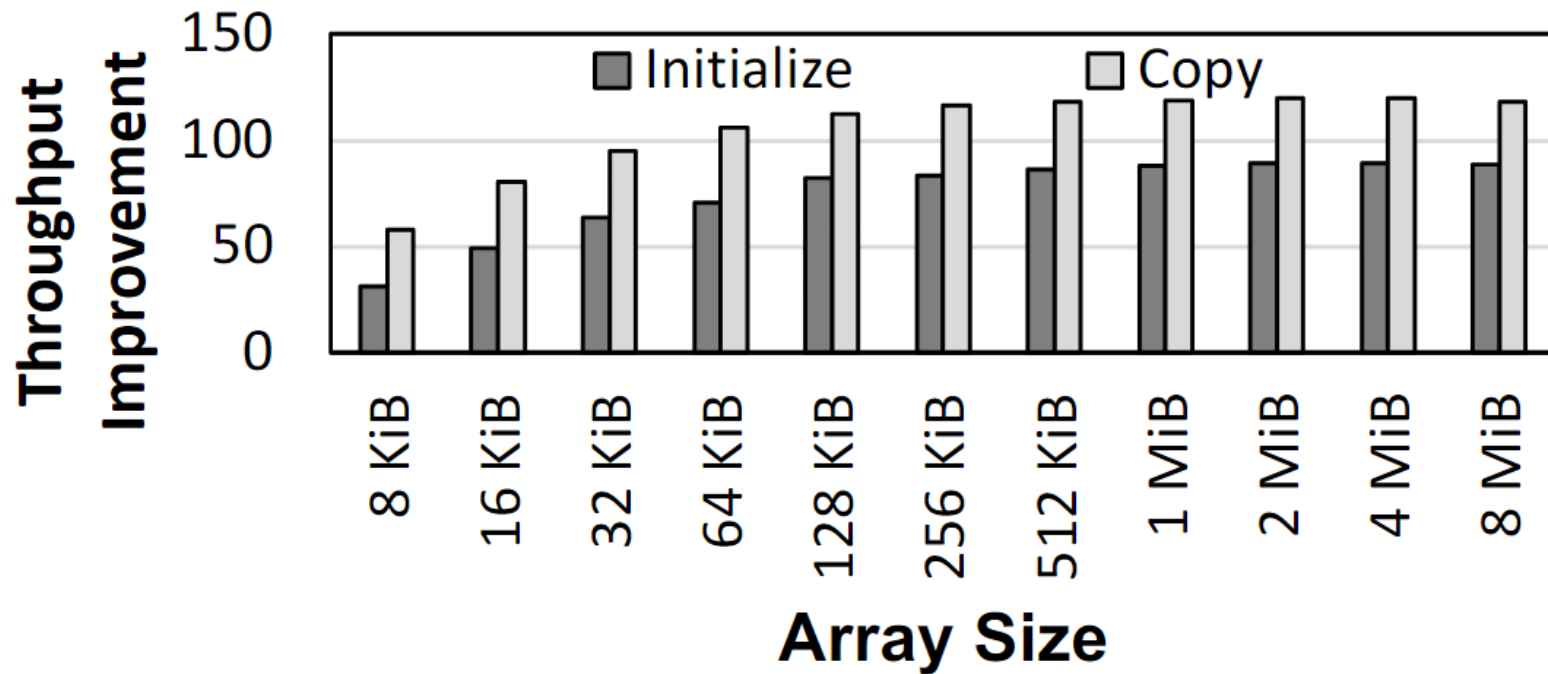
- 1- Open IP Catalog
- 2- Find "Memory Interface Generator (MIG 7 Series)" IP and double click

<https://arxiv.org/pdf/2111.00082.pdf>

<https://github.com/cmu-safari/pidram>

<https://www.youtube.com/watch?v=qeukNs5XI3g&t=4192s>

# Microbenchmark Copy/Initialization Throughput



**In-DRAM Copy and Initialization  
improve throughput by 119x and 89x**

# More on PiDRAM

- Ataberk Olgun, Juan Gomez Luna, Konstantinos Kanellopoulos, Behzad Salami, Hasan Hassan, Oguz Ergin, and Onur Mutlu,  
**["PiDRAM: A Holistic End-to-end FPGA-based Framework for Processing-in-DRAM"](#)**  
*[ACM Transactions on Architecture and Code Optimization \(TACO\)](#)*, March 2023.  
[\[arXiv version\]](#)  
Presented at the [18th HiPEAC Conference](#), Toulouse, France, January 2023.  
[\[Slides \(pptx\) \(pdf\)\]](#)  
[\[Longer Lecture Slides \(pptx\) \(pdf\)\]](#)  
[\[Lecture Video \(40 minutes\)\]](#)  
[\[PiDRAM Source Code\]](#)

## PiDRAM: A Holistic End-to-end FPGA-based Framework for Processing-in-DRAM

Ataberk Olgun<sup>§</sup>      Juan Gómez Luna<sup>§</sup>      Konstantinos Kanellopoulos<sup>§</sup>      Behzad Salami<sup>§</sup>  
Hasan Hassan<sup>§</sup>      Oğuz Ergin<sup>†</sup>      Onur Mutlu<sup>§</sup>

<sup>§</sup>ETH Zürich

<sup>†</sup>TOBB University of Economics and Technology



# Lecture on RowClone & Processing using DRAM

Mindset: Memory as an Accelerator

The diagram illustrates a system architecture where memory is treated as an accelerator. On the left, a large gray box represents the system, containing several components: four 'CPU core' blocks, one 'mini-CPU core', one 'video core', one 'imaging core', and four 'GPU (throughput) core' blocks. Below these is a large 'LLC' (Last Level Cache) block, which is connected to a 'Memory Controller' block. The 'Memory Controller' is connected to a 'Memory Bus'. To the right of the system box is a large 'Memory' block. A red rounded rectangle highlights a 'Specialized compute-capability in memory' block within the memory, which is also connected to the 'Memory Bus'. A video player interface is overlaid on the diagram, showing a play button, a progress bar at 43:48 / 2:03:45, and a red text overlay that reads 'Memory similar to a "conventional" accelerator'. A Zoom logo is visible in the bottom right corner of the video player.

Memory similar to a "conventional" accelerator

DEPARTMENT OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING (D-ITET)

Seminar in Computer Arch. - Meeting 3: RowClone: In-Memory Data Copy and Initialization (Fall 2021)

292 views • Streamed live on Oct 7, 2021

21 0 SHARE SAVE ...



Onur Mutlu Lectures  
19.1K subscribers

SUBSCRIBED



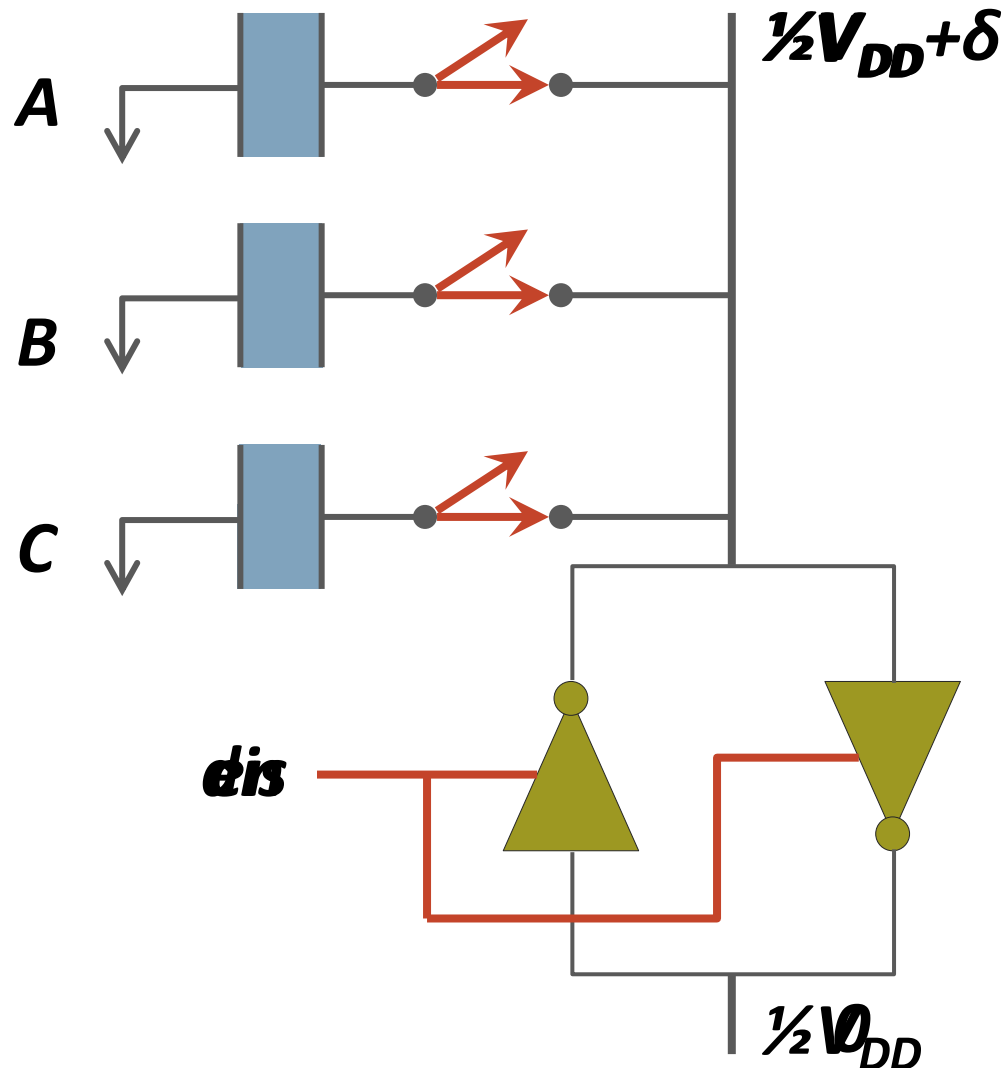
[https://www.youtube.com/watch?v=n6Pwg1qax\\_E&list=PL5Q2soXY2Zi\\_7UBNmC9B8Yr5JSwTG9yH4&index=4](https://www.youtube.com/watch?v=n6Pwg1qax_E&list=PL5Q2soXY2Zi_7UBNmC9B8Yr5JSwTG9yH4&index=4)

# (Truly) In-Memory Computation

---

- We can support in-DRAM AND, OR, NOT, MAJ
- At low cost
- Using analog computation capability of DRAM
  - Idea: activating multiple rows performs computation
- 30-60X performance and energy improvement
  - Seshadri+, “Ambit: In-Memory Accelerator for Bulk Bitwise Operations Using Commodity DRAM Technology,” MICRO 2017.
- New memory technologies enable even more opportunities
  - Memristors, resistive RAM, phase change mem, STT-MRAM, ...
  - Can operate on data with minimal movement

# In-DRAM AND/OR: Triple Row Activation

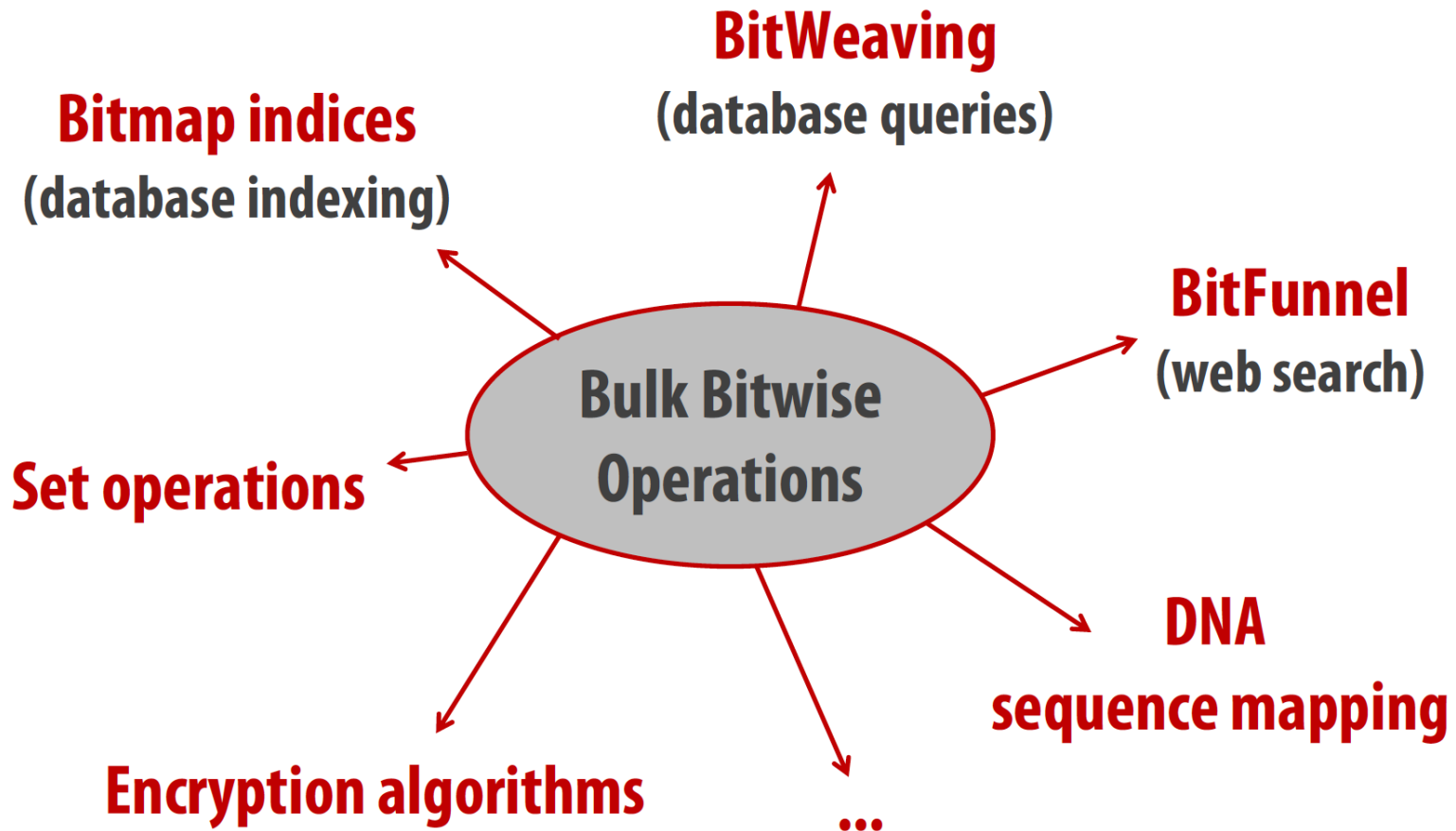


**Final State**  
 **$AB + BC + AC$**

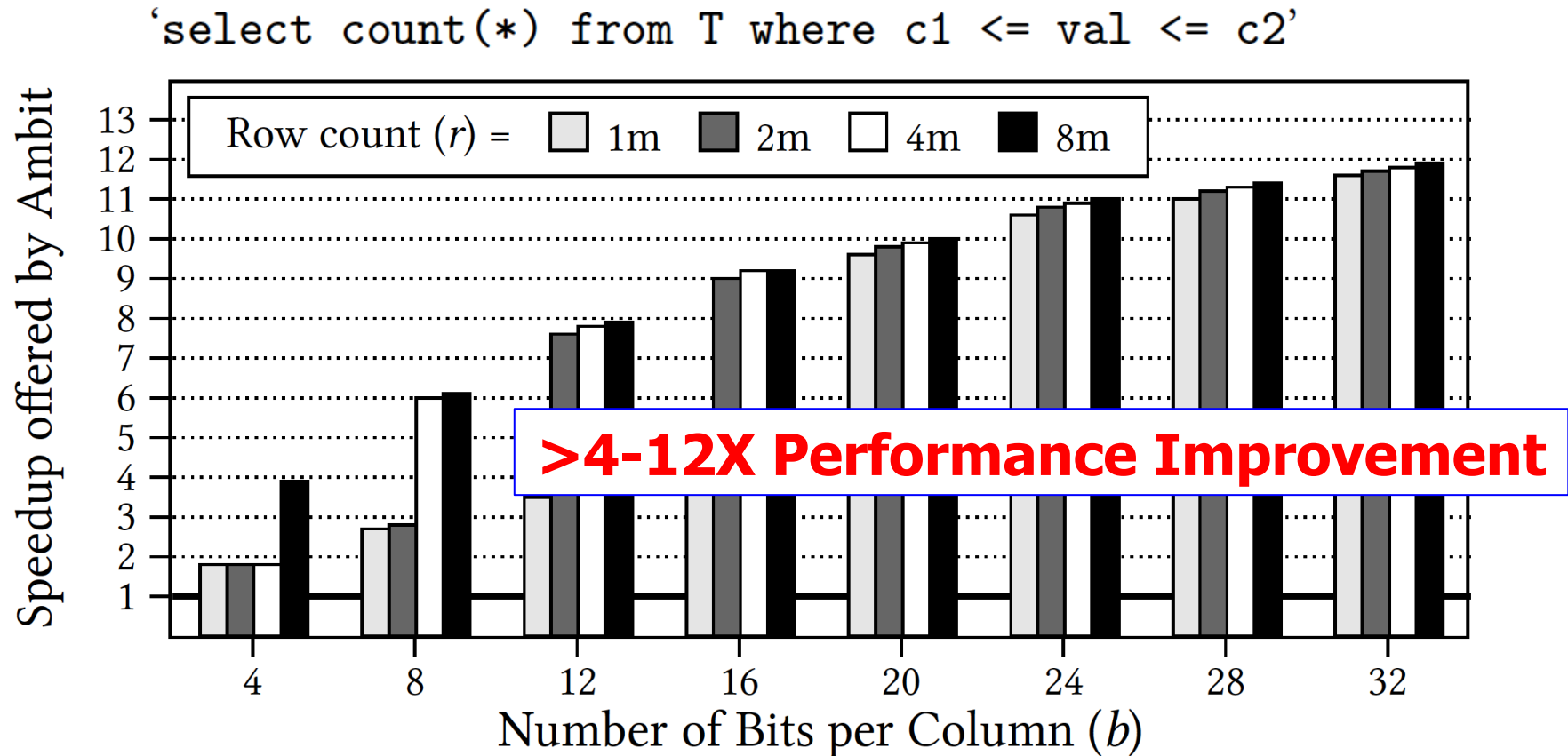
**$C(A + B) +$   
 **$\sim C(AB)$****

# Bulk Bitwise Operations in Workloads

---



# In-DRAM Acceleration of Database Queries



**Figure 11: Speedup offered by Ambit over baseline CPU with SIMD for BitWeaving**

Seshadri+, "Ambit: In-Memory Accelerator for Bulk Bitwise Operations using Commodity DRAM Technology," MICRO 2017.

# More on Ambit

---

- Vivek Seshadri, Donghyuk Lee, Thomas Mullins, Hasan Hassan, Amirali Boroumand, Jeremie Kim, Michael A. Kozuch, Onur Mutlu, Phillip B. Gibbons, and Todd C. Mowry,  
**"Ambit: In-Memory Accelerator for Bulk Bitwise Operations Using Commodity DRAM Technology"**  
*Proceedings of the 50th International Symposium on Microarchitecture (MICRO)*, Boston, MA, USA, October 2017.  
[\[Slides \(pptx\) \(pdf\)\]](#) [\[Lightning Session Slides \(pptx\) \(pdf\)\]](#) [\[Poster \(pptx\) \(pdf\)\]](#)

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

Vivek Seshadri<sup>1,5</sup> Donghyuk Lee<sup>2,5</sup> Thomas Mullins<sup>3,5</sup> Hasan Hassan<sup>4</sup> Amirali Boroumand<sup>5</sup>  
Jeremie Kim<sup>4,5</sup> Michael A. Kozuch<sup>3</sup> Onur Mutlu<sup>4,5</sup> Phillip B. Gibbons<sup>5</sup> Todd C. Mowry<sup>5</sup>

<sup>1</sup>Microsoft Research India   <sup>2</sup>NVIDIA Research   <sup>3</sup>Intel   <sup>4</sup>ETH Zürich   <sup>5</sup>Carnegie Mellon University



# In-DRAM Bulk Bitwise Execution

---

- 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

---

- 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

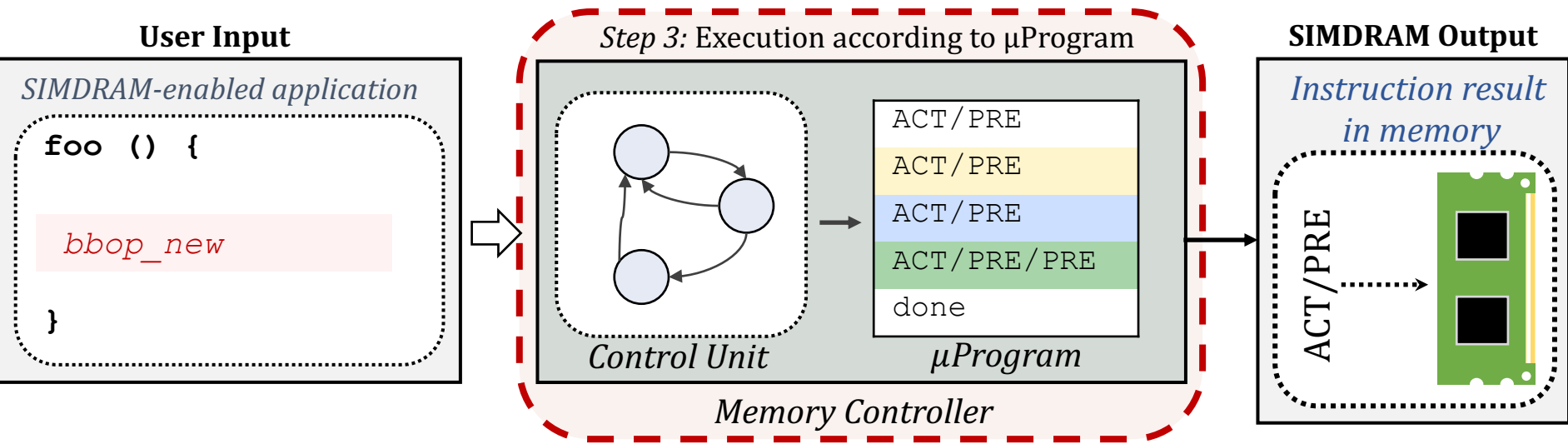
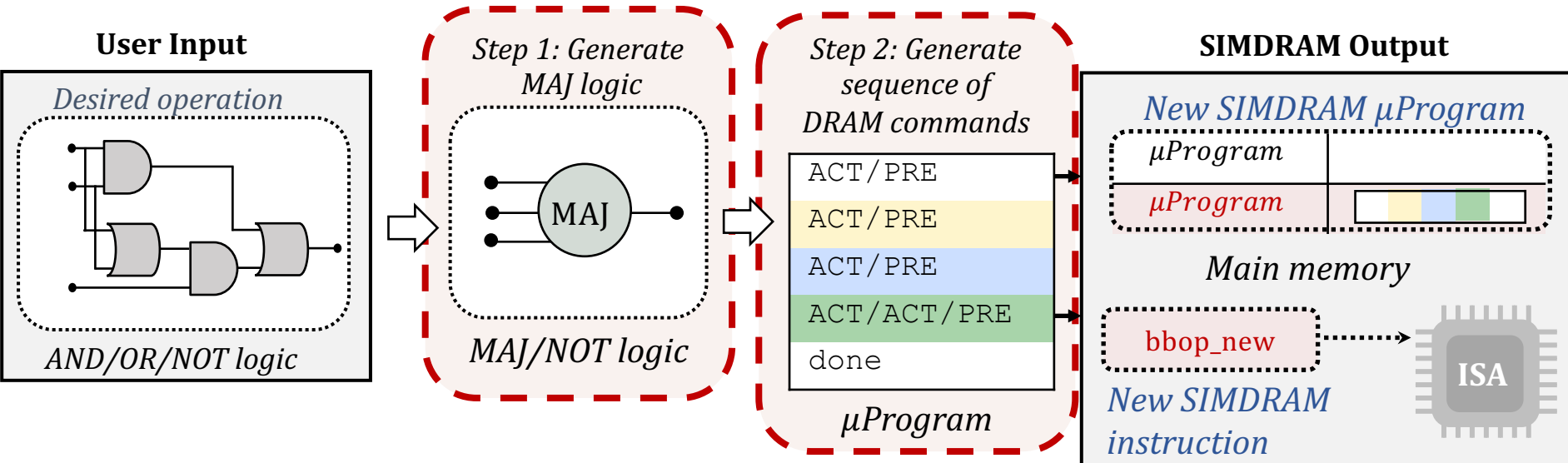
|                                    |                                   |                             |                                  |
|------------------------------------|-----------------------------------|-----------------------------|----------------------------------|
| *Nastaran Hajinazar <sup>1,2</sup> | *Geraldo F. Oliveira <sup>1</sup> | Sven Gregorio <sup>1</sup>  | João Dinis Ferreira <sup>1</sup> |
| Nika Mansouri Ghiasi <sup>1</sup>  | Minesh Patel <sup>1</sup>         | Mohammed Alser <sup>1</sup> | Saugata Ghose <sup>3</sup>       |
|                                    | Juan Gómez-Luna <sup>1</sup>      | Onur Mutlu <sup>1</sup>     |                                  |

<sup>1</sup>ETH Zürich

<sup>2</sup>Simon Fraser University

<sup>3</sup>University of Illinois at Urbana–Champaign

# SIMDRAM Framework: Overview



# SIMDRAM Key Results

Evaluated on:

- 16 complex in-DRAM operations
- 7 commonly-used real-world applications

**SIMDRAM provides:**

- **88×** and **5.8×** the **throughput** of a **CPU** and a **high-end GPU**, respectively, over **16 operations**
- **257×** and **31×** the **energy efficiency** of a **CPU** and a **high-end GPU**, respectively, over **16 operations**
- **21×** and **2.1×** the **performance** of a **CPU** and a **high-end GPU**, over **seven real-world applications**

**SAFARI**

# More on SIMD RAM

---

- 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 <sup>1,2</sup> | *Geraldo F. Oliveira <sup>1</sup> | Sven Gregorio <sup>1</sup>  | João Dinis Ferreira <sup>1</sup> |
| Nika Mansouri Ghiasi <sup>1</sup>  | Minesh Patel <sup>1</sup>         | Mohammed Alser <sup>1</sup> | Saugata Ghose <sup>3</sup>       |
|                                    | Juan Gómez-Luna <sup>1</sup>      | Onur Mutlu <sup>1</sup>     |                                  |

<sup>1</sup>ETH Zürich

<sup>2</sup>Simon Fraser University

<sup>3</sup>University of Illinois at Urbana–Champaign

# MIMDRAM: More Flexible Processing using DRAM

---

## ■ To appear at HPCA 2024

### **MIMDRAM: An End-to-End Processing-Using-DRAM System for High-Throughput, Energy-Efficient and Programmer-Transparent Multiple-Instruction Multiple-Data Computing**

Geraldo F. Oliveira<sup>†</sup>      Ataberk Olgun<sup>†</sup>      Abdullah Giray Yağlıkçı<sup>†</sup>      F. Nisa Bostancı<sup>†</sup>  
Juan Gómez-Luna<sup>†</sup>      Saugata Ghose<sup>‡</sup>      Onur Mutlu<sup>†</sup>

<sup>†</sup> *ETH Zürich*

<sup>‡</sup> *Univ. of Illinois Urbana-Champaign*

*Our **goal** is to design a flexible PUD system that overcomes the limitations caused by the large and rigid granularity of PUD. To this end, we propose MIMDRAM, a hardware/software co-designed PUD system that introduces new mechanisms to allocate and control only the necessary resources for a given PUD operation. The key idea of MIMDRAM is to leverage fine-grained DRAM (i.e., the ability to independently access smaller segments of a large DRAM row) for PUD computation. MIMDRAM exploits this key idea to enable a multiple-instruction multiple-data (MIMD) execution model in each DRAM subarray (and SIMD execution within each DRAM row segment).*

# 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<sup>§</sup>

Gabriel Falcao<sup>†</sup>

Juan Gómez-Luna<sup>§</sup>

Mohammed Alser<sup>§</sup>

Lois Orosa<sup>§∇</sup>

Mohammad Sadrosadati<sup>§</sup>

Jeremie S. Kim<sup>§</sup>

Geraldo F. Oliveira<sup>§</sup>

Taha Shahroodi<sup>‡</sup>

Anant Nori<sup>\*</sup>

Onur Mutlu<sup>§</sup>

<sup>§</sup>ETH Zürich

<sup>†</sup>IT, University of Coimbra

<sup>∇</sup>Galicia Supercomputing Center

<sup>‡</sup>TU Delft

<sup>\*</sup>Intel



# 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<sup>†§</sup>

Minesh Patel<sup>§</sup>

Hasan Hassan<sup>§</sup>

Onur Mutlu<sup>§†</sup>

<sup>†</sup>Carnegie Mellon University

<sup>§</sup>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<sup>‡§</sup>

Minesh Patel<sup>§</sup>

Hasan Hassan<sup>§</sup>

Lois Orosa<sup>§</sup>

Onur Mutlu<sup>§‡</sup>

<sup>‡</sup>Carnegie Mellon University

<sup>§</sup>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<sup>§†</sup>

Minesh Patel<sup>§</sup>

A. Giray Yağlıkçı<sup>§</sup>

Haocong Luo<sup>§</sup>

Jeremie S. Kim<sup>§</sup>

F. Nisa Bostanci<sup>§†</sup>

Nandita Vijaykumar<sup>§⊙</sup>

Oğuz Ergin<sup>†</sup>

Onur Mutlu<sup>§</sup>

<sup>§</sup>ETH Zürich

<sup>†</sup>TOBB University of Economics and Technology

<sup>⊙</sup>University of Toronto

# In-DRAM True Random Number Generation

---

- F. Nisa Bostanci, 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 Bostanci<sup>†§</sup>

Jeremie S. Kim<sup>§</sup>

Ataberk Olgun<sup>†§</sup>

Hasan Hassan<sup>§</sup>

Lois Orosa<sup>§</sup>

Oğuz Ergin<sup>†</sup>

A. Giray Yağlıkçı<sup>§</sup>

Onur Mutlu<sup>§</sup>

<sup>†</sup>*TOBB University of Economics and Technology*

<sup>§</sup>*ETH Zürich*

# DRAM Chips Are Already (Quite) Capable!

---

## ■ To appear at HPCA 2024

### Functionally-Complete Boolean Logic in Real DRAM Chips: Experimental Characterization and Analysis

İsmail Emir Yüksel   Yahya Can Tuğrul   Ataberk Olgun   F. Nisa Bostancı   A. Giray Yağlıkçı  
Geraldo F. Oliveira   Haocong Luo   Juan Gómez-Luna   Mohammad Sadrosadati   Onur Mutlu

ETH Zürich

*We experimentally demonstrate that COTS DRAM chips are capable of performing 1) functionally-complete Boolean operations: NOT, NAND, and NOR and 2) many-input (i.e., more than two-input) AND and OR operations. We present an extensive characterization of new bulk bitwise operations in 256 off-the-shelf modern DDR4 DRAM chips. We evaluate the reliability of these operations using a metric called success rate: the fraction of correctly performed bitwise operations. Among our 19 new observations, we highlight four major results. First, we can perform the NOT operation on COTS DRAM chips with 98.37% success rate on average. Second, we can perform up to 16-input NAND, NOR, AND, and OR operations on COTS DRAM chips with high reliability (e.g., 16-input NAND, NOR, AND, and OR with average success rate of 94.94%, 95.87%, 94.94%, and 95.85%, respectively). Third, data pattern only slightly*



# DRAM Chips Are Already (Quite) Capable!

---

- <https://arxiv.org/pdf/2312.02880.pdf>

## **PULSAR: Simultaneous Many-Row Activation for Reliable and High-Performance Computing in Off-the-Shelf DRAM Chips**

Ismail Emir Yuksel   Yahya Can Tugrul   F. Nisa Bostanci   Abdullah Giray Yaglikci   Ataberk Olgun  
Geraldo F. Oliveira   Melina Soysal   Haocong Luo   Juan Gomez Luna   Mohammad Sadrosadati  
Onur Mutlu

ETH Zurich

We propose PULSAR, a new technique to enable high-success-rate and high-performance PuM operations in off-the-shelf DRAM chips. PULSAR leverages our new observation that a carefully-crafted sequence of DRAM commands simultaneously activates up to 32 DRAM rows. PULSAR overcomes the limitations of existing techniques by 1) replicating the input data to improve the success rate and 2) enabling new bulk bitwise operations (e.g., many-input majority, *Multi-RowInit*, and *Bulk-Write*) to improve the performance.

# 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<sup>§∇</sup> Roknoddin Azizi<sup>§</sup> Geraldo F. Oliveira<sup>§</sup> Mohammad Sadrosadati<sup>§</sup>  
Rakesh Nadig<sup>§</sup> David Novo<sup>†</sup> Juan Gómez-Luna<sup>§</sup> Myungsuk Kim<sup>‡</sup> Onur Mutlu<sup>§</sup>

<sup>§</sup>ETH Zürich    <sup>∇</sup>POSTECH    <sup>†</sup>LIRMM, Univ. Montpellier, CNRS    <sup>‡</sup>Kyungpook National University



# Summary: Flash-Cosmos

---



The first work that enables  
in-flash multi-operand bulk bitwise operations  
with a single sensing operation and high reliability



Improves performance  
by 32x/25x/3.5x over OSP/ISP/ParaBit



Improves energy efficiency  
by 95x/13.4x/3.3x over OSP/ISP/ParaBit

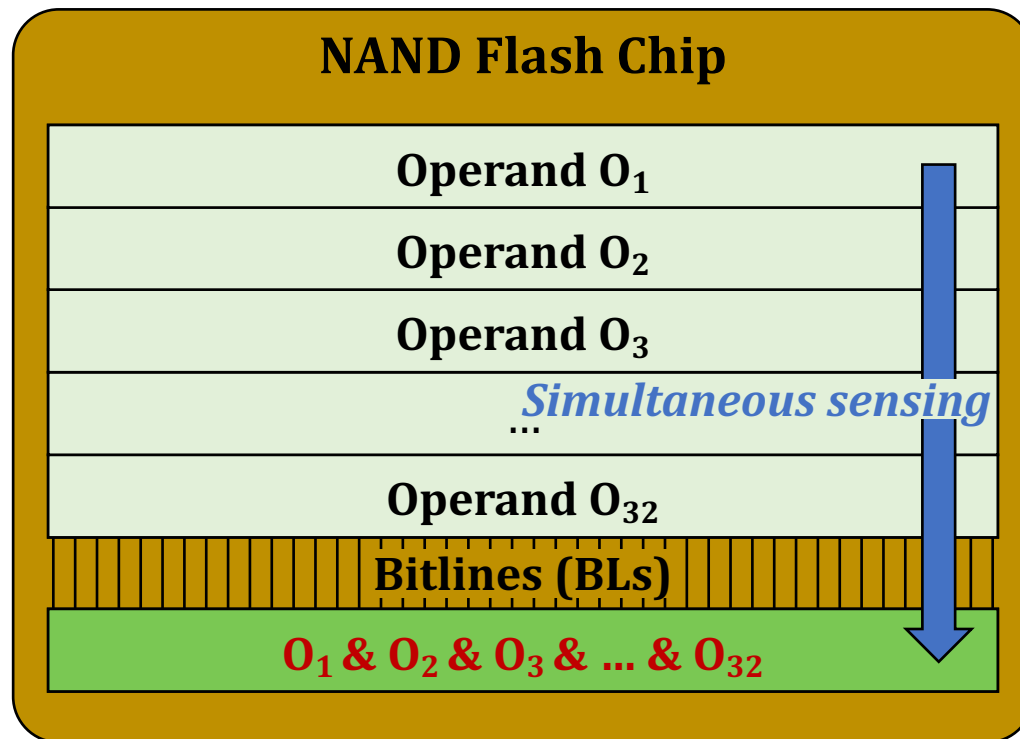


Low-cost & requires no changes to flash cell arrays

# Flash-Cosmos: Basic Ideas

## ▪ Flash-Cosmos enables

- Computation on multiple operands with a single sensing operation
- Accurate computation results by eliminating raw bit errors in stored data



# Key Ideas of Flash-Cosmos

---



**Multi-Wordline Sensing (MWS)**  
to enable in-flash bulk bitwise operations  
via a single sensing operation



**Enhanced SLC-Mode Programming (ESP)**  
to eliminate raw bit errors in stored data  
(and thus in computation results)

# Multi-Wordline Sensing (MWS): Bitwise AND

## ■ Intra-Block MWS:

Simultaneously activates multiple WLs in the same block

→ Bitwise AND of the stored data in the WLs

A bitline reads as '**1**' only when all the target cells store '**1**'  
→ Equivalent to the bitwise AND of all the target cells

*Operate  
as a resistance (1)  
or an open switch (0)*

WL<sub>2</sub>  
WL<sub>3</sub>  
WL<sub>4</sub>

Result: 0

0

0

0

BL<sub>1</sub>

BL<sub>2</sub>

BL<sub>3</sub>

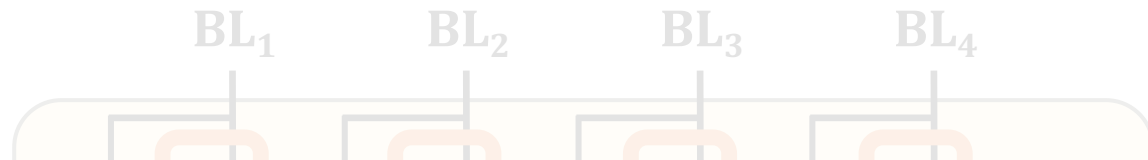
BL<sub>4</sub>

# Multi-Wordline Sensing (MWS): Bitwise AND

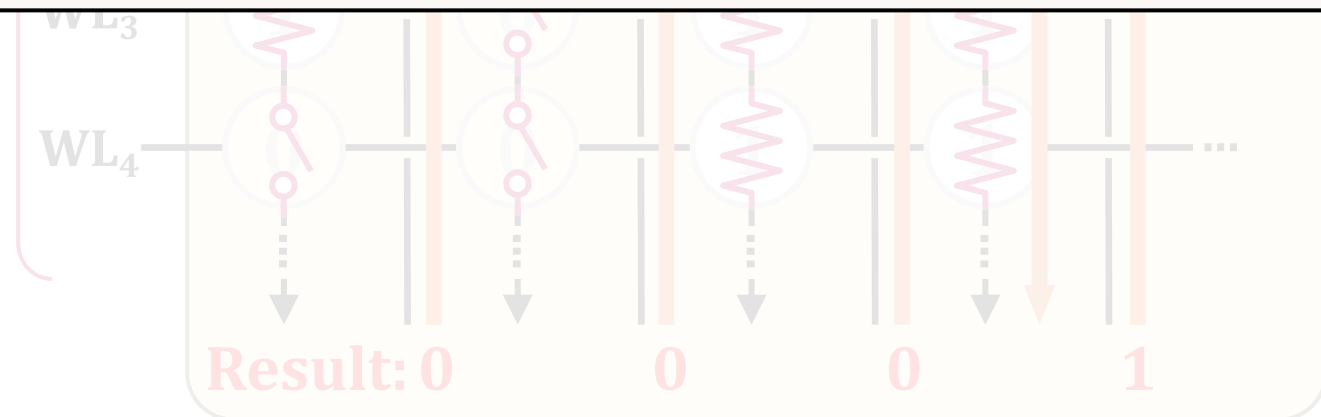
- Intra-Block MWS:

Simultaneously activates multiple WLs in the same block

→ Bitwise AND of the stored data in the WLs



**Flash-Cosmos (Intra-Block MWS)** enables bitwise AND of multiple pages in the same block via a single sensing operation

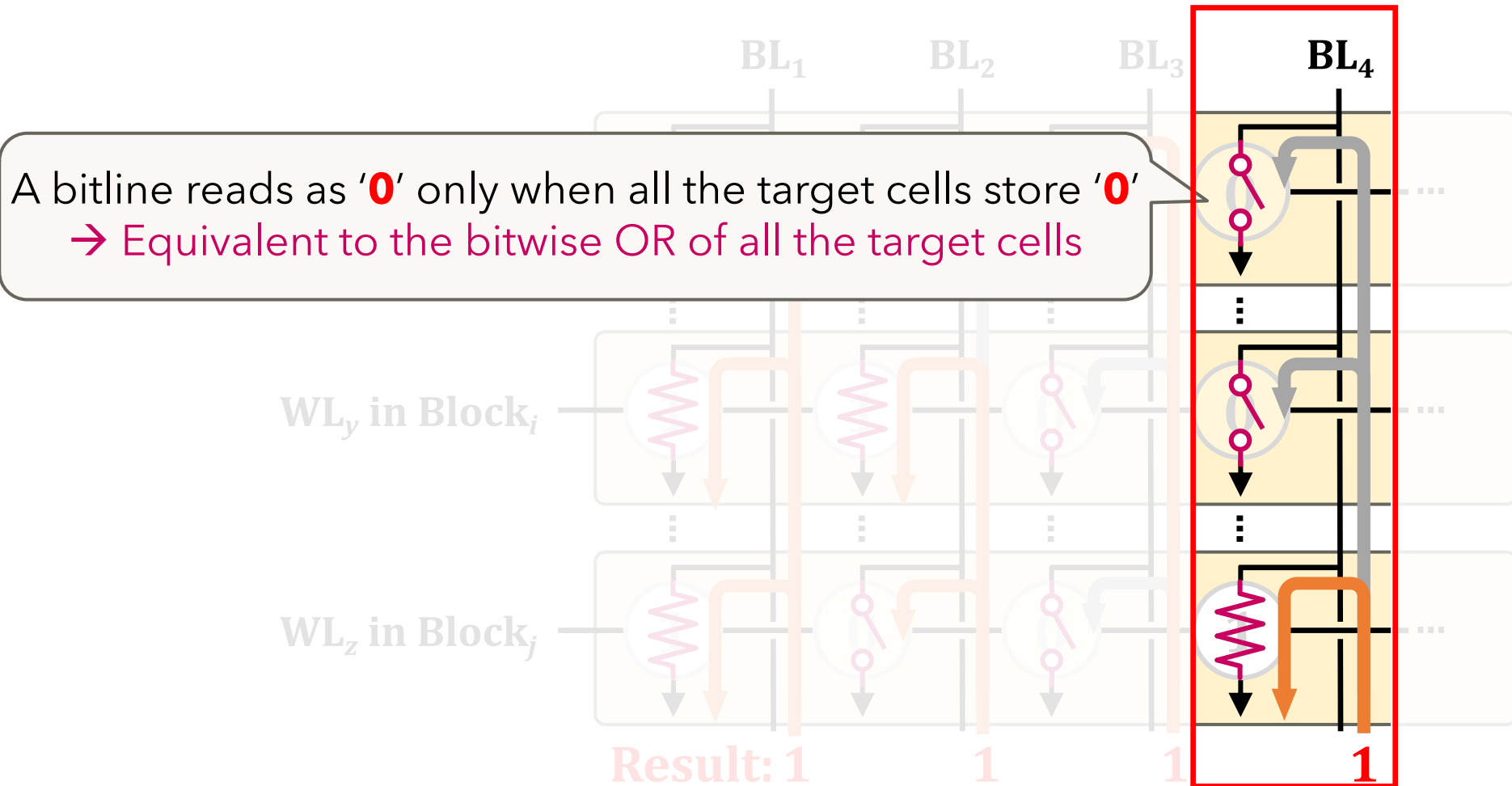


# Multi-Wordline Sensing (MWS): Bitwise OR

- **Inter-Block MWS:**

Simultaneously activates multiple WLs in different blocks

→ Bitwise OR of the stored data in the WLs





# Multi-Wordline Sensing (MWS): Bitwise OR

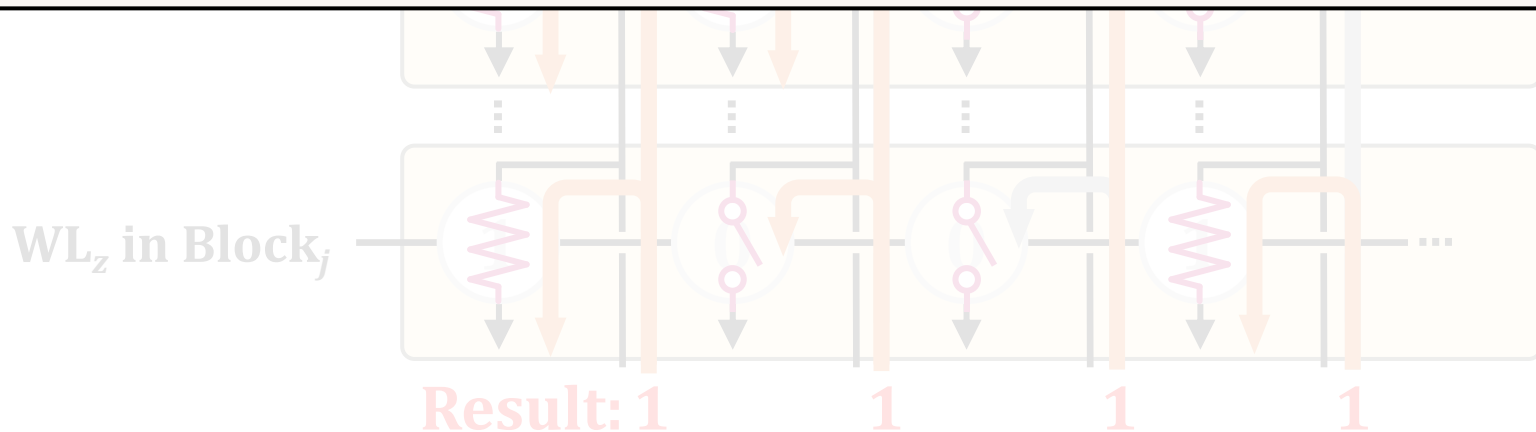
## ■ Inter-Block MWS:

Simultaneously activates multiple WLs in different blocks

→ Bitwise OR of the stored data in the WLs



**Flash-Cosmos (Inter-Block MWS)** enables  
bitwise OR of multiple pages in different blocks  
via a single sensing operation



# Other Types of Bitwise Operations

**Flash-Cosmos** also enables  
other types of bitwise operations  
(NOT/NAND/NOR/XOR/XNOR)  
leveraging **existing features** of NAND flash memory

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

Jisung Park<sup>§∇</sup> Roknoddin Azizi<sup>§</sup> Geraldo F. Oliveira<sup>§</sup> Mohammad Sadrosadati<sup>§</sup>  
Rakesh Nadig<sup>§</sup> David Novo<sup>†</sup> Juan Gómez-Luna<sup>§</sup> Myungsuk Kim<sup>‡</sup> Onur Mutlu<sup>§</sup>

<sup>§</sup>*ETH Zürich*    <sup>∇</sup>*POSTECH*    <sup>†</sup>*LIRMM, Univ. Montpellier, CNRS*    <sup>‡</sup>*Kyungpook National University*



<https://arxiv.org/abs/2209.05566.pdf>

# Results: Real-Device Characterization

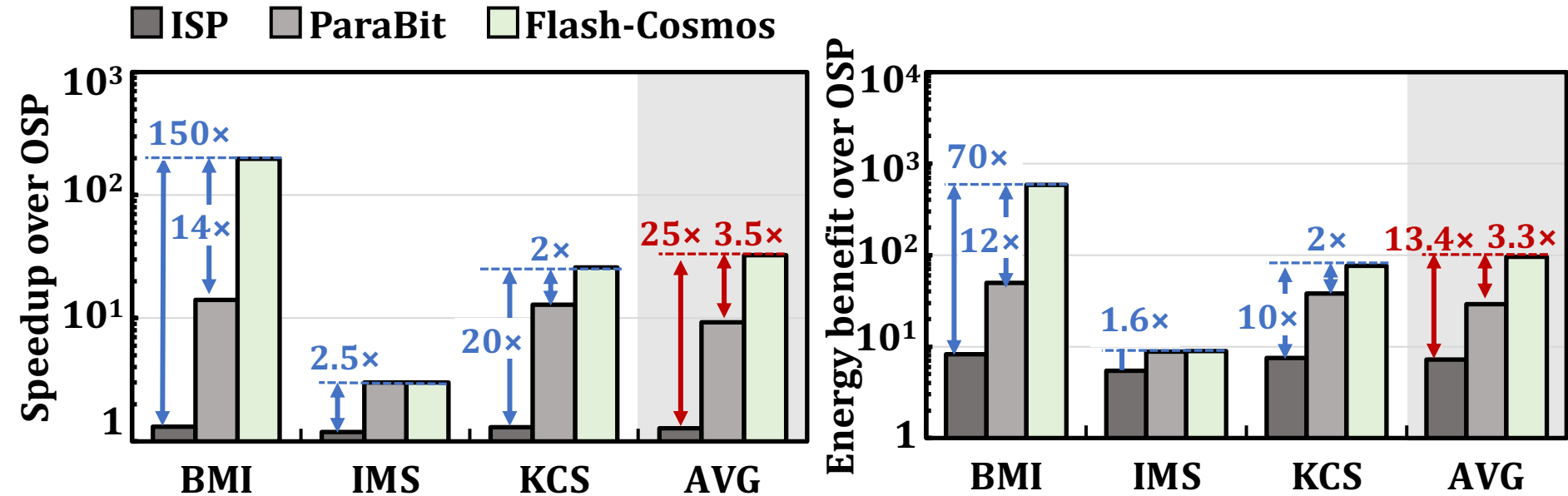
---

No changes to the cell array  
of commodity NAND flash chips

Can have many operands  
(AND: up to 48, OR: up to 4)  
with small increase in sensing latency ( $< 10\%$ )

ESP significantly improves  
the reliability of computation results  
(no observed bit error in the tested flash cells)

# Results: Performance & Energy



Flash-Cosmos provides **significant performance & energy benefits** over all the baselines

The larger the number of operands,  
the higher the performance & energy benefits

# Pinatubo: RowClone and Bitwise Ops in PCM

---

## **Pinatubo: A Processing-in-Memory Architecture for Bulk Bitwise Operations in Emerging Non-volatile Memories**

Shuangchen Li<sup>1\*</sup>, Cong Xu<sup>2</sup>, Qiaosha Zou<sup>1,5</sup>, Jishen Zhao<sup>3</sup>, Yu Lu<sup>4</sup>, and Yuan Xie<sup>1</sup>

University of California, Santa Barbara<sup>1</sup>, Hewlett Packard Labs<sup>2</sup>

University of California, Santa Cruz<sup>3</sup>, Qualcomm Inc.<sup>4</sup>, Huawei Technologies Inc.<sup>5</sup>  
{shuangchenli, yuanxie}@ece.ucsb.edu<sup>1</sup>

# Other Readings on Processing using NVM

---

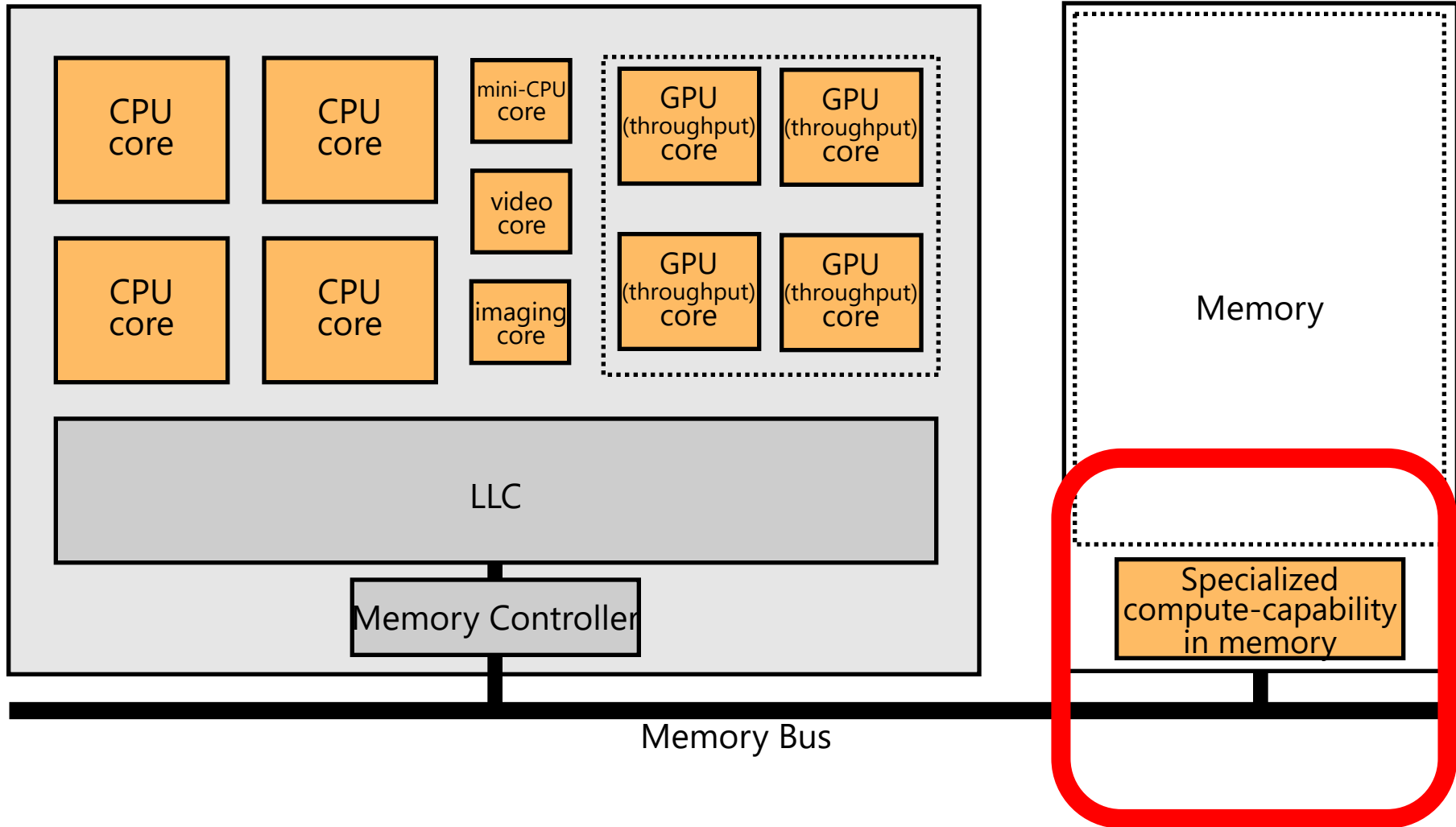
- Shafiee+, “ISAAC: A Convolutional Neural Network Accelerator with In-Situ Analog Arithmetic in Crossbars”, ISCA 2016.
- Chi+, “PRIME: A Novel Processing-in-memory Architecture for Neural Network Computation in ReRAM-based Main Memory”, ISCA 2016.
- Prezioso+, “Training and Operation of an Integrated Neuromorphic Network based on Metal-Oxide Memristors”, Nature 2015
- Ambrogio+, “Equivalent-accuracy accelerated neural-network training using analogue memory”, Nature 2018.

# Processing in Memory: Two Approaches

1. Processing **using** Memory
2. Processing **near** Memory



# Mindset: Memory as an Accelerator



**Memory similar to a "conventional" accelerator**

# Accelerating In-Memory Graph Analytics

- Large graphs are everywhere (circa 2015)



36 Million  
Wikipedia Pages



1.4 Billion  
Facebook Users

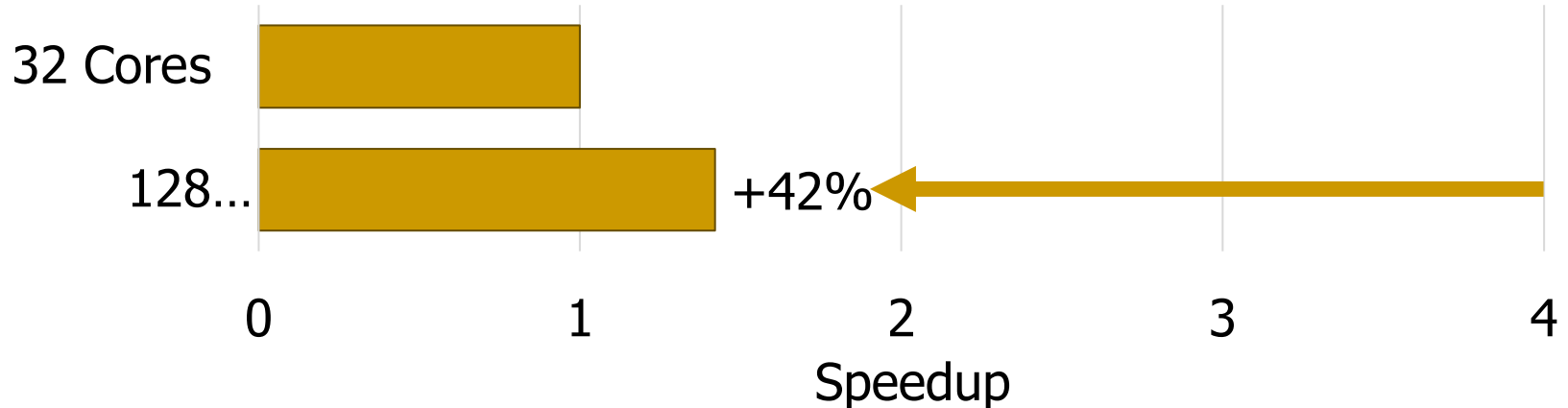


300 Million  
Twitter Users



30 Billion  
Instagram Photos

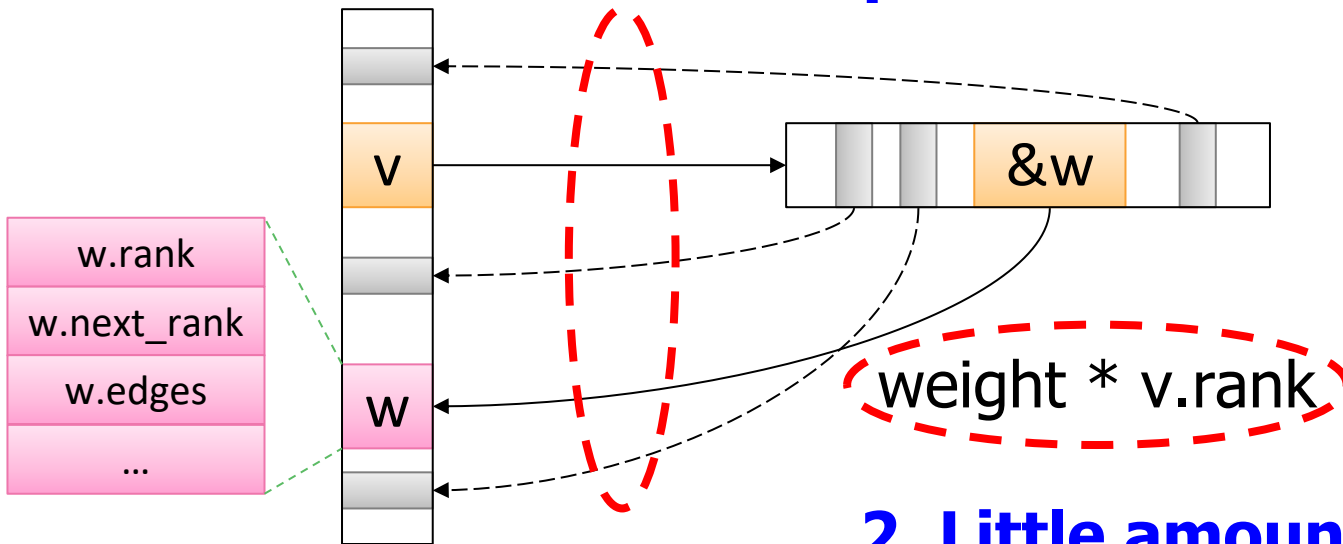
- Scalable large-scale graph processing is challenging



# Key Bottlenecks in Graph Processing

```
for (v: graph.vertices) {  
  for (w: v.successors) {  
    w.next_rank += weight * v.rank;  
  }  
}
```

## 1. Frequent random memory accesses



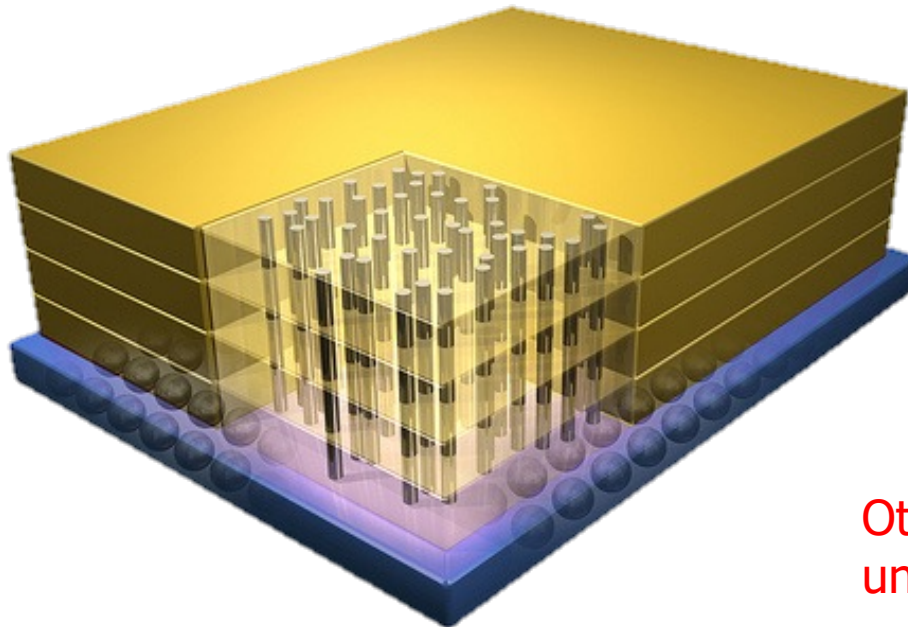
## 2. Little amount of computation

# Opportunity: 3D-Stacked Logic+Memory

---



Hybrid Memory Cube  
C O N S O R T I U M



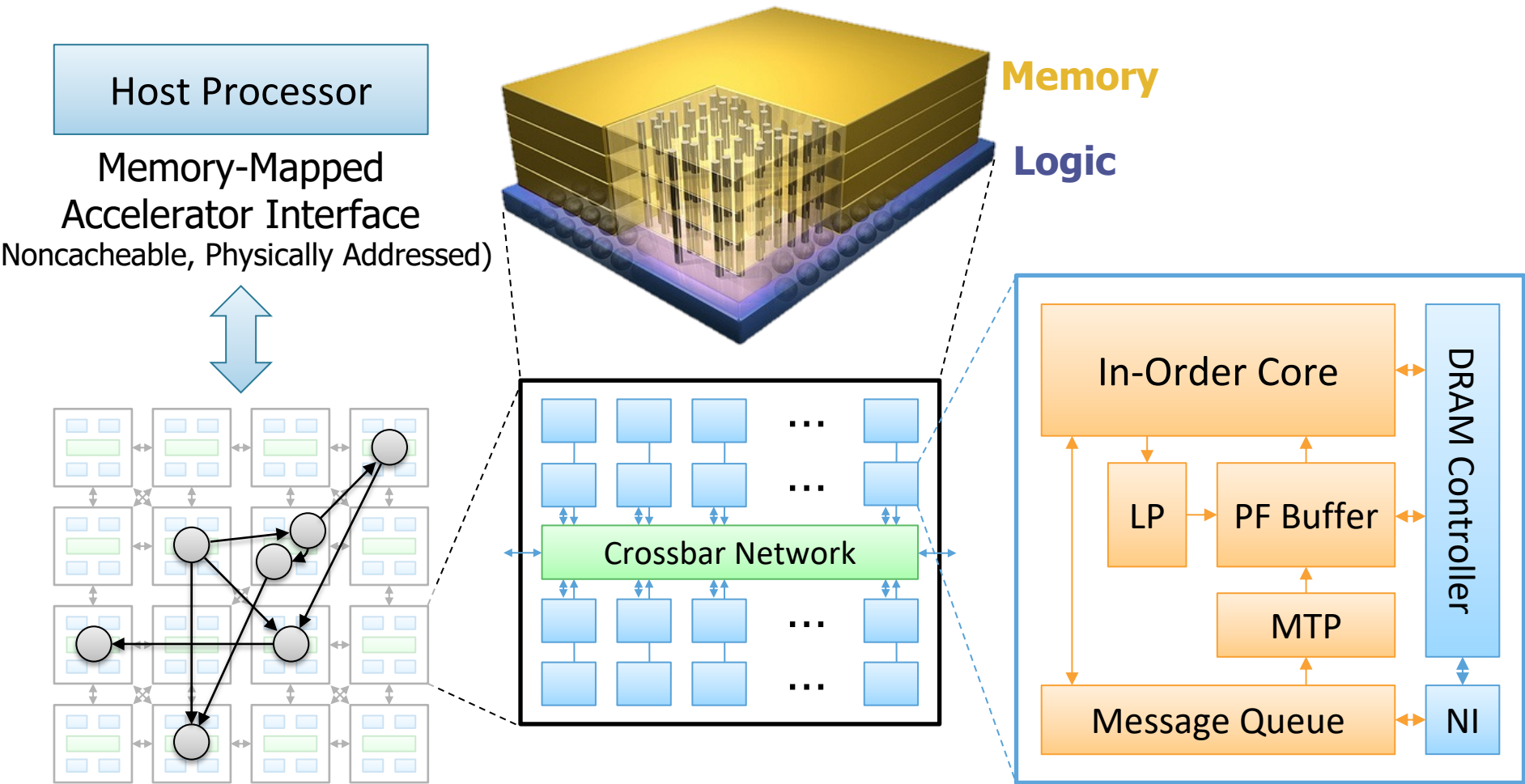
Memory

Logic

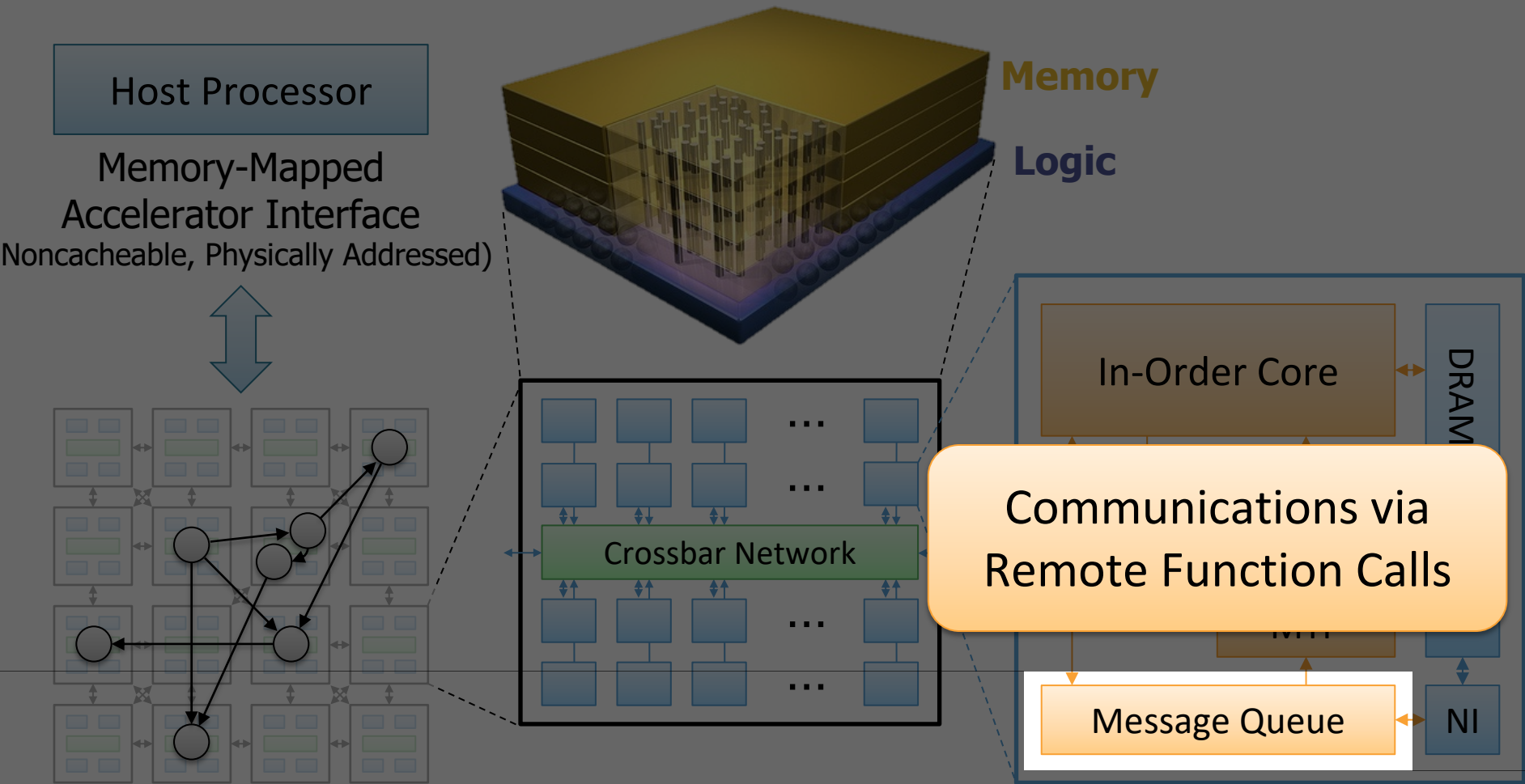
Other "True 3D" technologies  
under development

# Tesseract System for Graph Processing

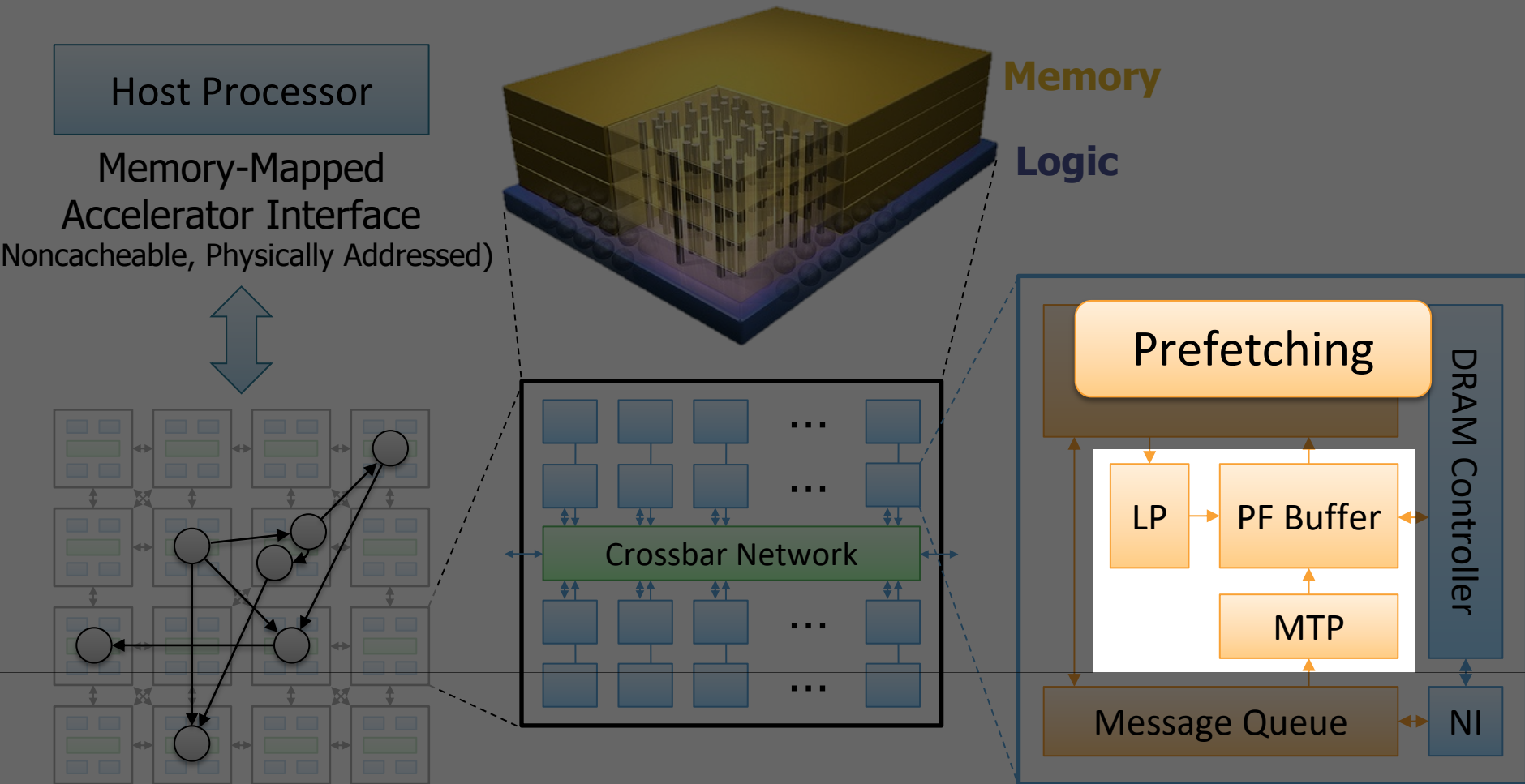
Interconnected set of 3D-stacked memory+logic chips with simple cores



# Tesseract System for Graph Processing



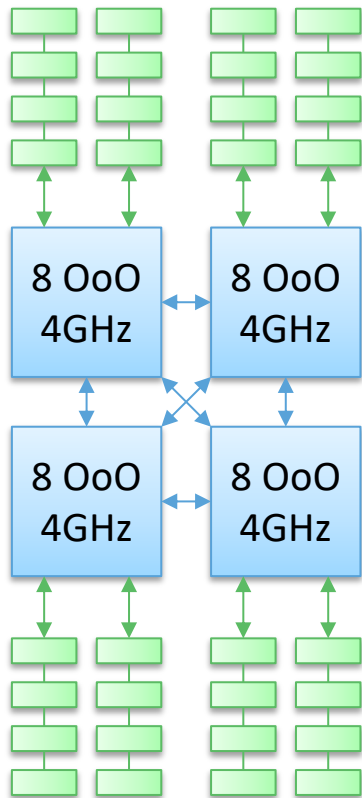
# Tesseract System for Graph Processing





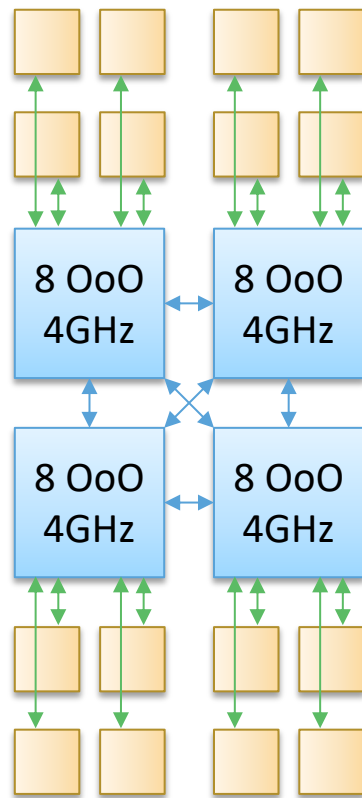
# Evaluated Systems

DDR3-OoO



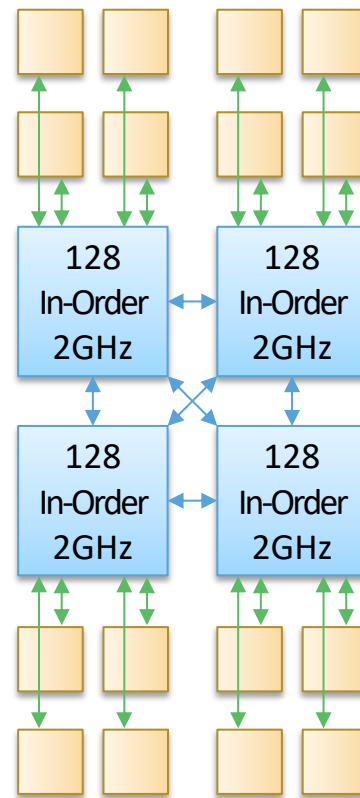
102.4GB/s

HMC-OoO



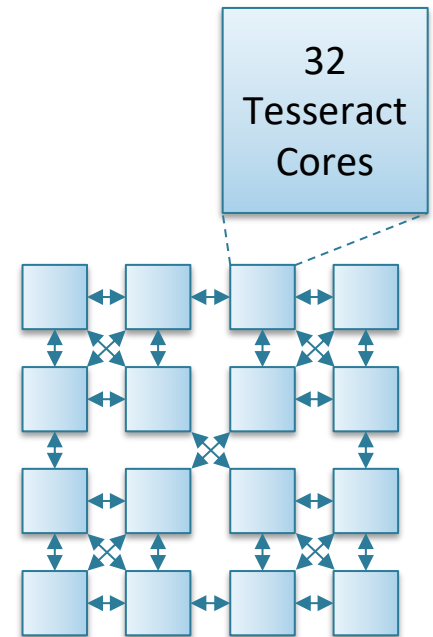
640GB/s

HMC-MC



640GB/s

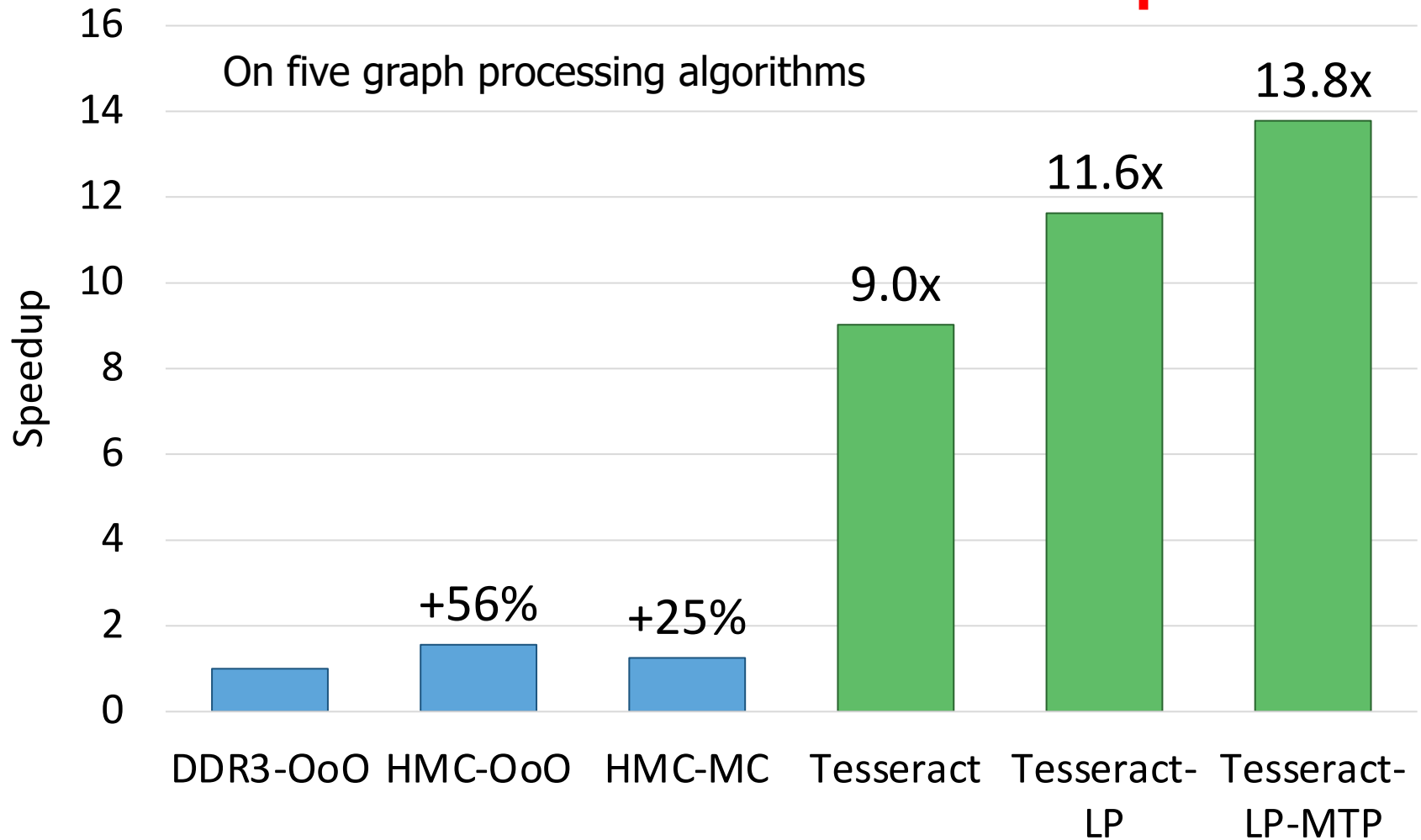
**Tesseract**



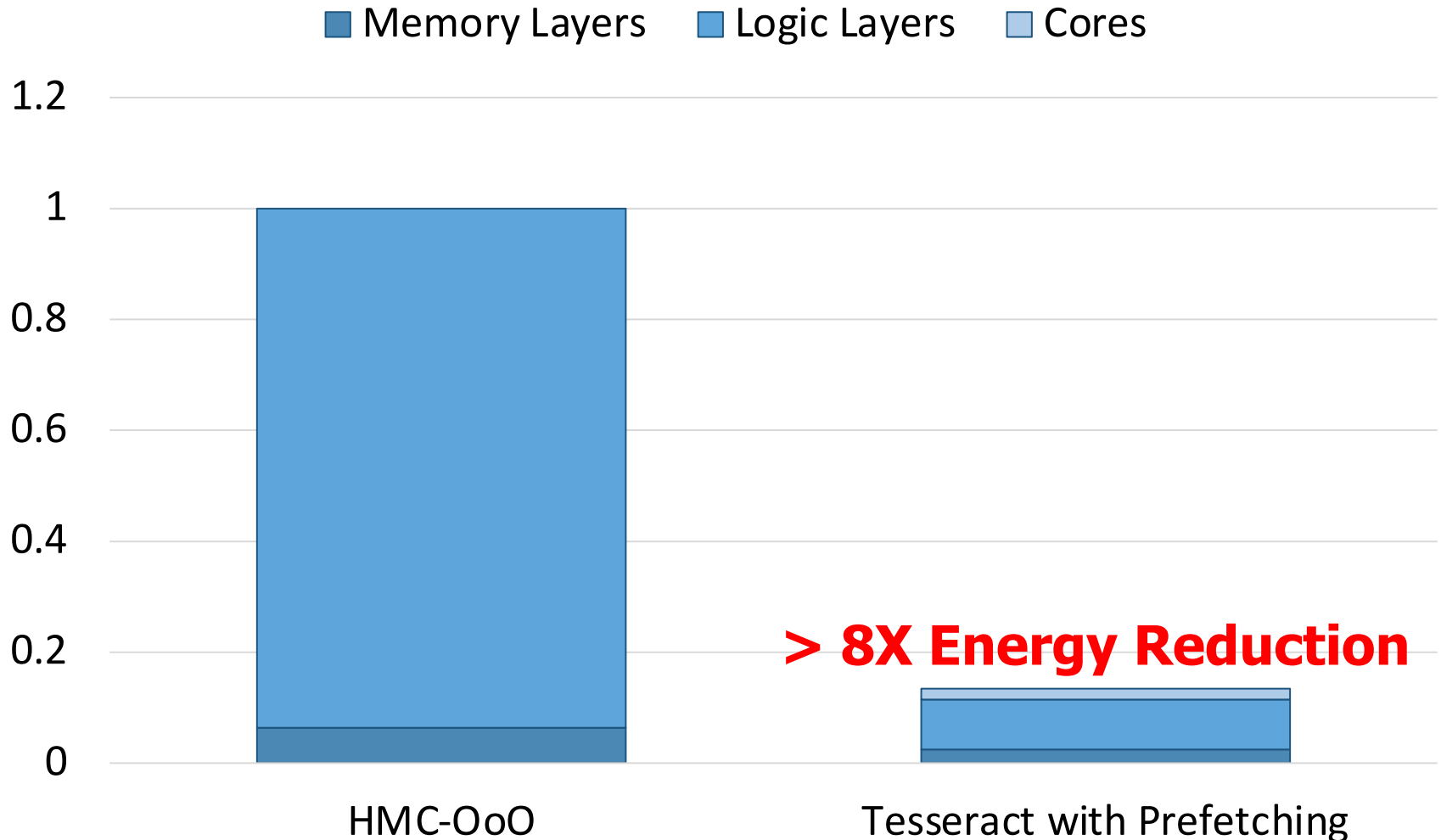
**8TB/s**

# Tesseract Graph Processing Performance

**>13X Performance Improvement**



# Tesseract Graph Processing System Energy



# More on Tesseract

---

- Junwhan Ahn, Sungpack Hong, Sungjoo Yoo, Onur Mutlu, and Kiyoun 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 (pptx) (pdf)] [Lightning Session Slides (pptx) (pdf)]*  
***Top Picks Honorable Mention by IEEE Micro.***  
***Selected to the ISCA-50 25-Year Retrospective Issue covering 1996-2020 in 2023 (Retrospective (pdf) Full Issue).***

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

Junwhan Ahn   Sungpack Hong<sup>§</sup>   Sungjoo Yoo   Onur Mutlu<sup>†</sup>   Kiyoun Choi  
junwhan@snu.ac.kr, sungpack.hong@oracle.com, sungjoo.yoo@gmail.com, onur@cmu.edu, kchoi@snu.ac.kr

Seoul National University

<sup>§</sup>Oracle Labs

<sup>†</sup>Carnegie Mellon University

# In-Storage Genomic Data 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)]

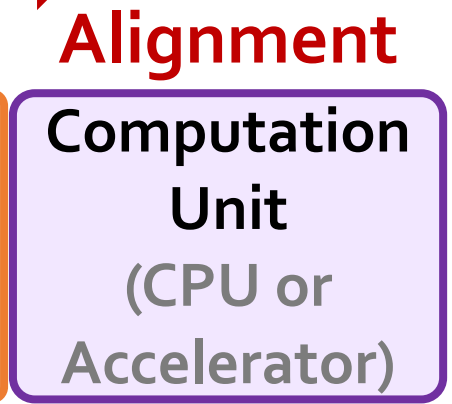
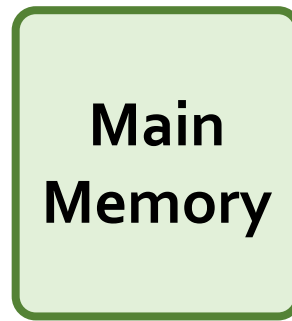
## GenStore: A High-Performance In-Storage Processing System for Genome Sequence Analysis

Nika Mansouri Ghiasi<sup>1</sup> Jisung Park<sup>1</sup> Harun Mustafa<sup>1</sup> Jeremie Kim<sup>1</sup> Ataberk Olgun<sup>1</sup>  
Arvid Gollwitzer<sup>1</sup> Damla Senol Cali<sup>2</sup> Can Firtina<sup>1</sup> Haiyu Mao<sup>1</sup> Nour Almadhoun Alserr<sup>1</sup>  
Rachata Ausavarungnirun<sup>3</sup> Nandita Vijaykumar<sup>4</sup> Mohammed Alser<sup>1</sup> Onur Mutlu<sup>1</sup>

<sup>1</sup>ETH Zürich <sup>2</sup>Bionano Genomics <sup>3</sup>KMUTNB <sup>4</sup>University of Toronto

# Genome Sequence Analysis

**Data Movement from Storage**

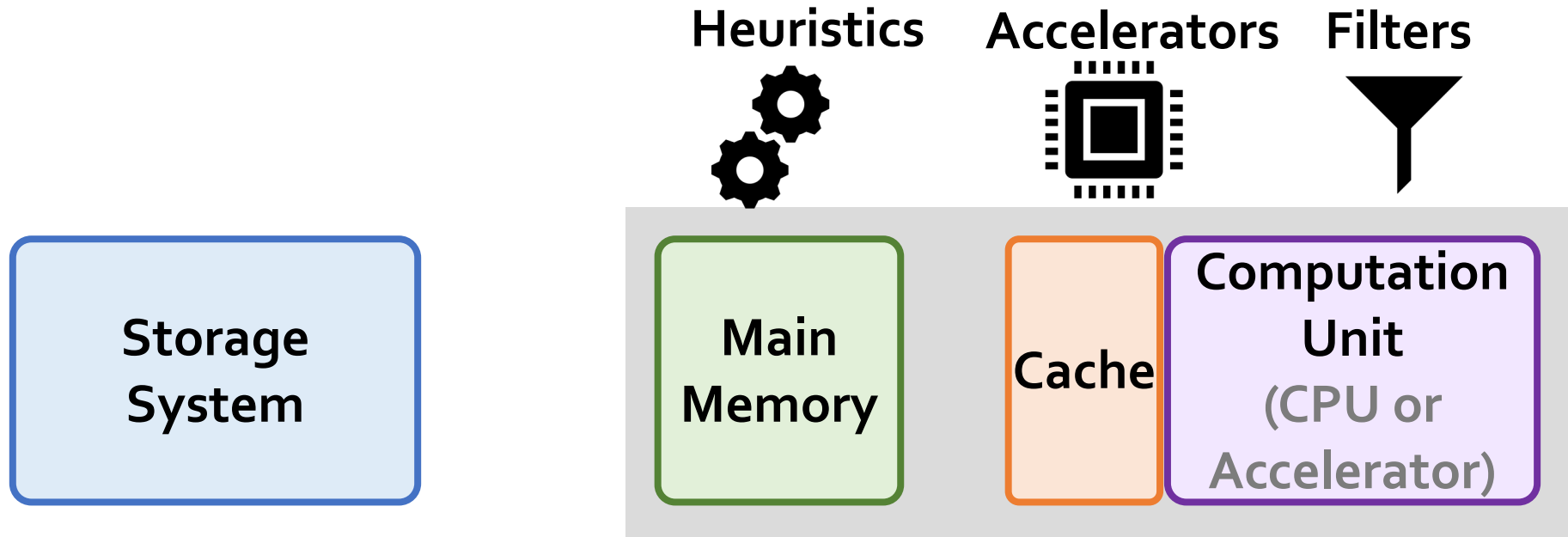


Computation overhead



Data movement overhead

# Compute-Centric Accelerators



Computation overhead



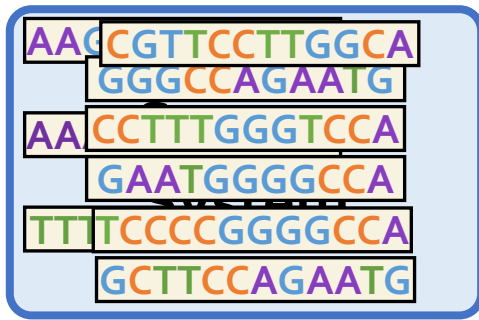
Data movement overhead



# Key Idea: In-Storage Filtering



*Filter reads that do **not** require alignment inside the storage system*



**Filtered Reads**

**Main  
Memory**

**Cache**

**Computation  
Unit**  
(CPU or  
Accelerator)

## **Exactly-matching** reads

Do not need expensive approximate string matching during alignment

## **Non-matching** reads

Do not have potential matching locations and can skip alignment

# GenStore



*Filter reads that do **not** require alignment  
inside the storage system*

GenStore-Enabled  
Storage  
System

Main  
Memory

Cache

Computation  
Unit  
(CPU or  
Accelerator)



Computation overhead



Data movement overhead

GenStore provides significant speedup (1.4x - 33.6x) and  
energy reduction (3.9x - 29.2x) at low cost

# In-Storage Genomic Data 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 Ghiasi<sup>1</sup> Jisung Park<sup>1</sup> Harun Mustafa<sup>1</sup> Jeremie Kim<sup>1</sup> Ataberk Olgun<sup>1</sup>  
Arvid Gollwitzer<sup>1</sup> Damla Senol Cali<sup>2</sup> Can Firtina<sup>1</sup> Haiyu Mao<sup>1</sup> Nour Almadhoun Alserr<sup>1</sup>  
Rachata Ausavarungnirun<sup>3</sup> Nandita Vijaykumar<sup>4</sup> Mohammed Alser<sup>1</sup> Onur Mutlu<sup>1</sup>

<sup>1</sup>ETH Zürich <sup>2</sup>Bionano Genomics <sup>3</sup>KMUTNB <sup>4</sup>University of Toronto

# Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks

**Amirali Boroumand**

Saugata Ghose, Youngsok Kim, Rachata Ausavarungnirun,  
Eric Shiu, Rahul Thakur, Daehyun Kim, Aki Kuusela,  
Allan Knies, Parthasarathy Ranganathan, Onur Mutlu

**SAFARI**

**Carnegie Mellon**

**Google**



SEOUL  
NATIONAL  
UNIVERSITY

**ETH** zürich

# Consumer Devices



**Consumer devices are everywhere!**

**Energy consumption is  
a first-class concern in consumer devices**



# Popular Consumer Workloads



**Chrome**

Google's web browser



**TensorFlow Mobile**

Google's machine learning  
framework

**VP9**



**Video Playback**

Google's **video codec**

**VP9**

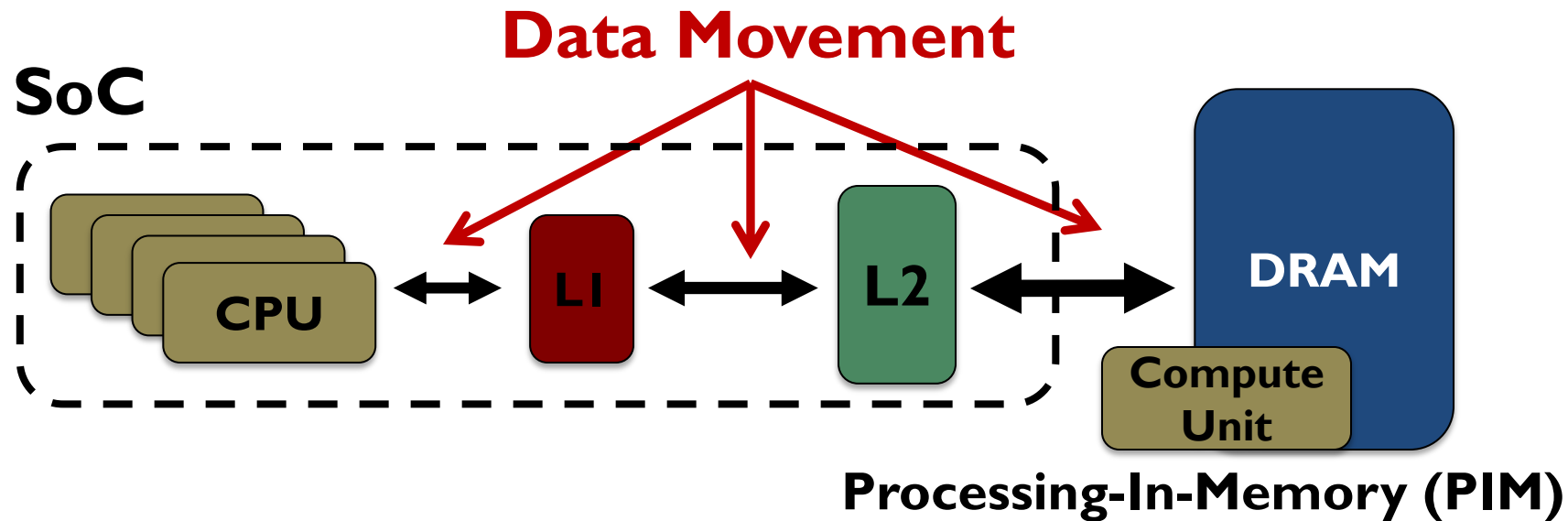


**Video Capture**

Google's **video codec**

# Energy Cost of Data Movement

**1<sup>st</sup> key observation:** **62.7%** of the total system energy is spent on **data movement**



**Potential solution:** move computation **close to data**

**Challenge:** limited area and energy budget



# Using PIM to Reduce Data Movement

**2<sup>nd</sup> key observation:** a significant fraction of the **data movement** often comes from **simple functions**

We can design lightweight logic to implement these simple functions in **memory**

Small embedded  
low-power core



Small fixed-function  
accelerators



Offloading to PIM logic reduces energy and improves performance, on average, by 2.3X and 2.2X

# Workload Analysis



**Chrome**

Google's web browser



**TensorFlow Mobile**

Google's machine learning  
framework



**Video Playback**

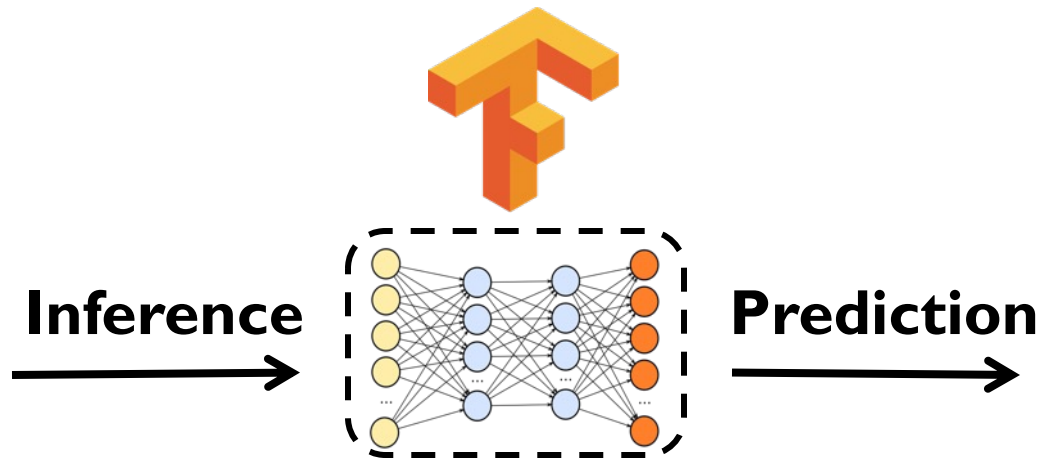
Google's **video codec**



**Video Capture**

Google's **video codec**

# TensorFlow Mobile

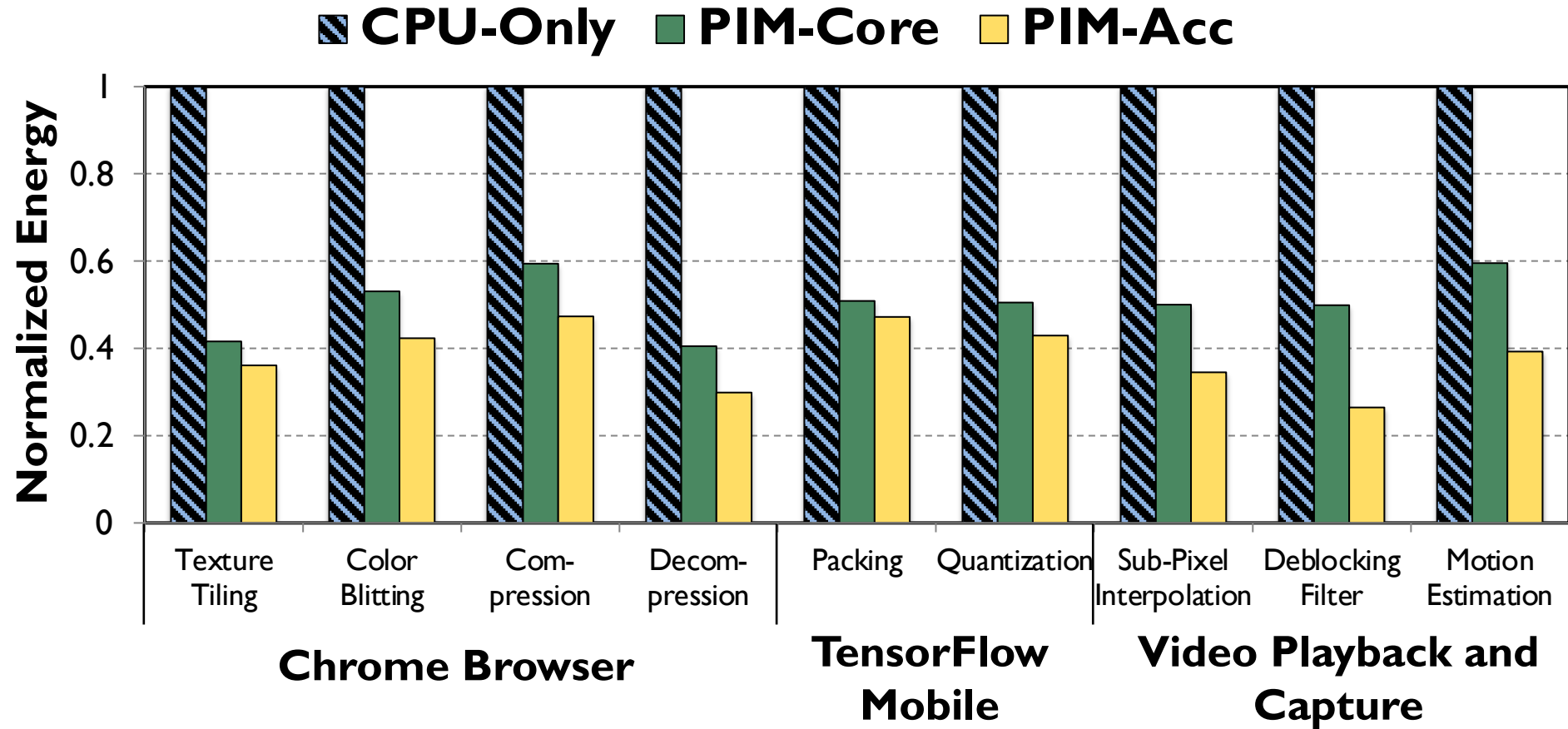


**57.3%** of the inference energy is spent on data movement



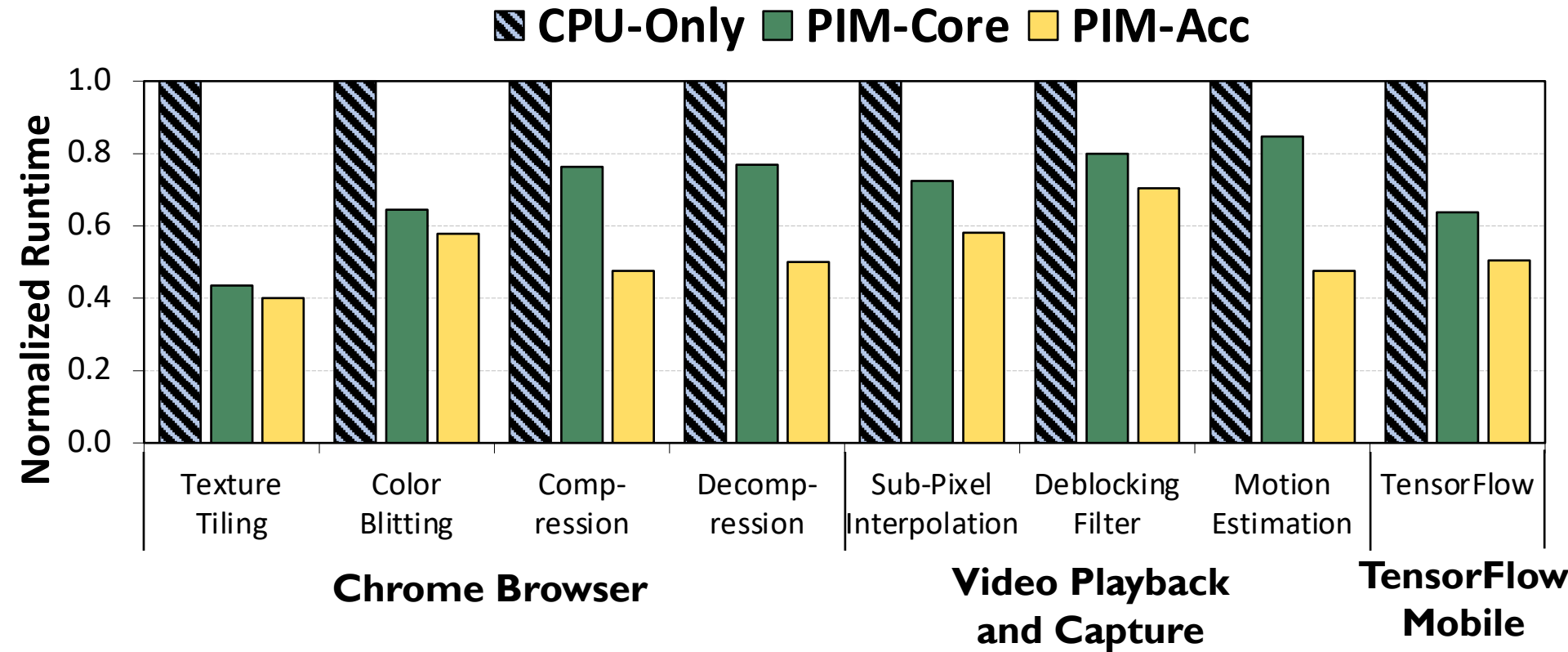
**54.4%** of the **data movement** energy comes from packing/unpacking and quantization

# Normalized Energy



**PIM core and PIM accelerator reduce**  
**energy consumption on average by 49.1% and 55.4%**

# Normalized Runtime



Offloading these kernels to **PIM core** and **PIM accelerator** reduces **program runtime** on average by **44.6%** and **54.2%**

# More on PIM for 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.*

[[Slides \(pptx\) \(pdf\)](#)] [[Lightning Session Slides \(pptx\) \(pdf\)](#)] [[Poster \(pptx\) \(pdf\)](#)]

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

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

## **Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks**

Amirali Boroumand<sup>1</sup>

Saugata Ghose<sup>1</sup>

Youngsok Kim<sup>2</sup>

Rachata Ausavarungnirun<sup>1</sup>

Eric Shiu<sup>3</sup>

Rahul Thakur<sup>3</sup>

Daehyun Kim<sup>4,3</sup>

Aki Kuusela<sup>3</sup>

Allan Knies<sup>3</sup>

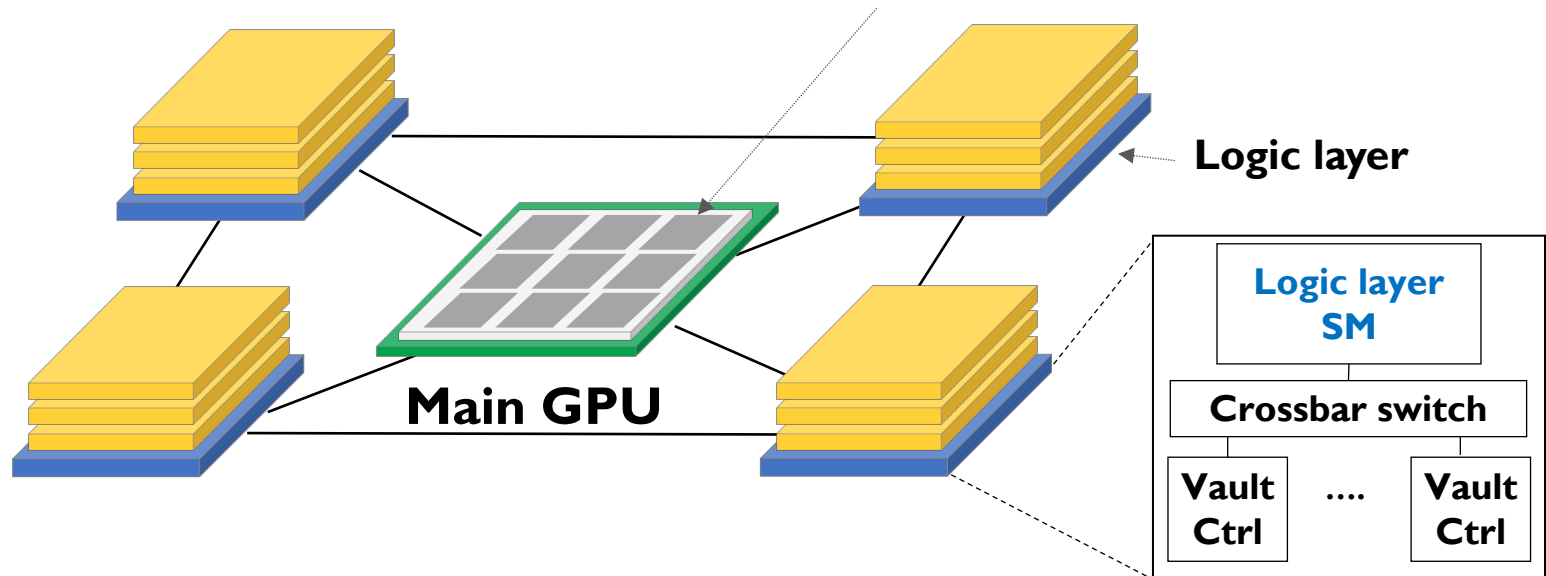
Parthasarathy Ranganathan<sup>3</sup>

Onur Mutlu<sup>5,1</sup>

# Truly Distributed GPU Processing with PIM

**3D-stacked memory  
(memory stack)**

**SM (Streaming Multiprocessor)**



```
__global__
void applyScaleFactorsKernel( uint8_T * const out,
                             uint8_T const * const in, const double *factor,
                             size_t const numRows, size_t const numCols )
{
    // Work out which pixel we are working on.
    const int rowIdx = blockIdx.x * blockDim.x + threadIdx.x;
    const int colIdx = blockIdx.y;
    const int sliceIdx = threadIdx.z;

    // Check this thread isn't off the image
    if( rowIdx >= numRows ) return;

    // Compute the index of my element
    size_t linearIdx = rowIdx + colIdx*numRows +
                      sliceIdx*numRows*numCols;
```



# 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<sup>‡</sup> Eiman Ebrahimi<sup>†</sup> Gwangsun Kim\* Niladrish Chatterjee<sup>†</sup> Mike O'Connor<sup>†</sup>  
Nandita Vijaykumar<sup>‡</sup> Onur Mutlu<sup>§‡</sup> Stephen W. Keckler<sup>†</sup>

<sup>‡</sup>Carnegie Mellon University <sup>†</sup>NVIDIA <sup>\*</sup>KAIST <sup>§</sup>ETH Zürich

# Accelerating GPU Execution with PIM (II)

---

- Ashutosh Pattnaik, Xulong Tang, Adwait Jog, Onur Kayiran, Asit K. Mishra, Mahmut T. Kandemir, Onur Mutlu, and Chita R. Das,  
**"Scheduling Techniques for GPU Architectures with Processing-In-Memory Capabilities"**  
*Proceedings of the 25th International Conference on Parallel Architectures and Compilation Techniques (PACT)*, Haifa, Israel, September 2016.

## Scheduling Techniques for GPU Architectures with Processing-In-Memory Capabilities

Ashutosh Pattnaik<sup>1</sup>    Xulong Tang<sup>1</sup>    Adwait Jog<sup>2</sup>    Onur Kayiran<sup>3</sup>  
Asit K. Mishra<sup>4</sup>    Mahmut T. Kandemir<sup>1</sup>    Onur Mutlu<sup>5,6</sup>    Chita R. Das<sup>1</sup>

<sup>1</sup>Pennsylvania State University    <sup>2</sup>College of William and Mary  
<sup>3</sup>Advanced Micro Devices, Inc.    <sup>4</sup>Intel Labs    <sup>5</sup>ETH Zürich    <sup>6</sup>Carnegie Mellon University

# 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<sup>†</sup> Samira Khan<sup>‡</sup> Nandita Vijaykumar<sup>†</sup>  
Kevin K. Chang<sup>†</sup> Amirali Boroumand<sup>†</sup> Saugata Ghose<sup>†</sup> Onur Mutlu<sup>§†</sup>  
<sup>†</sup>*Carnegie Mellon University*   <sup>‡</sup>*University of Virginia*   <sup>§</sup>*ETH Zürich*

# Accelerating Dependent Cache Misses

---

- Milad Hashemi, Khubaib, Eiman Ebrahimi, Onur Mutlu, and Yale N. Patt, **"Accelerating Dependent Cache Misses with an Enhanced Memory Controller"**

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

*[Slides (pptx) (pdf)]*

*[Lightning Session Slides (pptx) (pdf)]*

## Accelerating Dependent Cache Misses with an Enhanced Memory Controller

Milad Hashemi\*, Khubaib<sup>†</sup>, Eiman Ebrahimi<sup>‡</sup>, Onur Mutlu<sup>§</sup>, Yale N. Patt\*

*\*The University of Texas at Austin    <sup>†</sup>Apple    <sup>‡</sup>NVIDIA    <sup>§</sup>ETH Zürich & Carnegie Mellon University*

# Accelerating Runahead Execution

---

- Milad Hashemi, Onur Mutlu, and Yale N. Patt,  
**"Continuous Runahead: Transparent Hardware Acceleration for Memory Intensive Workloads"**  
*Proceedings of the 49th International Symposium on Microarchitecture (MICRO), Taipei, Taiwan, October 2016.*  
[\[Slides \(pptx\) \(pdf\)\]](#) [\[Lightning Session Slides \(pdf\)\]](#) [\[Poster \(pptx\) \(pdf\)\]](#)  
***Best paper session.***

## Continuous Runahead: Transparent Hardware Acceleration for Memory Intensive Workloads

Milad Hashemi\*, Onur Mutlu<sup>§</sup>, Yale N. Patt\*

\**The University of Texas at Austin*    <sup>§</sup>*ETH Zürich*

# Accelerating Climate Modeling

---

- 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<sup>a,b,c</sup>    Dionysios Diamantopoulos<sup>c</sup>    Christoph Hagleitner<sup>c</sup>    Juan Gómez-Luna<sup>b</sup>  
Sander Stuijk<sup>a</sup>    Onur Mutlu<sup>b</sup>    Henk Corporaal<sup>a</sup>  
<sup>a</sup>Eindhoven University of Technology    <sup>b</sup>ETH Zürich    <sup>c</sup>IBM Research Europe, Zurich

# Accelerating Approximate String Matching

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

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

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



# Accelerating Sequence-to-Graph Mapping

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

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

Damla Senol Cali<sup>1</sup> Konstantinos Kanellopoulos<sup>2</sup> Joël Lindegger<sup>2</sup> Zülal Bingöl<sup>3</sup>  
Gurpreet S. Kalsi<sup>4</sup> Ziyi Zuo<sup>5</sup> Can Firtina<sup>2</sup> Meryem Banu Cavlak<sup>2</sup> Jeremie Kim<sup>2</sup>  
Nika Mansouri Ghiasi<sup>2</sup> Gagandeep Singh<sup>2</sup> Juan Gómez-Luna<sup>2</sup> Nour Almadhoun Alserr<sup>2</sup>  
Mohammed Alser<sup>2</sup> Sreenivas Subramoney<sup>4</sup> Can Alkan<sup>3</sup> Saugata Ghose<sup>6</sup> Onur Mutlu<sup>2</sup>

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

# Accelerating Basecalling + Read Mapping

---

- Haiyu Mao, Mohammed Alser, Mohammad Sadrosadati, Can Firtina, Akanksha Baranwal, Damla Senol Cali, Aditya Manglik, Nour Almadhoun Alserr, and Onur Mutlu,  
**"GenPIP: In-Memory Acceleration of Genome Analysis via Tight Integration of Basecalling and Read Mapping"**  
*Proceedings of the 55th International Symposium on Microarchitecture (MICRO)*,  
Chicago, IL, USA, October 2022.  
[[Slides \(pptx\)](#)] [[pdf](#)]  
[[Longer Lecture Slides \(pptx\)](#)] [[pdf](#)]  
[[Lecture Video](#) (25 minutes)]  
[[arXiv version](#)]

## **GenPIP: In-Memory Acceleration of Genome Analysis via Tight Integration of Basecalling and Read Mapping**

Haiyu Mao<sup>1</sup> Mohammed Alser<sup>1</sup> Mohammad Sadrosadati<sup>1</sup> Can Firtina<sup>1</sup> Akanksha Baranwal<sup>1</sup>  
Damla Senol Cali<sup>2</sup> Aditya Manglik<sup>1</sup> Nour Almadhoun Alserr<sup>1</sup> Onur Mutlu<sup>1</sup>  
<sup>1</sup>*ETH Zürich*      <sup>2</sup>*Bionano Genomics*

# Accelerating Time Series Analysis

---

- Ivan Fernandez, Ricardo Quisiant, 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<sup>§</sup>

Ricardo Quisiant<sup>§</sup>

Christina Giannoula<sup>†</sup>

Mohammed Alser<sup>‡</sup>

Juan Gómez-Luna<sup>‡</sup>

Eladio Gutiérrez<sup>§</sup>

Oscar Plata<sup>§</sup>

Onur Mutlu<sup>‡</sup>

<sup>§</sup>*University of Malaga*

<sup>†</sup>*National Technical University of Athens*

<sup>‡</sup>*ETH Zürich*

# Accelerating Graph Pattern Mining

- Maciej Besta, Raghavendra Kanakagiri, Grzegorz Kwasniewski, Rachata Ausavarungnirun, Jakub Beránek, Konstantinos Kanellopoulos, Kacper Janda, Zur Vonarburg-Shmaria, Lukas Gianinazzi, Ioana Stefan, Juan Gómez-Luna, Marcin Copik, Lukas Kapp-Schwoerer, Salvatore Di Girolamo, Nils Blach, Marek Konieczny, Onur Mutlu, and Torsten Hoefler,

## **"SISA: Set-Centric Instruction Set Architecture for Graph Mining on Processing-in-Memory Systems"**

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

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

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

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

[[Full arXiv version](#)]

## **SISA: Set-Centric Instruction Set Architecture for Graph Mining on Processing-in-Memory Systems**

Maciej Besta<sup>1</sup>, Raghavendra Kanakagiri<sup>2</sup>, Grzegorz Kwasniewski<sup>1</sup>, Rachata Ausavarungnirun<sup>3</sup>, Jakub Beránek<sup>4</sup>, Konstantinos Kanellopoulos<sup>1</sup>, Kacper Janda<sup>5</sup>, Zur Vonarburg-Shmaria<sup>1</sup>, Lukas Gianinazzi<sup>1</sup>, Ioana Stefan<sup>1</sup>, Juan Gómez-Luna<sup>1</sup>, Marcin Copik<sup>1</sup>, Lukas Kapp-Schwoerer<sup>1</sup>, Salvatore Di Girolamo<sup>1</sup>, Nils Blach<sup>1</sup>, Marek Konieczny<sup>5</sup>, Onur Mutlu<sup>1</sup>, Torsten Hoefler<sup>1</sup>

<sup>1</sup>ETH Zurich, Switzerland  
Thailand

<sup>2</sup>IIT Tirupati, India

<sup>3</sup>King Mongkut's University of Technology North Bangkok,  
<sup>4</sup>Technical University of Ostrava, Czech Republic

<sup>5</sup>AGH-UST, Poland

# Accelerating HTAP Database Systems

---

- Amirali Boroumand, Saugata Ghose, Geraldo F. Oliveira, and Onur Mutlu,  
**"Polynesia: Enabling High-Performance and Energy-Efficient Hybrid Transactional/Analytical Databases with Hardware/Software Co-Design"**  
*Proceedings of the 38th International Conference on Data Engineering (ICDE)*,  
Virtual, May 2022.  
[[arXiv version](#)]  
[[Slides \(pptx\)](#) ([pdf](#))]  
[[Short Talk Slides \(pptx\)](#) ([pdf](#))]

## Polynesia: Enabling High-Performance and Energy-Efficient Hybrid Transactional/Analytical Databases with Hardware/Software Co-Design

Amirali Boroumand<sup>†</sup>      Saugata Ghose<sup>◇</sup>      Geraldo F. Oliveira<sup>‡</sup>      Onur Mutlu<sup>‡</sup>  
<sup>†</sup>*Google*      <sup>◇</sup>*Univ. of Illinois Urbana-Champaign*      <sup>‡</sup>*ETH Zürich*

# Accelerating Neural Network Inference

---

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

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

Amirali Boroumand<sup>†◇</sup>

Geraldo F. Oliveira<sup>\*</sup>

Saugata Ghose<sup>‡</sup>

Xiaoyu Ma<sup>§</sup>

Berkin Akin<sup>§</sup>

Eric Shiu<sup>§</sup>

Ravi Narayanaswami<sup>§</sup>

Onur Mutlu<sup>\*†</sup>

<sup>†</sup>*Carnegie Mellon Univ.*

<sup>◇</sup>*Stanford Univ.*

<sup>‡</sup>*Univ. of Illinois Urbana-Champaign*

<sup>§</sup>*Google*

<sup>\*</sup>*ETH Zürich*

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

**Amirali Boroumand**

**Saugata Ghose**

**Berkin Akin**

**Ravi Narayanaswami**

**Geraldo F. Oliveira**

**Xiaoyu Ma**

**Eric Shiu**

**Onur Mutlu**

**PACT 2021**

**SAFARI**

**Carnegie Mellon**



UNIVERSITY OF  
**ILLINOIS**  
URBANA-CHAMPAIGN



**ETH** zürich



# Executive Summary

**Context:** We extensively analyze a state-of-the-art edge ML accelerator (Google Edge TPU) using 24 Google edge models

- Wide range of models (CNNs, LSTMs, Transducers, RCNNs)

**Problem:** The Edge TPU accelerator suffers from **three challenges:**

- It operates **significantly below** its peak throughput
- It operates **significantly below** its theoretical energy efficiency
- It **inefficiently** handles memory accesses

**Key Insight:** These shortcomings arise from **the monolithic design** of the Edge TPU accelerator

- The Edge TPU accelerator design does not account for **layer heterogeneity**

**Key Mechanism:** A new framework called **Mensa**

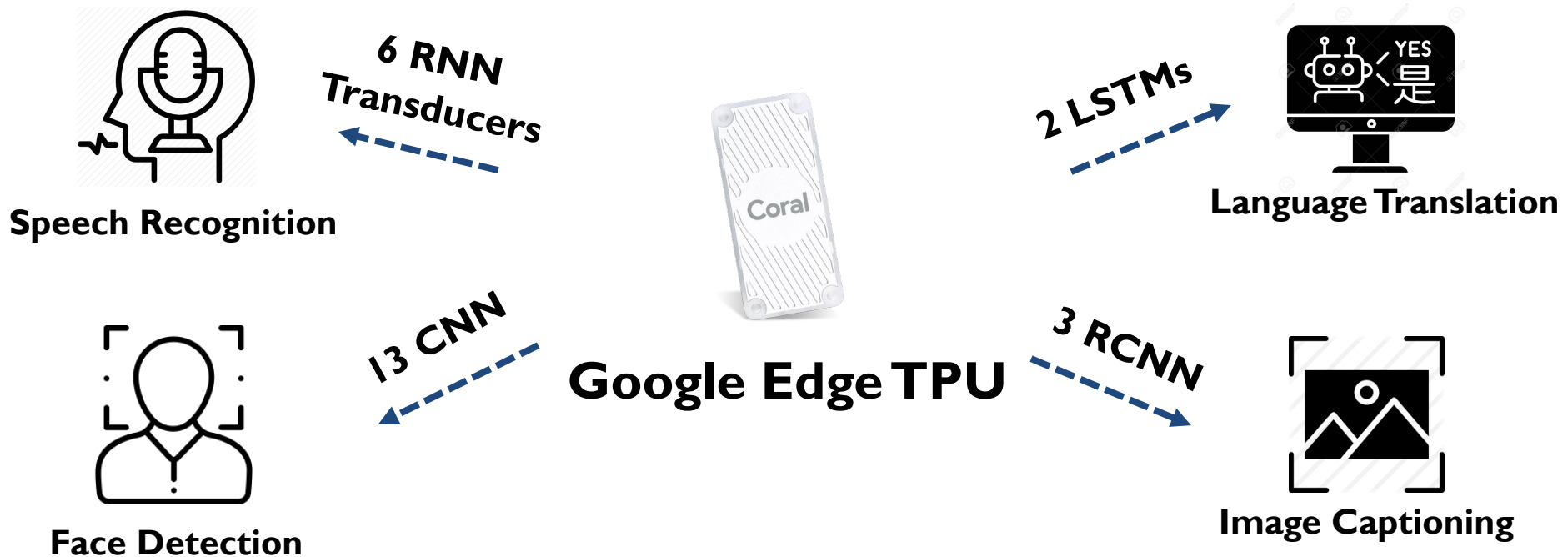
- Mensa consists of heterogeneous accelerators whose dataflow and hardware are specialized for specific families of layers

**Key Results:** We design a version of Mensa for Google edge ML models

- Mensa improves performance and energy by **3.0X** and **3.1X**
- Mensa reduces cost and improves area efficiency

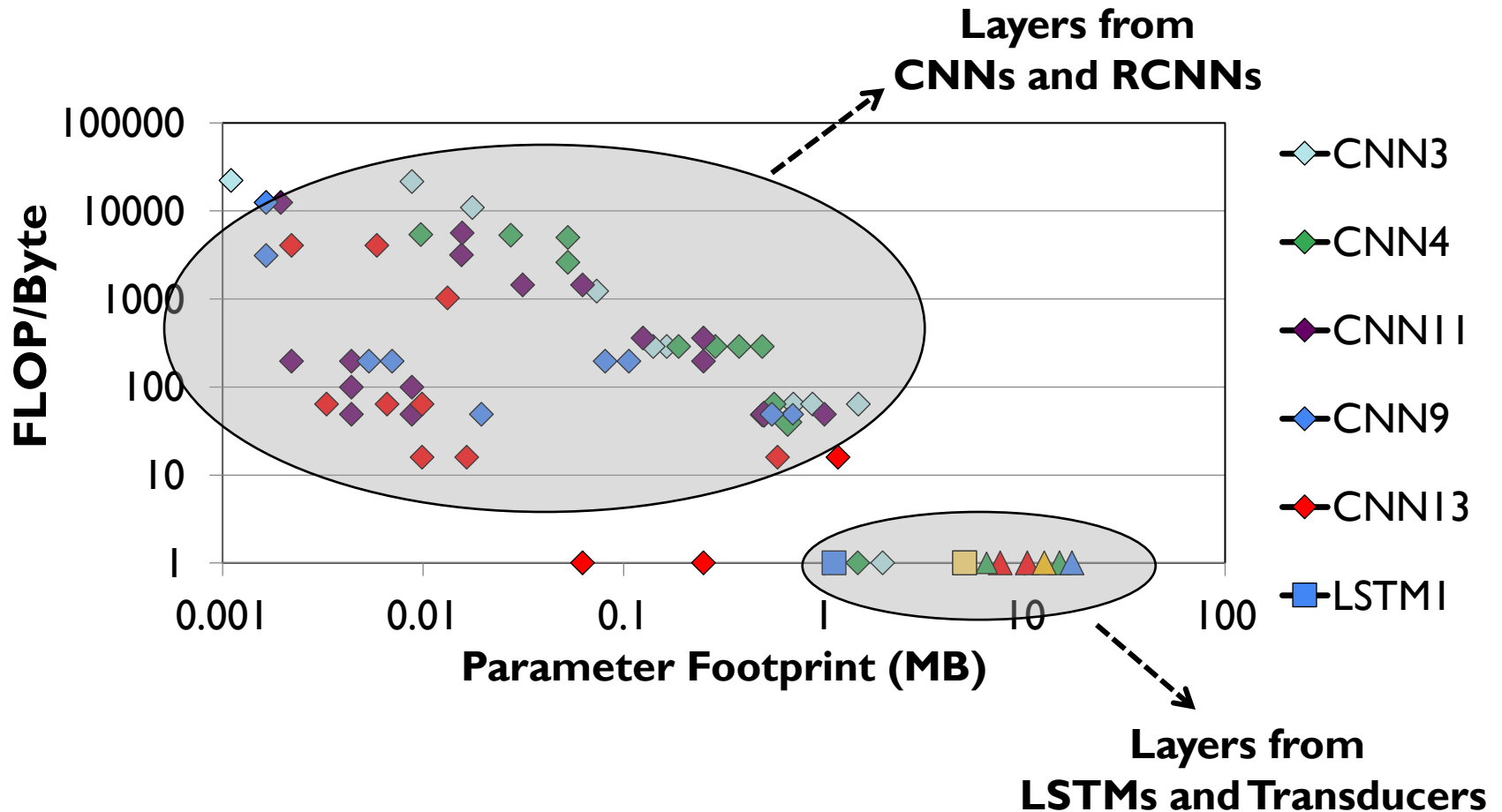
# Google Edge Neural Network Models

We analyze inference execution using 24 edge NN models



# Diversity Across the Models

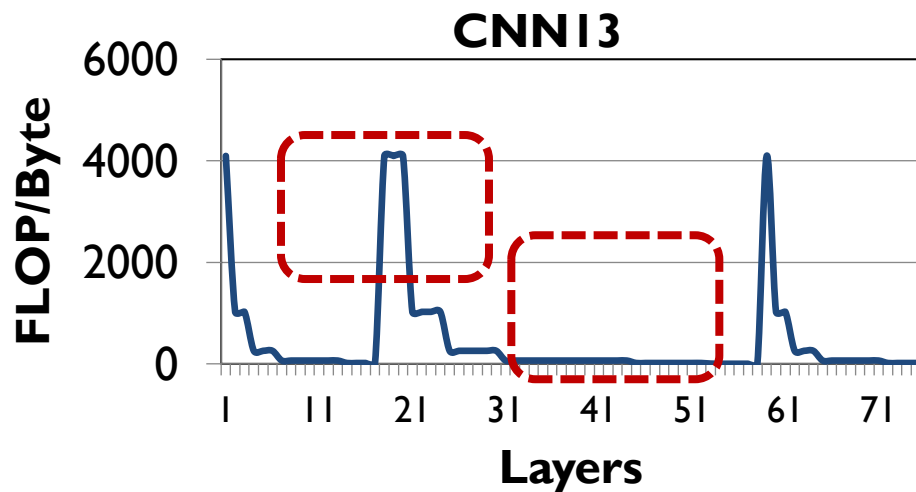
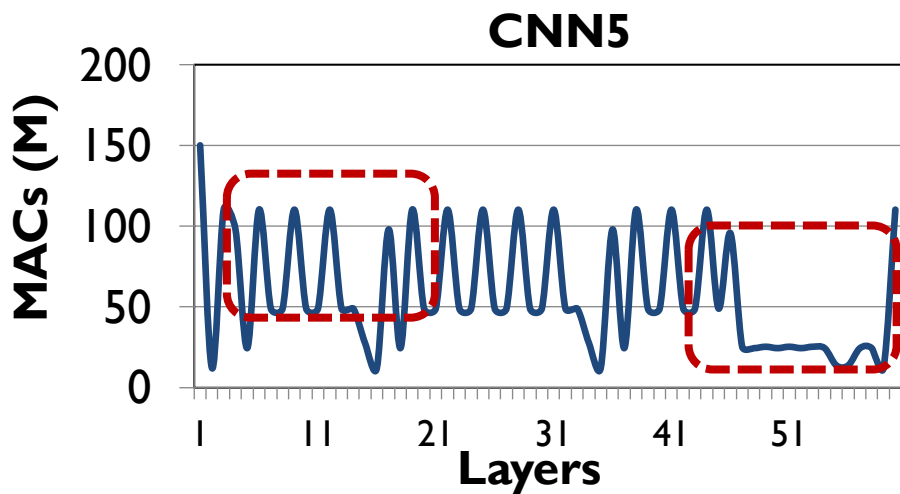
**Insight 1:** there is **significant variation** in terms of layer characteristics **across the models**



# Diversity Within the Models

**Insight 2:** even **within** each model, layers exhibit **significant variation** in terms of layer characteristics

For example, our analysis of edge **CNN** models shows:



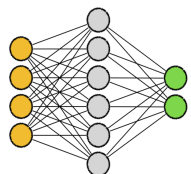
Variation in **MAC intensity**: up to **200x** across layers

Variation in **FLOP/Byte**: up to **244x** across layers

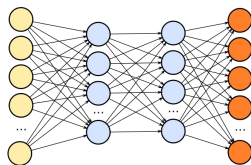
# Mensa High-Level Overview

## Edge TPU Accelerator

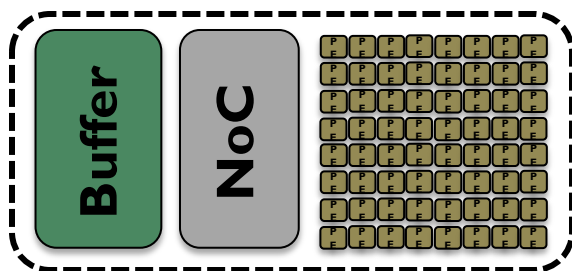
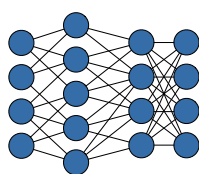
Model A



Model B



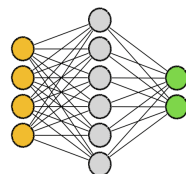
Model C



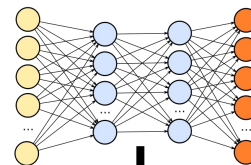
Monolithic Accelerator

## Mensa

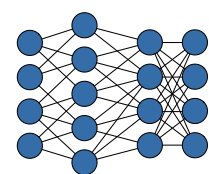
Model A



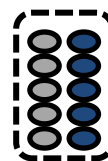
Model B



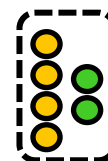
Model C



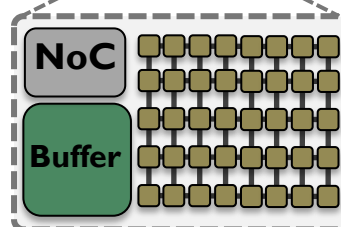
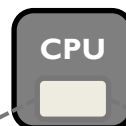
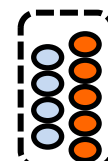
Family 1



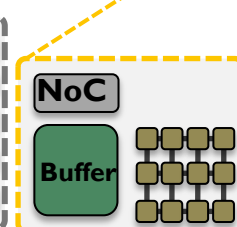
Family 2



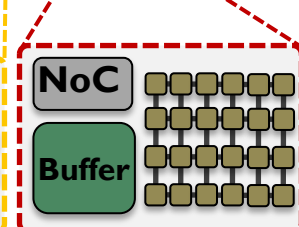
Family 3



Acc. 1



Acc. 2

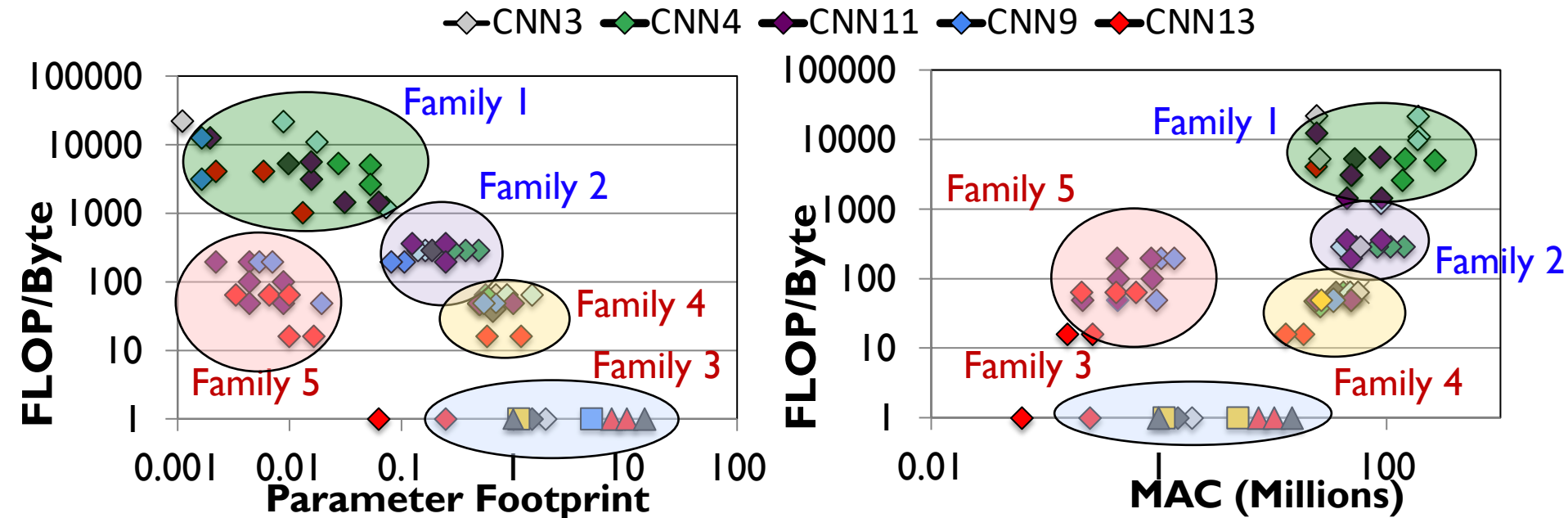


Acc. 3

Heterogeneous Accelerators

# Identifying Layer Families

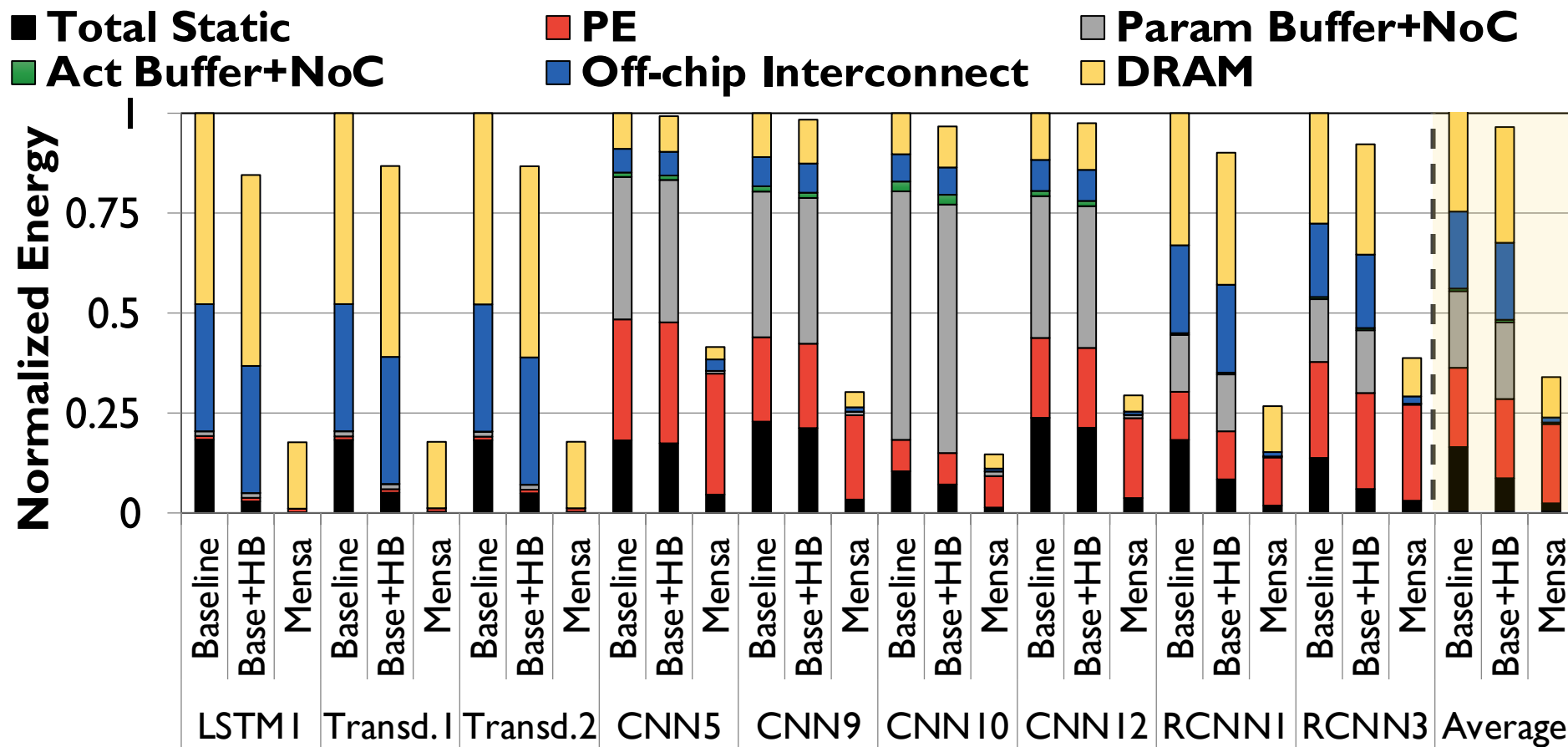
**Key observation: the majority of layers group into a small number of layer families**



**Families 1 & 2: low parameter footprint, high data reuse and **MAC** intensity**  
→ compute-centric layers

**Families 3, 4 & 5: high parameter footprint, low data reuse and **MAC** intensity**  
→ data-centric layers

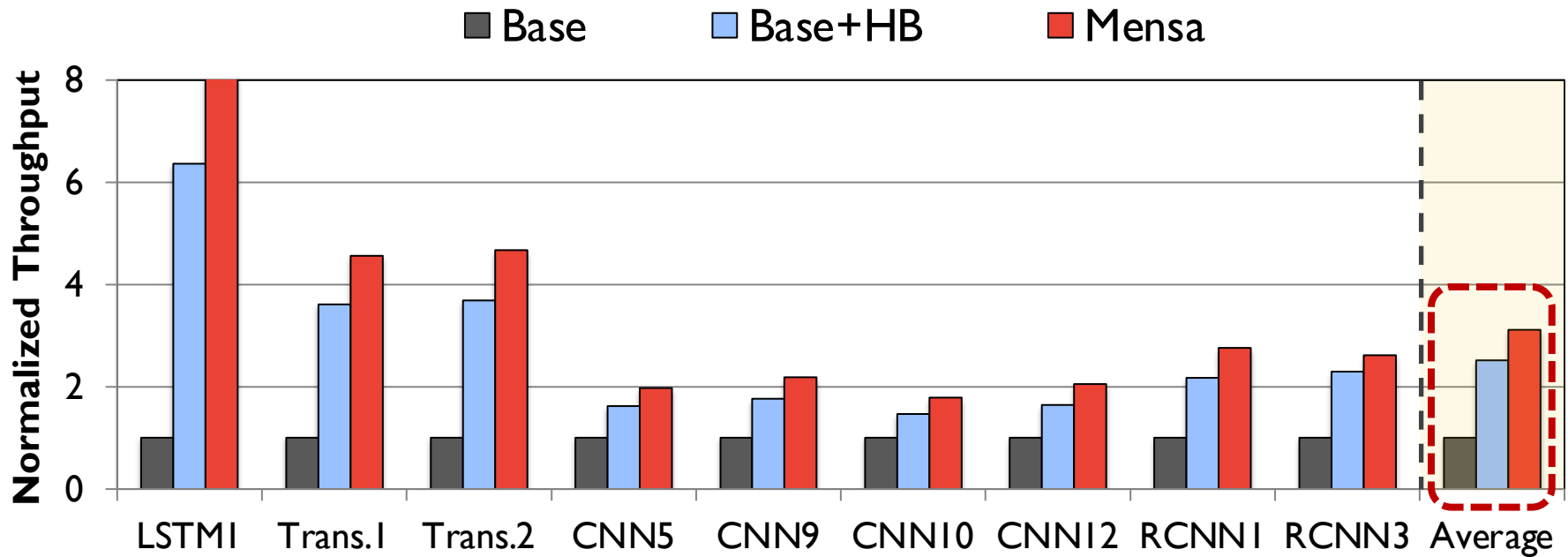
# Mensa: Energy Reduction



**Mensa-G reduces energy consumption by 3.0X**  
compared to the baseline EdgeTPU



# Mensa: Throughput Improvement



**Mensa-G improves inference throughput by 3.1X**  
compared to the baseline Edge TPU

# Mensa: Highly-Efficient ML Inference

---

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

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

Amirali Boroumand<sup>†◇</sup>

Geraldo F. Oliveira<sup>\*</sup>

Saugata Ghose<sup>‡</sup>

Xiaoyu Ma<sup>§</sup>

Berkin Akin<sup>§</sup>

Eric Shiu<sup>§</sup>

Ravi Narayanaswami<sup>§</sup>

Onur Mutlu<sup>\*†</sup>

<sup>†</sup>Carnegie Mellon Univ.

<sup>◇</sup>Stanford Univ.

<sup>‡</sup>Univ. of Illinois Urbana-Champaign

<sup>§</sup>Google

<sup>\*</sup>ETH Zürich

# Accelerating Data-Intensive Workloads

---

- Junwhan Ahn, Sungjoo Yoo, Onur Mutlu, and Kiyoun 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<sup>†</sup>   Kiyoun Choi

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

Seoul National University

<sup>†</sup>Carnegie Mellon University

# 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<sup>◇</sup> Mohammed Alser<sup>◇</sup> Damla Senol Cali<sup>✕</sup>

Dionysios Diamantopoulos<sup>▽</sup> Juan Gómez-Luna<sup>◇</sup>

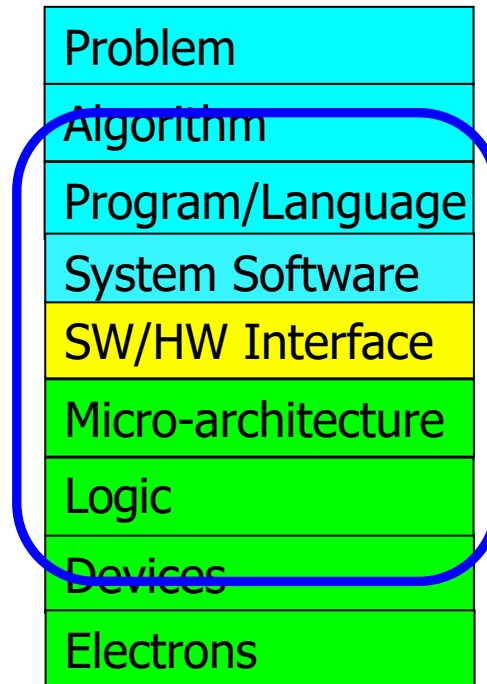
Henk Corporaal<sup>★</sup> Onur Mutlu<sup>◇✕</sup>

<sup>◇</sup>*ETH Zürich*    <sup>✕</sup>*Carnegie Mellon University*

<sup>★</sup>*Eindhoven University of Technology*    <sup>▽</sup>*IBM Research Europe*

# We Need to Revisit the Entire Stack

---



**We can get there step by step**

# PIM Review and Open Problems

---

## A Modern Primer on Processing in Memory

Onur Mutlu<sup>a,b</sup>, Saugata Ghose<sup>b,c</sup>, Juan Gómez-Luna<sup>a</sup>, Rachata Ausavarungnirun<sup>d</sup>

*SAFARI Research Group*

<sup>a</sup>*ETH Zürich*

<sup>b</sup>*Carnegie Mellon University*

<sup>c</sup>*University of Illinois at Urbana-Champaign*

<sup>d</sup>*King Mongkut's University of Technology North Bangkok*

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

**"A Modern Primer on Processing in Memory"**

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

# PIM Review and Open Problems (II)

---

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

Saugata Ghose<sup>†</sup>   Amirali Boroumand<sup>†</sup>   Jeremie S. Kim<sup>†§</sup>   Juan Gómez-Luna<sup>§</sup>   Onur Mutlu<sup>§†</sup>

<sup>†</sup>*Carnegie Mellon University*

<sup>§</sup>*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]



# Processing in Memory: Adoption Challenges

1. Processing **using** Memory
2. Processing **near** Memory

## How to Enable Adoption of Processing in Memory

# Potential Barriers to Adoption of PIM

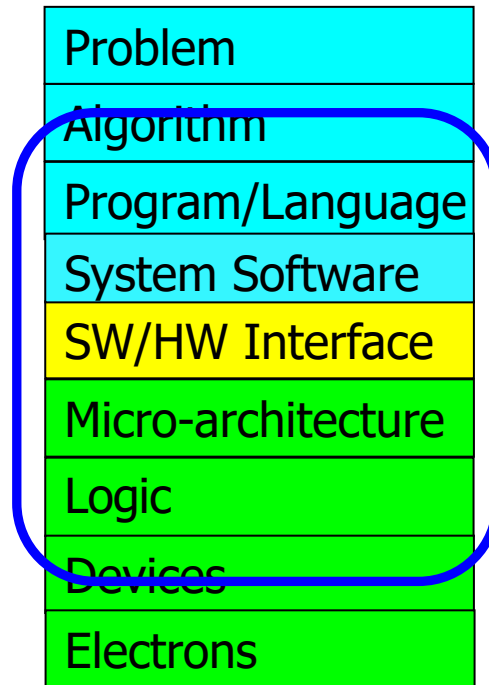
---

1. **Applications & software** for PIM
2. Ease of **programming** (interfaces and compiler/HW support)
3. **System** and **security** support: coherence, synchronization, virtual memory, isolation, communication interfaces, ...
4. **Runtime** and **compilation** systems for adaptive scheduling, data mapping, access/sharing control, ...
5. **Infrastructures** to assess benefits and feasibility

**All can be solved with change of mindset**

# We Need to Revisit the Entire Stack

---



**We can get there step by step**

# Adoption: How to Keep It Simple?

---

- Junwhan Ahn, Sungjoo Yoo, Onur Mutlu, and Kiyoungh 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<sup>†</sup>   Kiyoungh Choi

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

Seoul National University

<sup>†</sup>Carnegie Mellon University

# Adoption: How to Ease Programmability? (I)

---

- Geraldo F. Oliveira, Alain Kohli, David Novo, Juan Gómez-Luna, Onur Mutlu,  
**“DaPPA: A Data-Parallel Framework for Processing-in-Memory Architectures,”**  
in *PACT SRC Student Competition*, Vienna, Austria, October 2023.

## **DaPPA: A Data-Parallel Framework for Processing-in-Memory Architectures**

Geraldo F. Oliveira\*

Alain Kohli\*

David Novo‡

Juan Gómez-Luna\*

Onur Mutlu\*

\**ETH Zürich*

‡*LIRMM, Univ. Montpellier, CNRS*

# Adoption: How to Ease Programmability? (II)

---

- Jinfan Chen, Juan Gómez-Luna, Izzat El Hajj, YuXin Guo, and Onur Mutlu,  
**"SimplePIM: A Software Framework for Productive and Efficient Processing in Memory"**  
*Proceedings of the 32nd International Conference on Parallel Architectures and Compilation Techniques (PACT), Vienna, Austria, October 2023.*

## SimplePIM: A Software Framework for Productive and Efficient Processing-in-Memory

Jinfan Chen<sup>1</sup>   Juan Gómez-Luna<sup>1</sup>   Izzat El Hajj<sup>2</sup>   Yuxin Guo<sup>1</sup>   Onur Mutlu<sup>1</sup>  
<sup>1</sup>ETH Zürich   <sup>2</sup>American University of Beirut



# Adoption: How to Maintain Coherence? (I)

---

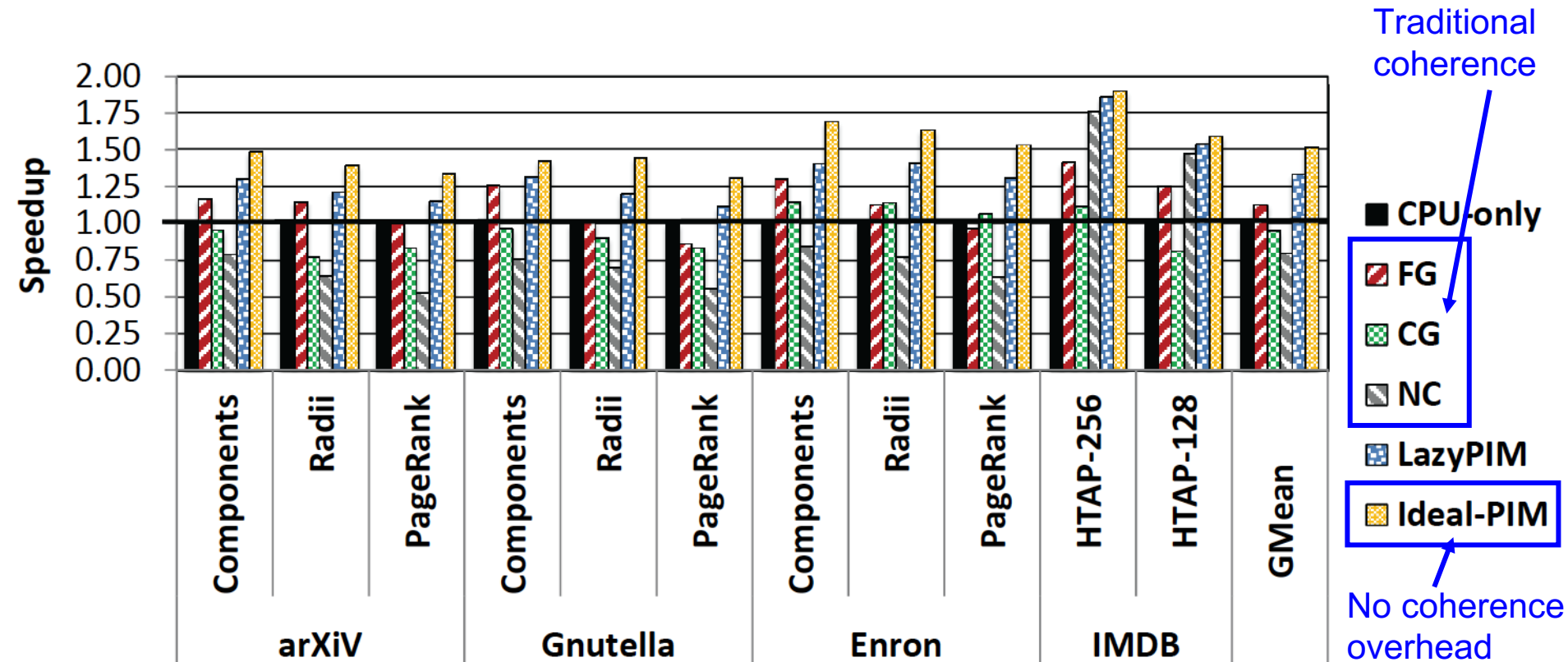
- Amirali Boroumand, Saugata Ghose, Minesh Patel, Hasan Hassan, Brandon Lucia, Kevin Hsieh, Krishna T. Malladi, Hongzhong Zheng, and Onur Mutlu,  
**"LazyPIM: An Efficient Cache Coherence Mechanism for Processing-in-Memory"**  
*IEEE Computer Architecture Letters* (**CAL**), June 2016.

## LazyPIM: An Efficient Cache Coherence Mechanism for Processing-in-Memory

Amirali Boroumand<sup>†</sup>, Saugata Ghose<sup>†</sup>, Minesh Patel<sup>†</sup>, Hasan Hassan<sup>†§</sup>, Brandon Lucia<sup>†</sup>,  
Kevin Hsieh<sup>†</sup>, Krishna T. Malladi<sup>\*</sup>, Hongzhong Zheng<sup>\*</sup>, and Onur Mutlu<sup>††</sup>

<sup>†</sup>Carnegie Mellon University   <sup>\*</sup>Samsung Semiconductor, Inc.   <sup>§</sup>TOBB ETÜ   <sup>‡</sup>ETH Zürich

# Challenge: Coherence for Hybrid CPU-PIM Apps



# Adoption: How to Maintain Coherence? (II)

---

- Amirali Boroumand, Saugata Ghose, Minesh Patel, Hasan Hassan, Brandon Lucia, Kevin Hsieh, Krishna T. Malladi, Hongzhong Zheng, and Onur Mutlu,  
**"CoNDA: Efficient Cache Coherence Support for Near-Data Accelerators"**

*Proceedings of the 46th International Symposium on Computer Architecture (ISCA), Phoenix, AZ, USA, June 2019.*

## CoNDA: Efficient Cache Coherence Support for Near-Data Accelerators

Amirali Boroumand<sup>†</sup>

Saugata Ghose<sup>†</sup>

Minesh Patel<sup>★</sup>

Hasan Hassan<sup>★</sup>

Brandon Lucia<sup>†</sup>

Rachata Ausavarungnirun<sup>†‡</sup>

Kevin Hsieh<sup>†</sup>

Nastaran Hajinazar<sup>◇†</sup>

Krishna T. Malladi<sup>§</sup>

Hongzhong Zheng<sup>§</sup>

Onur Mutlu<sup>★†</sup>

<sup>†</sup>Carnegie Mellon University

<sup>★</sup>ETH Zürich

<sup>‡</sup>KMUTNB

<sup>◇</sup>Simon Fraser University

<sup>§</sup>Samsung Semiconductor, Inc.

# Adoption: How to Support Synchronization?

---

- Christina Giannoula, Nandita Vijaykumar, Nikela Papadopoulou, Vasileios Karakostas, Ivan Fernandez, Juan Gómez-Luna, Lois Orosa, Nectarios Koziris, Georgios Goumas, 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.  
[[Slides \(pptx\)](#)] [[pdf](#)]  
[[Short Talk Slides \(pptx\)](#)] [[pdf](#)]  
[[Talk Video](#) (21 minutes)]  
[[Short Talk Video](#) (7 minutes)]

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

Christina Giannoula<sup>†‡</sup> Nandita Vijaykumar<sup>\*‡</sup> Nikela Papadopoulou<sup>†</sup> Vasileios Karakostas<sup>†</sup> Ivan Fernandez<sup>§‡</sup>  
Juan Gómez-Luna<sup>‡</sup> Lois Orosa<sup>‡</sup> Nectarios Koziris<sup>†</sup> Georgios Goumas<sup>†</sup> Onur Mutlu<sup>‡</sup>  
<sup>†</sup>*National Technical University of Athens*    <sup>‡</sup>*ETH Zürich*    <sup>\*</sup>*University of Toronto*    <sup>§</sup>*University of Malaga*

# Adoption: How to Support Virtual Memory?

---

- 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.

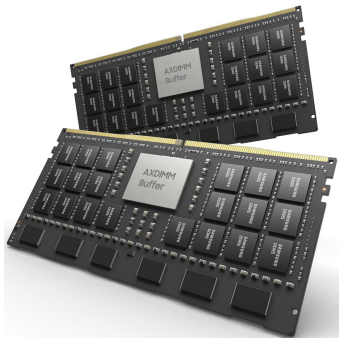
## Accelerating Pointer Chasing in 3D-Stacked Memory: Challenges, Mechanisms, Evaluation

Kevin Hsieh<sup>†</sup> Samira Khan<sup>‡</sup> Nandita Vijaykumar<sup>†</sup>  
Kevin K. Chang<sup>†</sup> Amirali Boroumand<sup>†</sup> Saugata Ghose<sup>†</sup> Onur Mutlu<sup>§†</sup>  
<sup>†</sup>*Carnegie Mellon University*   <sup>‡</sup>*University of Virginia*   <sup>§</sup>*ETH Zürich*

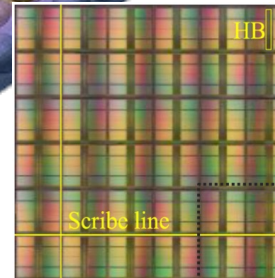
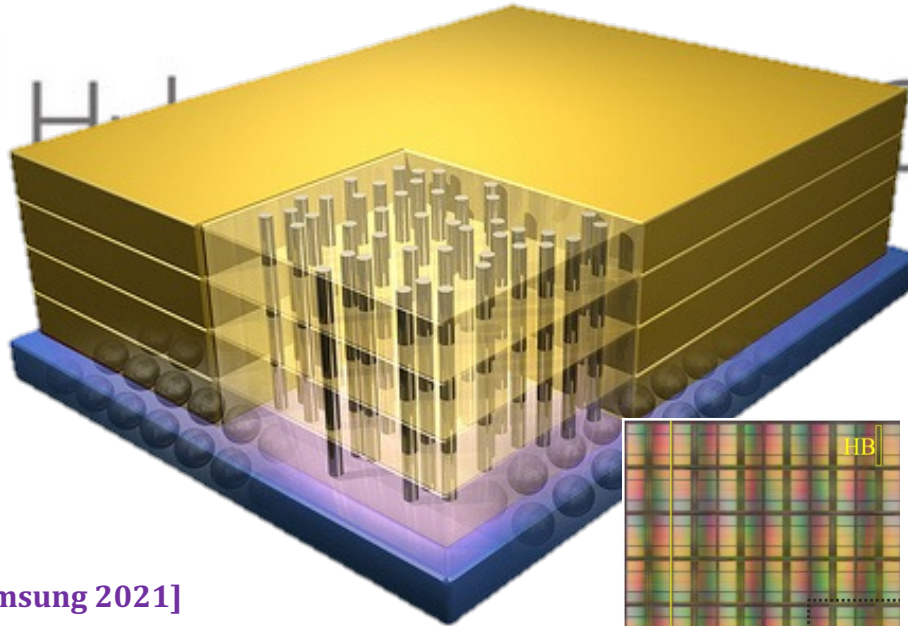
## Processing-in-Memory in the Real World



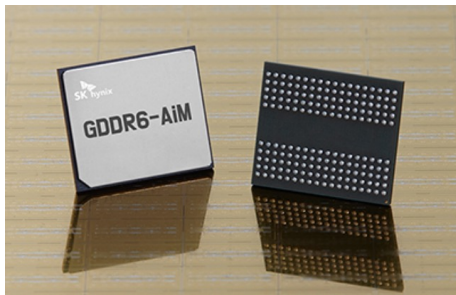
# Processing-in-Memory Landscape Today



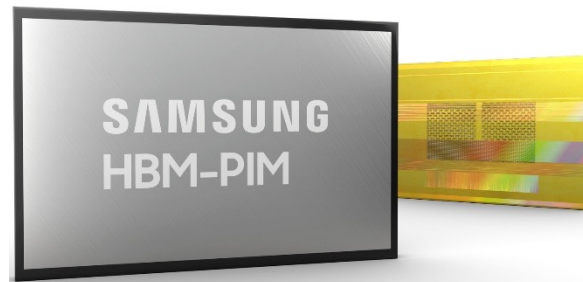
[Samsung 2021]



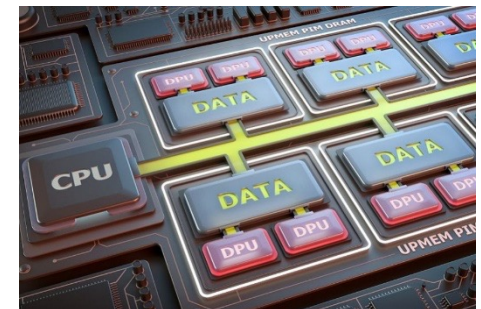
[Alibaba 2022]



[SK Hynix 2022]



[Samsung 2021]



[UPMEM 2019]

# Processing-in-Memory Landscape Today

IEEE COMPUTER ARCHITECTURE LETTERS, VOL. 22, NO. 1, JANUARY-JUNE

## Computational CXL-Memory Solution for Accelerating Memory-Intensive Applications

Joonseop Sim<sup>ID</sup>, Soohong Ahn<sup>ID</sup>, Taeyoung Ahn<sup>ID</sup>,  
Seungyong Lee<sup>ID</sup>, Myunghyun Rhee, Jooyoung Kim<sup>ID</sup>,  
Kwangsik Shin, Donguk Moon<sup>ID</sup>,  
Euseok Kim, and Kyoung Park<sup>ID</sup>

**Abstract**—CXL interface is the up-to-date technology that enables effective memory expansion by providing a memory-sharing protocol in configuring heterogeneous devices. However, its limited physical bandwidth can be a significant bottleneck for emerging data-intensive applications. In this work, we propose a novel CXL-based memory disaggregation architecture with a real-world prototype demonstration, which overcomes the bandwidth limitation of the CXL interface using near-data processing. The experimental results demonstrate that our design achieves up to  $1.9\times$  better performance/power efficiency than the existing CPU system.

**Index Terms**—Compute express link (CXL), near-data-processing (NDP)

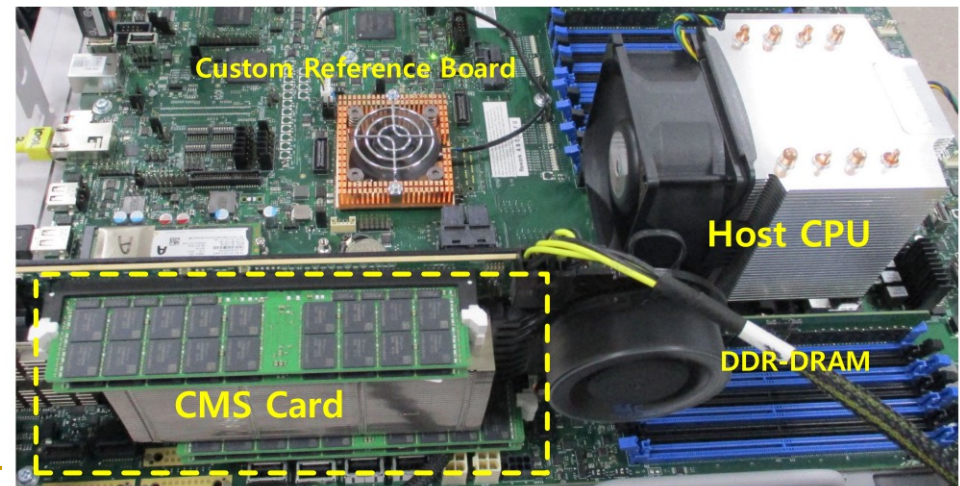


Fig. 6. FPGA prototype of proposed CMS card.



# Processing-in-Memory Landscape Today

## Samsung Processing in Memory Technology at Hot Chips 2023

By **Patrick Kennedy** - August 28, 2023



*Samsung PIM PNM For Transformer Based AI HC35\_Page\_24*

# Real PIM Tutorials [MICRO'23, ISCA'23, ASPLOS'23, HPCA'23]

- June, March, Feb : Lectures + Hands-on labs + Invited talks



## ISCA 2023 Real-World PIM Tutorial

Search

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

Trace: • [start](#)

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

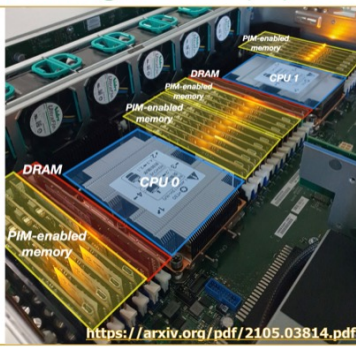
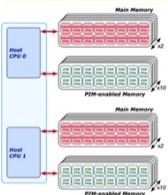
#### Tutorial Description

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.

#### Table of Contents

- [Real-world Processing-in-Memory Systems for Modern Workloads](#)
- [Tutorial Description](#)
- [Organizers](#)
- [Agenda \(June 18, 2023\)](#)
- [Lectures \(tentative\)](#)
- [Hands-on Labs \(tentative\)](#)
- [Learning Materials](#)

<https://events.safari.ethz.ch/isca-pim-tutorial/>

# Real PIM Tutorial [ISCA 2023]

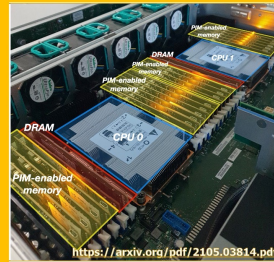
## ■ June 18: Lectures + Hands-on labs + Invited talks

### 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



International Symposium on Computer Architecture (ISCA)

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

<https://www.youtube.com/live/GIb5EgSrWk0?feature=share>

Room: Magnolia 16  
Marriott World Center Orlando  
Orlando, FL, USA  
July 18th, 2023

SAFARI zoom

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

Onur Mutlu Lectures  
33.9K subscribers

Subscribed

57

Share

Download

Clip

...

1,687 views Streamed live on Jun 18, 2023 Livestream - Data-Centric Architectures: Fundamentally Improving Performance and Energy (Spring 2023)

[https://www.youtube.com/  
live/GIb5EgSrWk0](https://www.youtube.com/live/GIb5EgSrWk0)

[https://events.safari.ethz.ch/  
isca-pim-tutorial/](https://events.safari.ethz.ch/isca-pim-tutorial/)

### Tutorial Materials

| Time            | Speaker                             | Title                                                                                                               | Materials                                                             |
|-----------------|-------------------------------------|---------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------|
| 8:55am-9:00am   | Dr. Juan Gómez Luna                 | Welcome & Agenda                                                                                                    | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 9:00am-10:20am  | Prof. Onur Mutlu                    | Memory-Centric Computing                                                                                            | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 10:20am-11:00am | Dr. Juan Gómez Luna                 | Processing-Near-Memory: Real PNM Architectures / Programming General-purpose PIM                                    | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 11:20am-11:50am | Prof. Izzat El Hajj                 | High-throughput Sequence Alignment using Real Processing-in-Memory Systems                                          | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 11:50am-12:30pm | Dr. Christina Giannoula             | SparseP: Towards Efficient Sparse Matrix Vector Multiplication for Real Processing-In-Memory Systems                | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 2:00pm-2:45pm   | Dr. Sukhan Lee                      | Introducing Real-world HBM-PIM Powered System for Memory-bound Applications                                         | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 2:45pm-3:30pm   | Dr. Juan Gómez Luna / Ataberk Olgun | Processing-Using-Memory: Exploiting the Analog Operational Properties of Memory Components / PUM Prototypes: PiDRAM | <a href="#">(PDF)</a> <a href="#">(PPT)</a> <a href="#">(PPT)</a>     |
| 4:00pm-4:40pm   | Dr. Juan Gómez Luna                 | Accelerating Modern Workloads on a General-purpose PIM System                                                       | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 4:40pm-5:20pm   | Dr. Juan Gómez Luna                 | Adoption Issues: How to Enable PIM?                                                                                 | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 5:20pm-5:30pm   | Dr. Juan Gómez Luna                 | Hands-on Lab: Programming and Understanding a Real Processing-in-Memory Architecture                                | <a href="#">(Handout)</a> <a href="#">(PDF)</a> <a href="#">(PPT)</a> |

# Real PIM Tutorial [ASPLOS 2023]

## ■ March 26: Lectures + Hands-on labs + Invited talks

ASPLOS 2023 Real-World PIM Tutorial

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) assess realistic strategies for PIM kernels, and (3)

### Tutorial Materials

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

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

Onur Mutlu Lectures  
32.1K subscribers

Subscribed

33

Share

Clip

Save

views Streamed 7 days ago Livestream - Data-Centric Architectures: Fundamentally Improving Performance and Energy (Spring 2023)

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

<https://events.safari.ethz.ch/asplos-2023/>

<https://www.youtube.com/watch?v=oYCaLcT0Kmo>

<https://events.safari.ethz.ch/asplos-pim-tutorial/>



# Real PIM Tutorial [HPCA 2023]

## ■ February 26: Lectures + Hands-on labs + Invited Talks

HPCA 2023 Real-World PIM Tutorial

Search

Recent Changes Media Manager Sitemap

Trace: - start

### Real-world Processing-in-Memory Architectures

**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, Mythic) are already commercializing real PIM hardware, each with its own design approach and target applications. Several major vendors (e.g., Samsung, SK Hynix, Allbaba) have presented real PIM chip prototypes in the last two years.

#### 2,560-DPU Processing-in-Memory System

Most of these architectures have in common that they place compute units near the memory arrays. But, there is more to come: Academia and Industry are actively exploring other types of PIM by, e.g., exploiting the analog operation of DRAM, SRAM, flash memory and emerging non-volatile memories.

PIM can provide large improvements in both performance and energy consumption, thereby enabling a commercially viable way of dealing with huge amounts of data that is bottlenecking our computing systems. Yet, it is critical to examine and research adoption issues of PIM using especially learnings from real PIM systems that are available today.

This tutorial focuses on the latest advances in PIM technology. 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 using real PIM systems, and (4) shed light on how to enable the adoption of PIM in future computing systems.

### Goal: Processing Inside Memory

Processor Core

Memory

Database

Graphs

Media

Query

Results

Interconnect

- Many questions ... How do we design the:
  - compute-capable memory & controllers?
  - processors & communication units?
  - software & hardware interfaces?
  - system software, compilers, languages?
  - algorithms & theoretical foundations?

HPCA 2023 Tutorial: Real-World Processing-in-Memory Architectures

Onur Mutlu Lectures

32.1K subscribers

50

Share

Clip

Save

1.8K views Streamed 1 month ago Livestream - P&S Data-Centric Architectures: Fundamentally Improving Performance and Energy (Fall 2022)

HPCA 2023 Tutorial: Real-World Processing-in-Memory Architectures

<https://events.safari.ethz.ch/real-pi...>

| Time            | Speaker                 | Title                                                                                                     | Materials                                                             |
|-----------------|-------------------------|-----------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------|
| 8:00am-8:40am   | Prof. Onur Mutlu        | Memory-Centric Computing                                                                                  | <a href="#">PDF</a><br><a href="#">PPT</a>                            |
| 8:40am-10:00am  | Dr. Juan Gómez Luna     | Processing-Near-Memory: Real PNM Architectures Programming General-purpose PIM                            | <a href="#">PDF</a><br><a href="#">PPT</a>                            |
| 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 | <a href="#">PDF</a><br><a href="#">PPT</a>                            |
| 1:30pm-2:10pm   | Dr. Juan Gómez Luna     | Processing-Using-Memory: Exploiting the Analog Operational Properties of Memory Components                | <a href="#">PDF</a><br><a href="#">PPT</a>                            |
| 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?                                                          | <a href="#">PDF</a><br><a href="#">PPT</a>                            |
| 3:40pm-5:40pm   | Dr. Juan Gómez Luna     | Hands-on Lab: Programming and Understanding a Real Processing-in-Memory Architecture                      | <a href="#">Handout</a><br><a href="#">PDF</a><br><a href="#">PPT</a> |

<https://www.youtube.com/watch?v=f5-nT1tbz5w>

<https://events.safari.ethz.ch/real-pim-tutorial/>

- **October 29:** Lectures + Hands-on labs + Invited talks

# 2,560-DPU Processing-in-Memory System

**Host CPU 0**

**Main Memory** (4x8)

**PIM-enabled Memory** (4x16)

**Host CPU 1**

**Main Memory** (4x8)

**PIM-enabled Memory** (4x16)

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

JOAN GOMEZ, CPUA, PhD thesis, University of Valencia, Spain  
 JOAN FERRANDEZ, PhD thesis, University of Valencia, Spain  
 JOAN FERRANDEZ, CPUA, PhD thesis, University of Valencia, Spain  
 JOAN FERRANDEZ, CPUA, PhD thesis, University of Valencia, Spain  
 JOAN FERRANDEZ, CPUA, PhD thesis, University of Valencia, Spain  
 JOAN FERRANDEZ, CPUA, PhD thesis, University of Valencia, Spain

**Live in 74 days**  
 October 29 at 6:00PM

**Notify me**

**DRAM**

**PIM-enabled memory**

**CPU 0**

**CPU 1**

**DRAM**

**PIM-enabled memory**

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

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

**Joan Ferrandez**  
34.6K subscribers

**Subscribe**

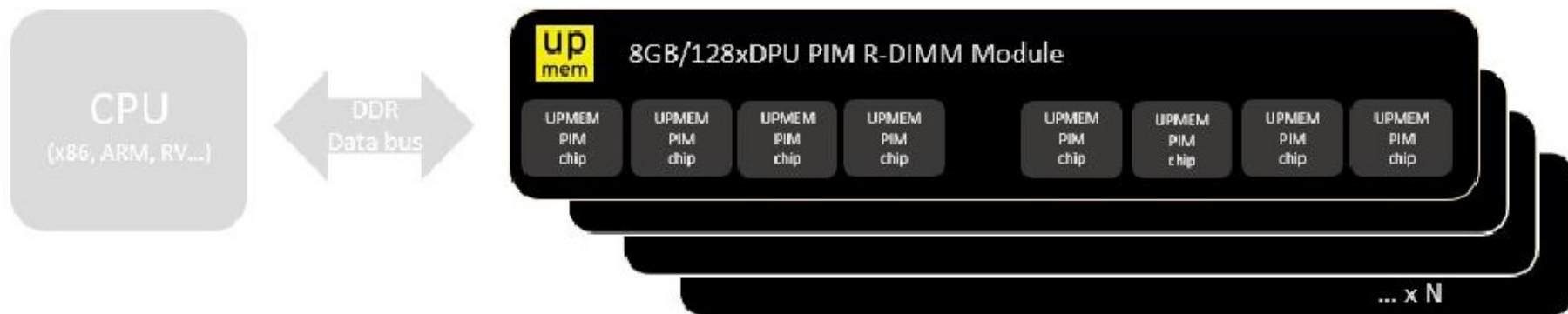
5
1
Share
Save

<https://events.safari.ethz.ch/micro-pim-tutorial>

## 213

# UPMEM Processing-in-DRAM Engine (2019)

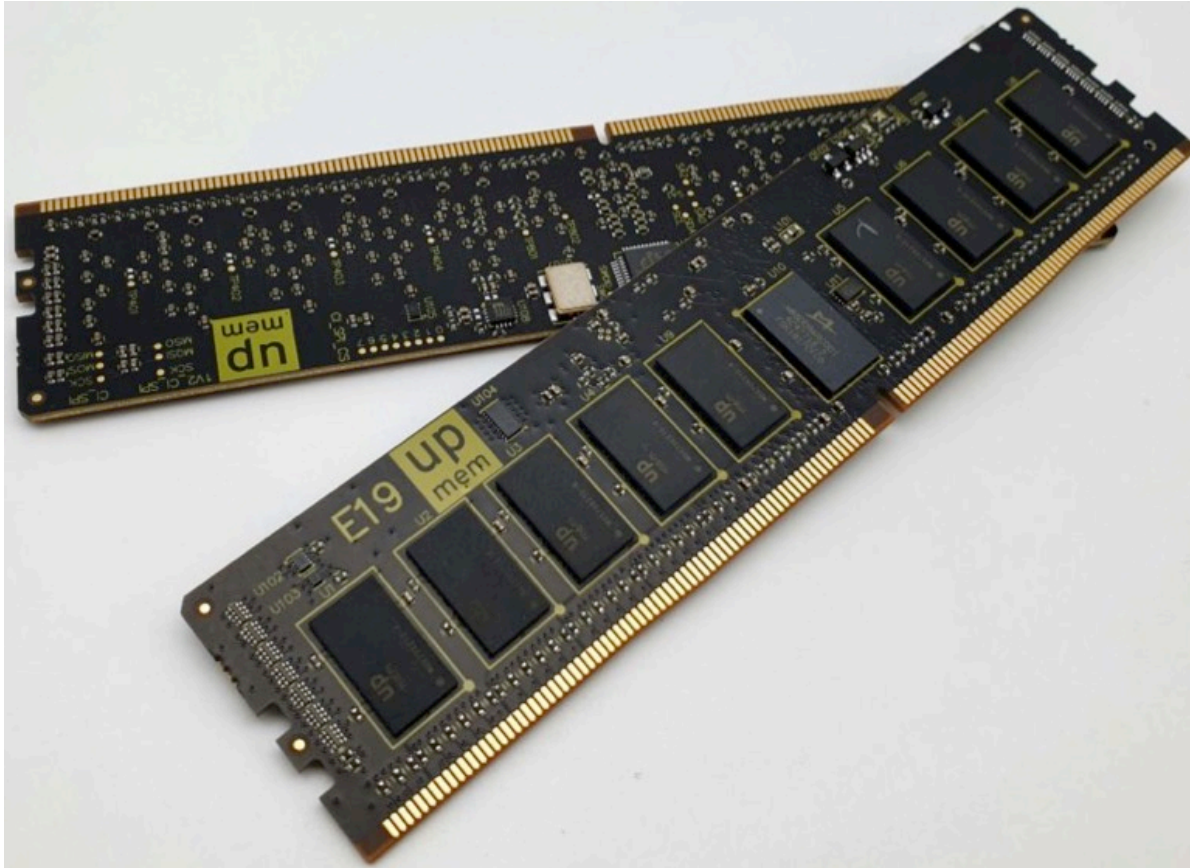
- **Processing in DRAM Engine**
- Includes **standard DIMM modules**, with a **large number of DPU processors** combined with DRAM chips.
- Replaces **standard DIMMs**
  - DDR4 R-DIMM modules
    - 8GB+128 DPUs (16 PIM chips)
    - Standard 2x-nm DRAM process
  - **Large amounts of** compute & memory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





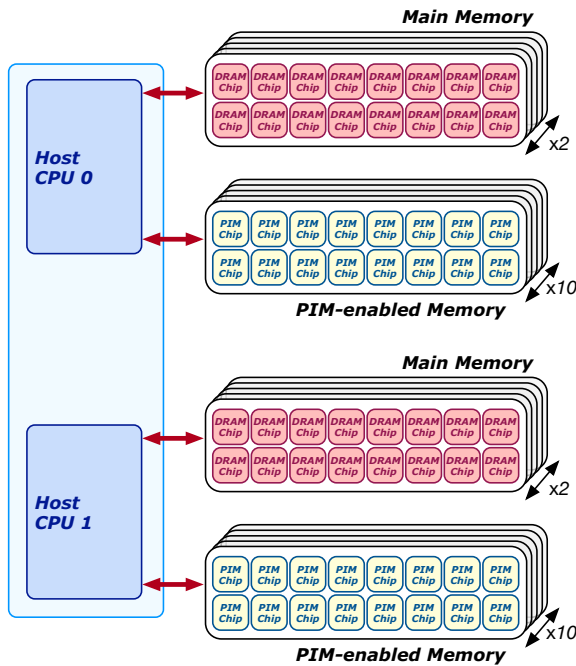
# UPMEM Memory Modules

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





# 2,560-DPU Processing-in-Memory System



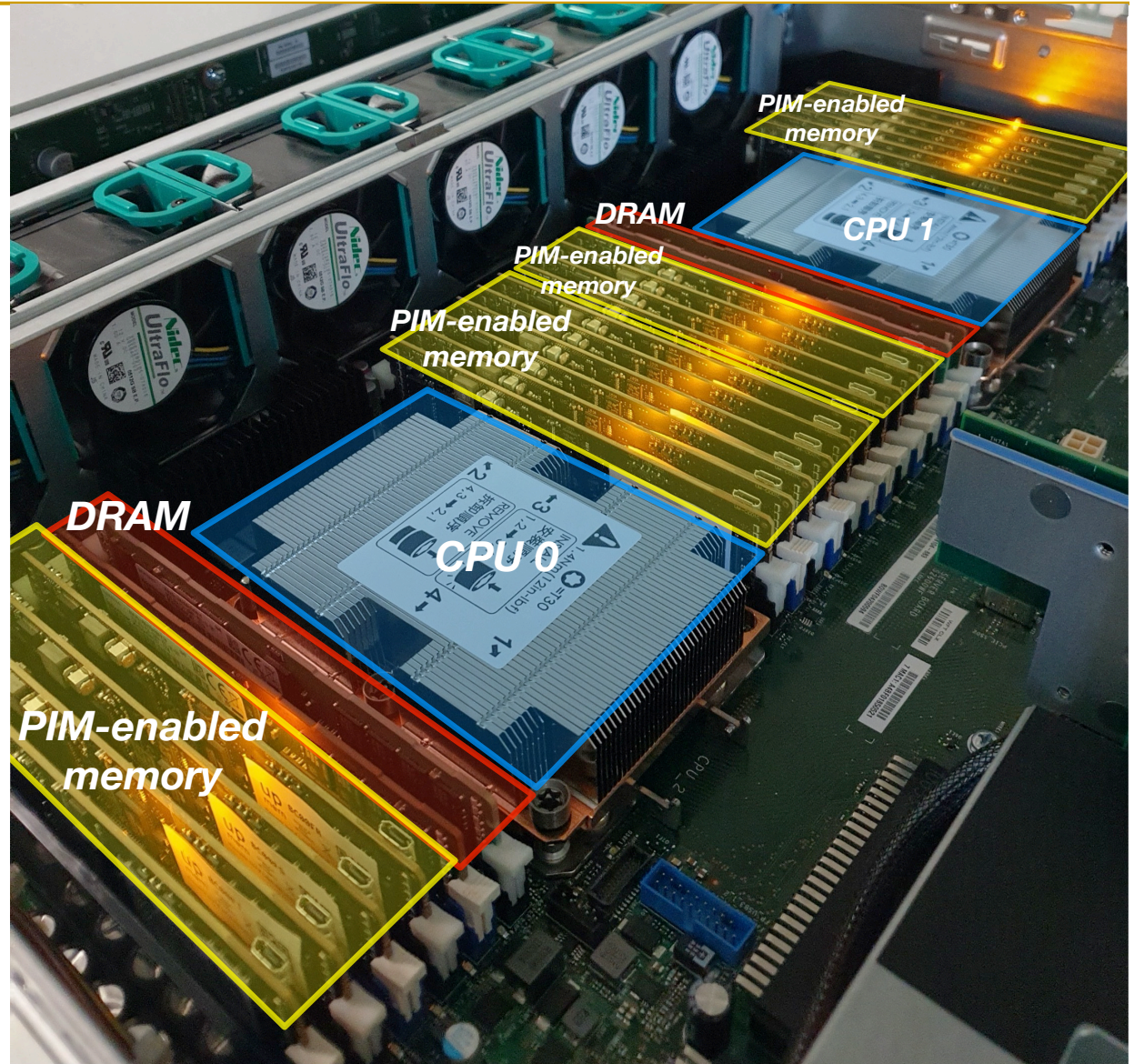
## Benchmarking a New Paradigm: An Experimental Analysis of a Real Processing-in-Memory Architecture

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

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

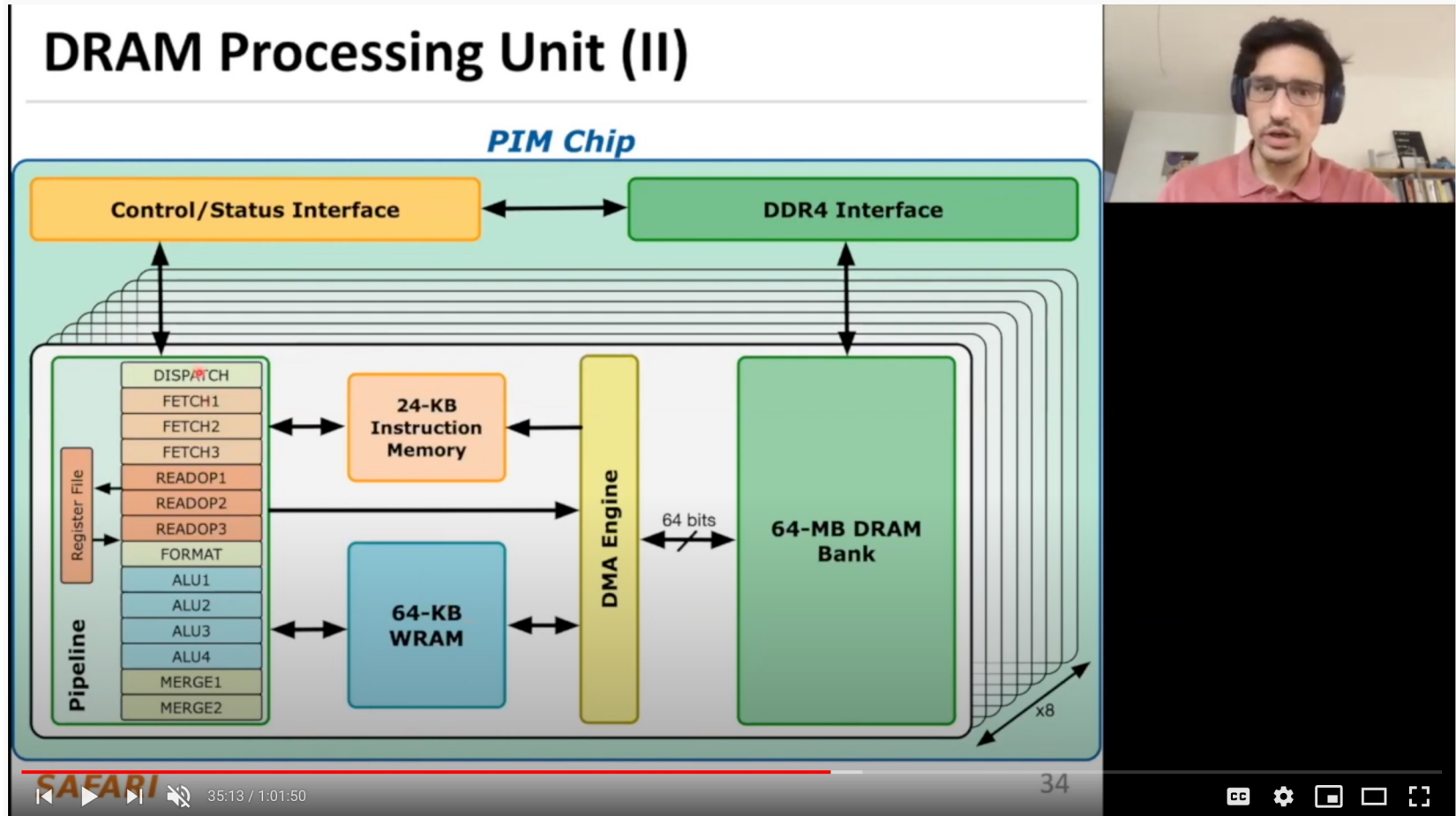
Recent research explores different forms of PIM architectures, motivated by the emergence of new 3D-stacked memory technologies that integrate memory with a logic layer where processing elements can be easily placed. Past works evaluate these architectures in simulation or, at best, with simplified hardware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

This paper provides the first comprehensive analysis of the first publicly-available real-world PIM architecture. We make two key contributions. First, we conduct an experimental characterization of the UPMEM-based PIM system using microbenchmarks to assess various architecture limits such as compute throughput and memory bandwidth, yielding new insights. Second, we present *PrIM (Processing-In-Memory benchmarks)*, a benchmark suite of 16 workloads from different application domains (e.g., dense/sparse linear algebra, databases, data analytics, graph processing, neural networks, bioinformatics, image processing), which we identify as memory-bound. We evaluate the performance and scaling characteristics of PrIM benchmarks on the UPMEM PIM architecture, and compare their performance and energy consumption to their state-of-the-art CPU and GPU counterparts. Our extensive evaluation conducted on two real UPMEM-based PIM systems with 640 and 2,560 DPUs provides new insights about suitability of different workloads to the PIM system, programming recommendations for software designers, and suggestions and hints for hardware and architecture designers of future PIM systems.



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

# More on the UPMEM PIM System



ETH ZÜRICH HAUPTGEBÄUDE

Computer Architecture - Lecture 12d: Real Processing-in-DRAM with UPMEM (ETH Zürich, Fall 2020)

1,120 views • Oct 31, 2020

30 0 SHARE SAVE ...



**Onur Mutlu Lectures**  
16.7K subscribers

ANALYTICS

EDIT VIDEO

<https://www.youtube.com/watch?v=Sscy1Wrr22A&list=PL5Q2soXY2Zi9xidyIgBxUz7xRPS-wisBN&index=26>



# Experimental Analysis of the UPMEM PIM Engine

---

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

JUAN GÓMEZ-LUNA, ETH Zürich, Switzerland

IZZAT EL HAJJ, American University of Beirut, Lebanon

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

CHRISTINA GIANNOULA, ETH Zürich, Switzerland and NTUA, Greece

GERALDO F. OLIVEIRA, ETH Zürich, Switzerland

ONUR MUTLU, ETH Zürich, Switzerland

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

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

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

# 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>

# Recent SRC TECHCON Presentation

## ■ Dr. Juan Gomez-Luna

- Benchmarking Memory-Centric Computing Systems: Analysis of Real Processing-in-Memory Hardware
- Based on two major works
  - <https://arxiv.org/pdf/2105.03814.pdf>
  - <https://arxiv.org/pdf/2207.07886.pdf>



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

Year: 2021, Pages: 1-7

DOI Bookmark: [10.1109/IGSC54211.2021.9651614](https://doi.org/10.1109/IGSC54211.2021.9651614)

### Authors

Juan Gómez-Luna, ETH Zürich

Izzat El Hajj, American University of Beirut

Ivan Fernandez, University of Malaga

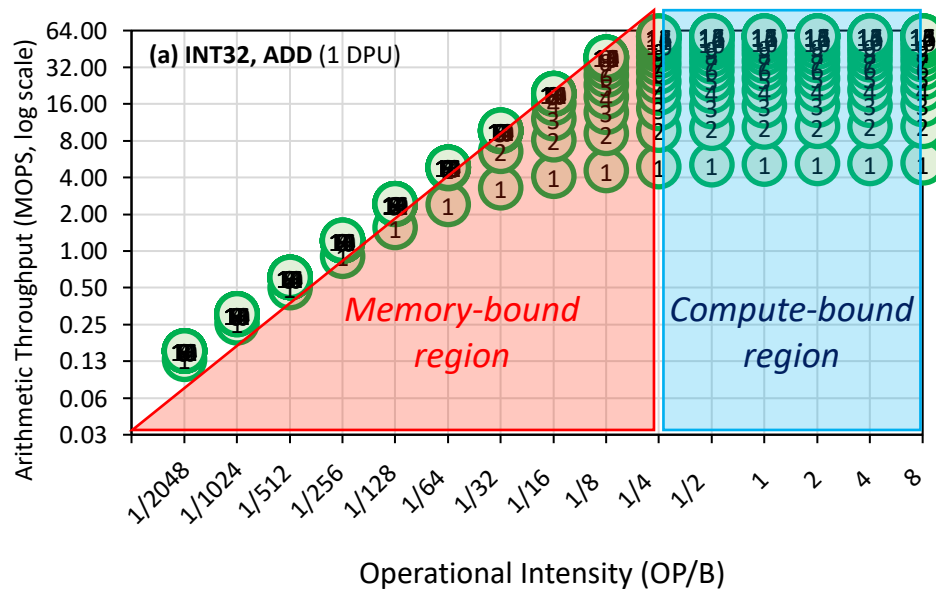
Christina Giannoula, National Technical University of Athens

Geraldo F. Oliveira, ETH Zürich

Onur Mutlu, ETH Zürich



# Key Takeaway 1



The throughput saturation point is as low as  $\frac{1}{4}$  OP/B, i.e., 1 integer addition per every 32-bit element fetched

## KEY TAKEAWAY 1

The UPMEM PIM architecture is fundamentally compute bound. As a result, the most suitable workloads are memory-bound.

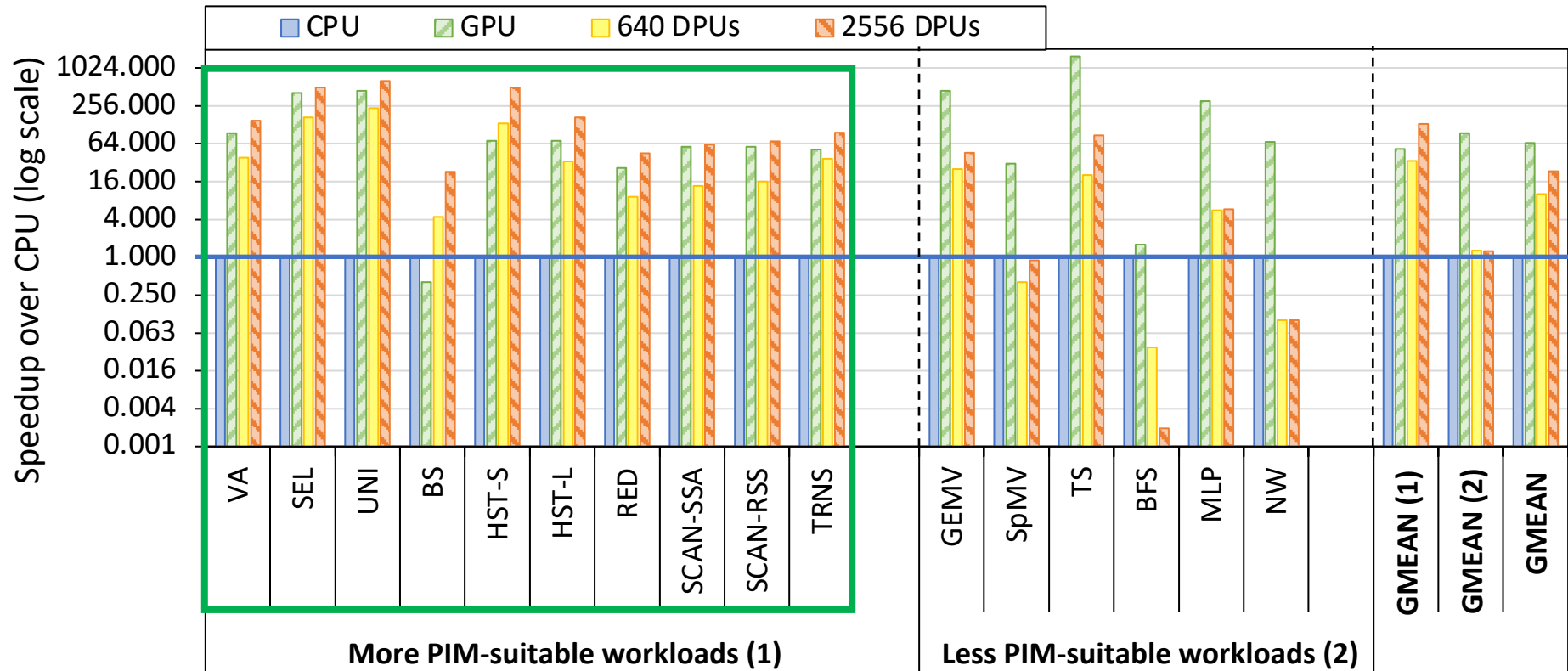


# Key Takeaway 2

Table 4: Evaluated CPU, GPU, and UPMEM-based PIM Systems.

| System                          | Process Node | Processor Cores       |           |                  | Memory    |                 | TDP                |
|---------------------------------|--------------|-----------------------|-----------|------------------|-----------|-----------------|--------------------|
|                                 |              | Total Cores           | Frequency | Peak Performance | Capacity  | Total Bandwidth |                    |
| Intel Xeon E3-1225 v6 CPU [241] | 14 nm        | 4 (8 threads)         | 3.3 GHz   | 26.4 GFLOPS*     | 32 GB     | 37.5 GB/s       | 73 W               |
| NVIDIA Titan V GPU [277]        | 14 nm        | 80 (5,120 SIMD lanes) | 1.2 GHz   | 12,288.0 GFLOPS  | 12 GB     | 652.8 GB/s      | 250 W              |
| 2,556-DPU PIM System            | 2x nm        | 2,556 <sup>†</sup>    | 350 MHz   | 894.6 GOPS       | 159.75 GB | 1.7 TB/s        | 383 W <sup>†</sup> |
| 640-DPU PIM System              | 2x nm        | 640                   | 267 MHz   | 170.9 GOPS       | 40 GB     | 333.75 GB/s     | 96 W <sup>†</sup>  |

\* Estimated GFLOPS = 3.3 GHz × 4 cores × 2 instructions per cycle.  
<sup>†</sup> Estimated TDP =  $\frac{\text{Total DPU}}{\text{DPU}_{\text{chip}}}$  × 1.2 W/chip [199].



## KEY TAKEAWAY 2

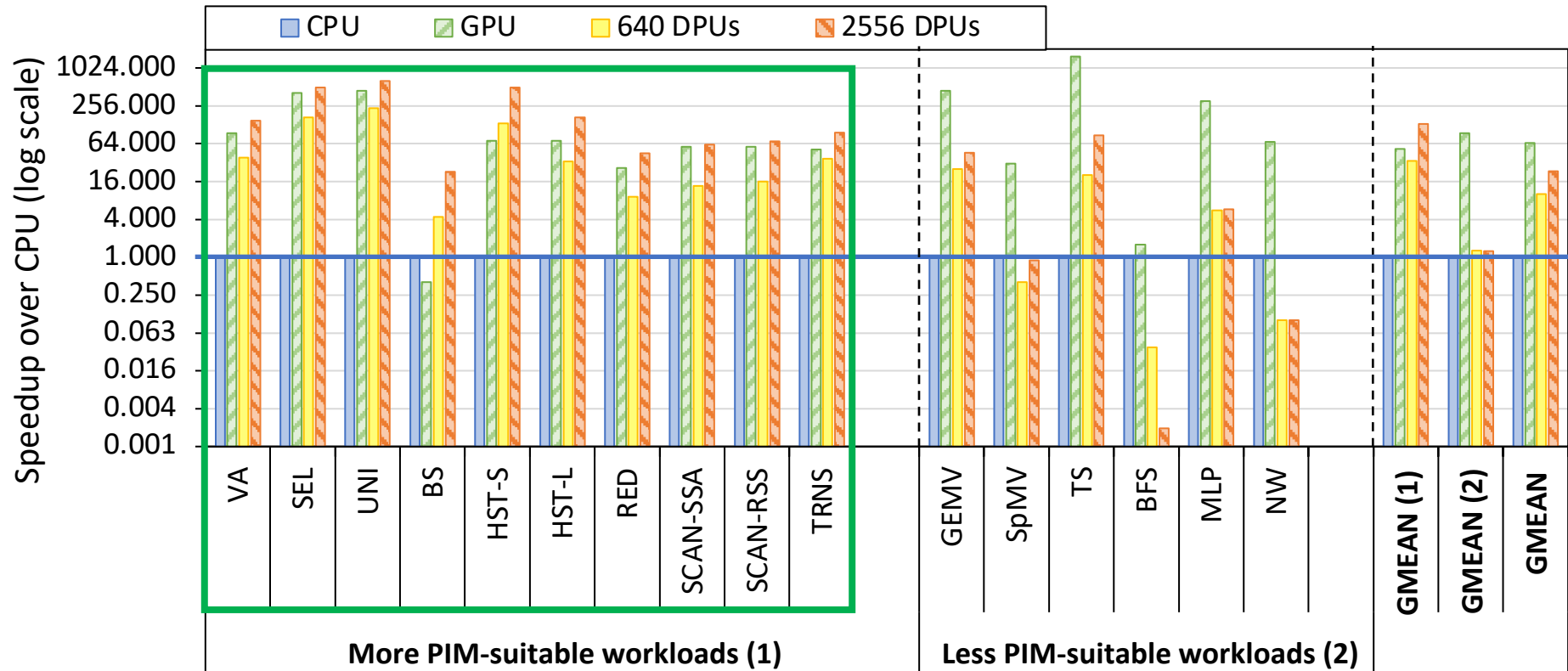
The most well-suited workloads for the UPMEM PIM architecture use no arithmetic operations or use only simple operations (e.g., bitwise operations and integer addition/subtraction).

# Key Takeaway 3

Table 4: Evaluated CPU, GPU, and UPMEM-based PIM Systems.

| System                          | Process Node | Processor Cores       |           |                  | Memory    |                 | TDP                |
|---------------------------------|--------------|-----------------------|-----------|------------------|-----------|-----------------|--------------------|
|                                 |              | Total Cores           | Frequency | Peak Performance | Capacity  | Total Bandwidth |                    |
| Intel Xeon E3-1225 v6 CPU [241] | 14 nm        | 4 (8 threads)         | 3.3 GHz   | 26.4 GFLOPS*     | 32 GB     | 37.5 GB/s       | 73 W               |
| NVIDIA Titan V GPU [277]        | 14 nm        | 80 (5,120 SIMD lanes) | 1.2 GHz   | 12,288.0 GFLOPS  | 12 GB     | 652.8 GB/s      | 250 W              |
| 2,556-DPU PIM System            | 2x nm        | 2,556 <sup>9</sup>    | 350 MHz   | 894.6 GOPS       | 159.75 GB | 1.7 TB/s        | 383 W <sup>†</sup> |
| 640-DPU PIM System              | 2x nm        | 640                   | 267 MHz   | 170.9 GOPS       | 40 GB     | 333.75 GB/s     | 96 W <sup>†</sup>  |

\*Estimated GFLOPS = 3.3 GHz × 4 cores × 2 instructions per cycle.  
<sup>9</sup>Estimated TDP =  $\frac{\text{Total DPU}}{\text{DPU}_1/\text{chip}} \times 1.2 \text{ W/chip}$  [199].



## KEY TAKEAWAY 3

The most well-suited workloads for the UPMEM PIM architecture require little or no communication across DPUs (inter-DPU communication).

# 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

[el1goluj@gmail.com](mailto:el1goluj@gmail.com)

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

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

# 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

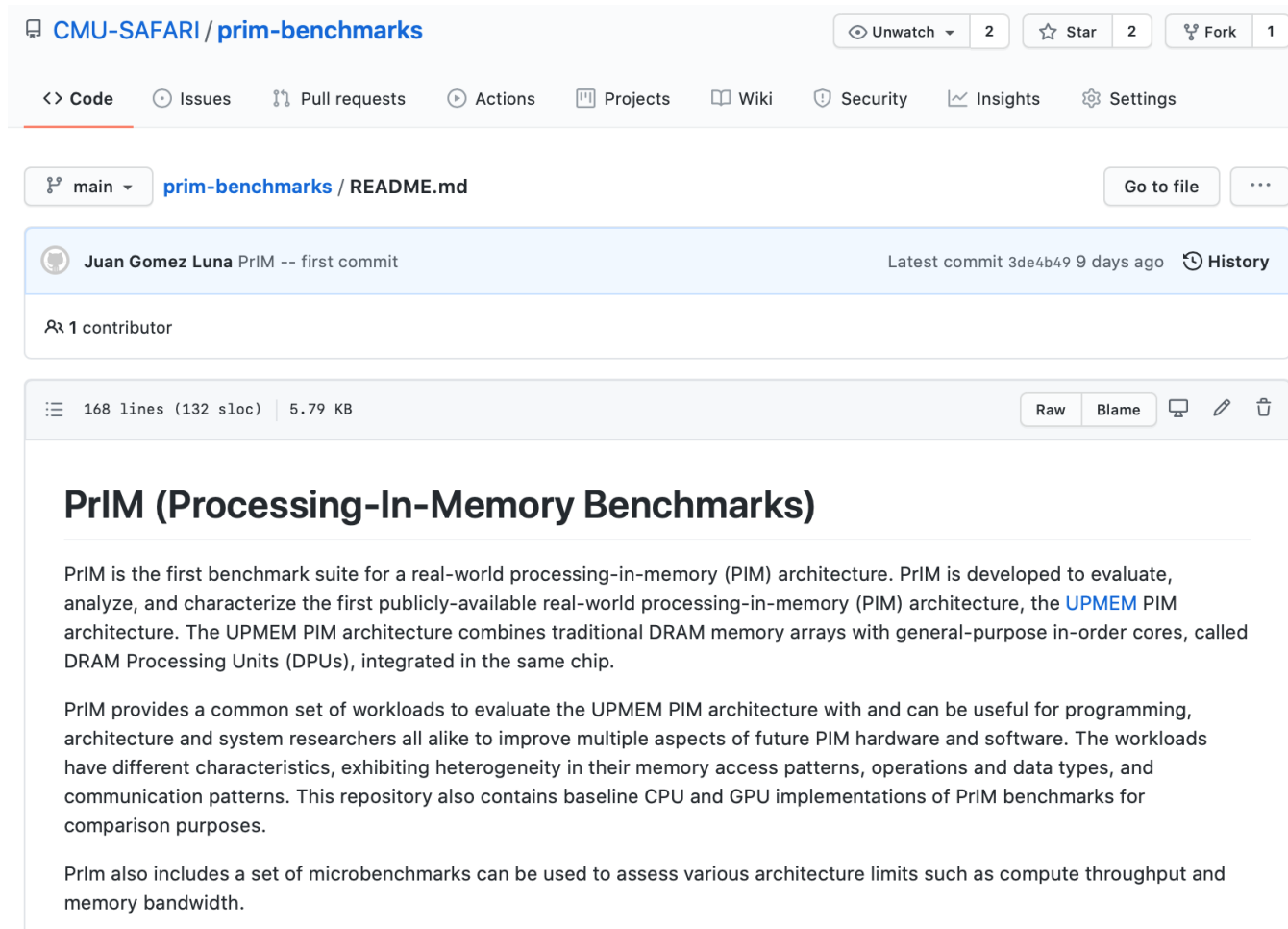
|                   |                                          |                                 |                                                    |                     |                   |
|-------------------|------------------------------------------|---------------------------------|----------------------------------------------------|-------------------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

# 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>



CMU-SAFARI / **prim-benchmarks** Unwatch 2 Star 2 Fork 1

<> 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-LUNA<sup>1</sup>, IZZAT EL HAJJ<sup>2</sup>, IVAN FERNANDEZ<sup>1,3</sup>, CHRISTINA GIANNOULA<sup>1,4</sup>,  
GERALDO F. OLIVEIRA<sup>1</sup>, AND ONUR MUTLU<sup>1</sup>**

<sup>1</sup>ETH Zürich

<sup>2</sup>American University of Beirut

<sup>3</sup>University of Malaga

<sup>4</sup>National Technical University of Athens

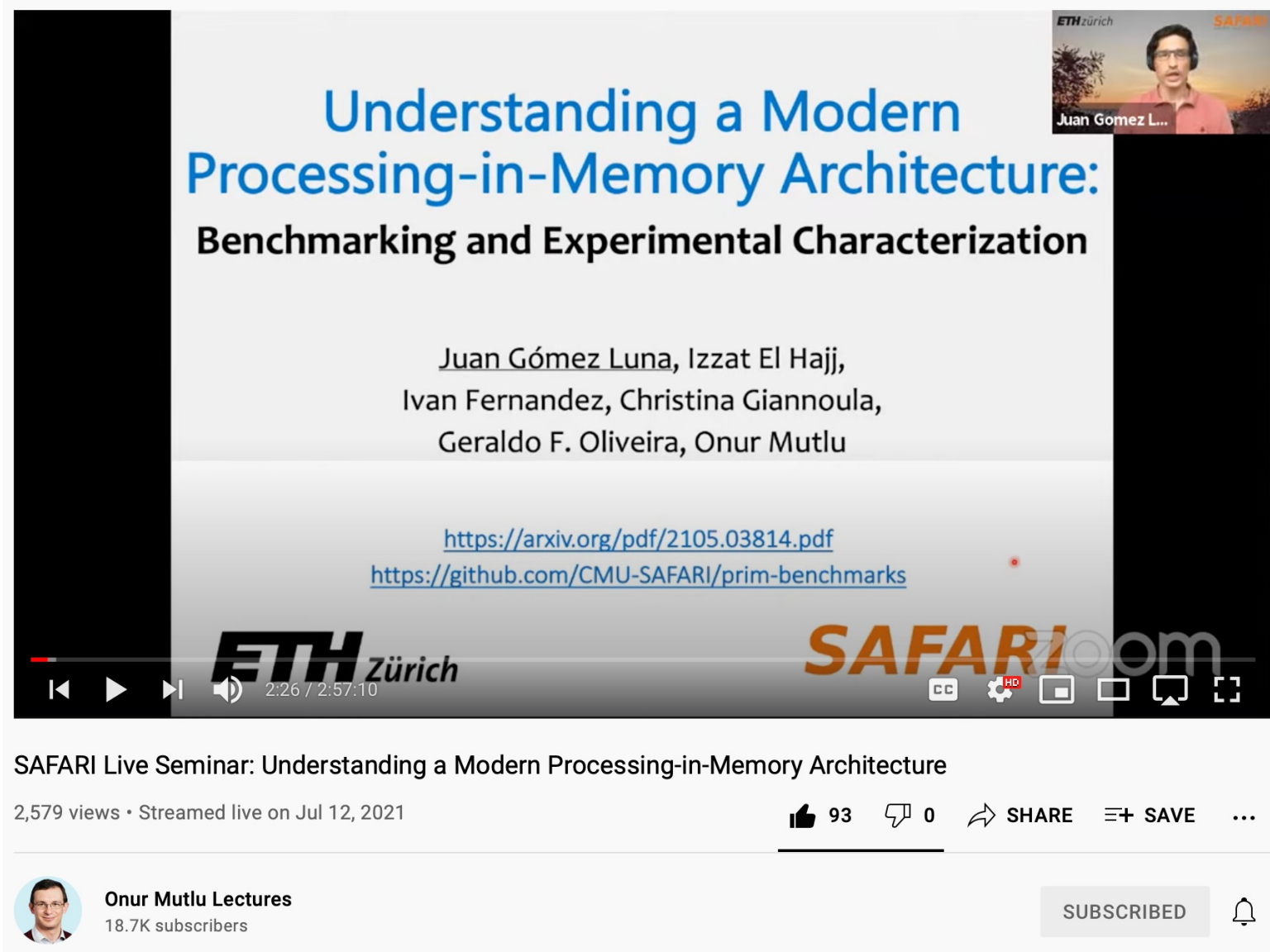
Corresponding author: Juan Gómez-Luna (e-mail: [juang@ethz.ch](mailto:juang@ethz.ch)).

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

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



# Understanding a Modern PIM Architecture



The image shows a YouTube video player interface. The video title is "Understanding a Modern Processing-in-Memory Architecture: Benchmarking and Experimental Characterization". The speaker is Juan Gómez Luna, Izzat El Hajj, Ivan Fernandez, Christina Giannoula, Geraldo F. Oliveira, and Onur Mutlu. The video is from the channel "Onur Mutlu Lectures" and has 18.7K subscribers. The video is titled "SAFARI Live Seminar: Understanding a Modern Processing-in-Memory Architecture" and has 2,579 views. The video is streamed live on Jul 12, 2021. The video player shows the title, speaker names, and links to the arXiv paper and GitHub repository. The video player also shows the ETH Zürich and SAFARI logos. The video player has a progress bar at 2:26 / 2:57:10. The video player has a like button with 93 likes, a comment button with 0 comments, a share button, a save button, and a bell icon. The video player has a "SUBSCRIBED" button.

**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

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

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

93 0 SHARE SAVE ...

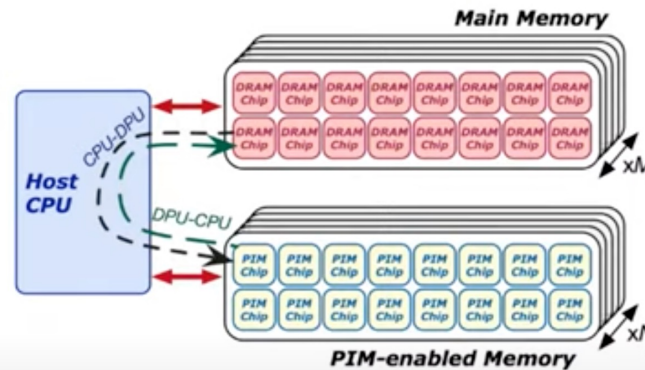
**Onur Mutlu Lectures**  
18.7K subscribers

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 0 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

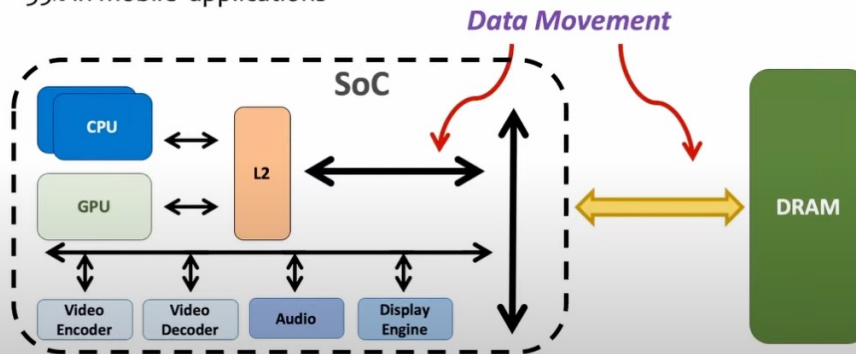
EDIT VIDEO

[https://www.youtube.com/watch?v=D8Hjy2IU9I4&list=PL5Q2soXY2Zi\\_tOTAYm--dYByNPL7JhwR9](https://www.youtube.com/watch?v=D8Hjy2IU9I4&list=PL5Q2soXY2Zi_tOTAYm--dYByNPL7JhwR9)

# 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

\* Pandiyan and Wu, "Quantifying the energy cost of data movement for emerging smart phone workloads on mobile platforms," IISWC 2014

SAFARI

3

Understanding a Modern Processing-in-Memory Arch: Benchmarking & Experimental Characterization; 21m

3,482 views • Premiered Jul 25, 2021

38 0 SHARE SAVE ...



Onur Mutlu Lectures

17.9K subscribers

ANALYTICS

EDIT VIDEO

[https://www.youtube.com/watch?v=Pp9jSU2b9oM&list=PL5Q2soXY2Zi8\\_VVChACnON4sfh2bJ5IrD&index=159](https://www.youtube.com/watch?v=Pp9jSU2b9oM&list=PL5Q2soXY2Zi8_VVChACnON4sfh2bJ5IrD&index=159)

# ML Training on a Real PIM System

---

## Machine Learning Training on a Real Processing-in-Memory System

Juan Gómez-Luna<sup>1</sup> Yuxin Guo<sup>1</sup> Sylvan Brocard<sup>2</sup> Julien Legriel<sup>2</sup>  
Remy Cimadomo<sup>2</sup> Geraldo F. Oliveira<sup>1</sup> Gagandeep Singh<sup>1</sup> Onur Mutlu<sup>1</sup>

<sup>1</sup>ETH Zürich <sup>2</sup>UPMEM

## An Experimental Evaluation of Machine Learning Training on a Real Processing-in-Memory System

Juan Gómez-Luna<sup>1</sup> Yuxin Guo<sup>1</sup> Sylvan Brocard<sup>2</sup> Julien Legriel<sup>2</sup>  
Remy Cimadomo<sup>2</sup> Geraldo F. Oliveira<sup>1</sup> Gagandeep Singh<sup>1</sup> Onur Mutlu<sup>1</sup>

<sup>1</sup>ETH Zürich <sup>2</sup>UPMEM

Short version: <https://arxiv.org/pdf/2206.06022.pdf>

Long version: <https://arxiv.org/pdf/2207.07886.pdf>

<https://www.youtube.com/watch?v=qeukNs5XI3g&t=11226s>

# ML Training on a Real PIM System

---

- Need to optimize data representation
  - (1) fixed-point
  - (2) quantization
  - (3) hybrid precision
- Use **lookup tables (LUTs)** to implement complex functions (e.g., sigmoid)
- Optimize data placement & layout for **streaming**
- Large speedups: **2.8X/27X vs. CPU, 1.3x/3.2x vs. GPU**

# ML Training on Real PIM Talk Video

## Comparison to CPU and GPU (III)

• Decision tree and K-means with Criteo 1TB dataset

**(a) Decision Tree**

| Configuration | Execution Time (ms) |
|---------------|---------------------|
| PIM Kernel    | ~10,000             |
| CPU-PIM       | ~10,000             |
| Inter PIM     | ~10,000             |
| PIM-CPU       | ~10,000             |
| CPU           | ~100,000            |
| GPU           | ~10,000             |

**(b) K-means**

| Configuration | Execution Time (ms) |
|---------------|---------------------|
| PIM Kernel    | ~10,000             |
| CPU-PIM       | ~10,000             |
| Inter PIM     | ~10,000             |
| PIM-CPU       | ~10,000             |
| CPU           | ~100,000            |
| GPU           | ~10,000             |

**PIM version of DTR is 62x faster than the CPU version and 4.5x faster than the GPU version**

**PIM version of KME is 2.7x faster than the CPU version and 3.2x faster than the GPU version**

13:39 / 15:20 • Comparison to CPU and GPU (II) >

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



Onur Mutlu Lectures  
32.9K subscribers

Analytics

Edit video

9



Share



Download



Clip



Save



242 views 11 days ago Livestream - Data-Centric Architectures: Fundamentally Improving Performance and Energy (Spring 2023)  
Evaluating Machine Learning Workloads on Memory-centric Computing Systems



# ML Training on Real PIM Systems

---

- Juan Gómez Luna, Yuxin Guo, Sylvan Brocard, Julien Legriel, Remy Cimadomo, Geraldo F. Oliveira, Gagandeep Singh, and Onur Mutlu,  
**"Evaluating Machine Learning Workloads on Memory-Centric Computing Systems"**  
*Proceedings of the 2023 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*, Raleigh, North Carolina, USA, April 2023.  
[[arXiv version](#), 16 July 2022.]  
[[PIM-ML Source Code](#)]  
***Best paper session.***

## An Experimental Evaluation of Machine Learning Training on a Real Processing-in-Memory System

Juan Gómez-Luna<sup>1</sup> Yuxin Guo<sup>1</sup> Sylvan Brocard<sup>2</sup> Julien Legriel<sup>2</sup>  
Remy Cimadomo<sup>2</sup> Geraldo F. Oliveira<sup>1</sup> Gagandeep Singh<sup>1</sup> Onur Mutlu<sup>1</sup>  
<sup>1</sup>ETH Zürich <sup>2</sup>UPMEM

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



# SpMV Multiplication on Real PIM Systems

---

- Appears at SIGMETRICS 2022

## ***SparseP*: Towards Efficient Sparse Matrix Vector Multiplication on Real Processing-In-Memory Systems**

CHRISTINA GIANNOULA, ETH Zürich, Switzerland and National Technical University of Athens, Greece

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

JUAN GÓMEZ-LUNA, ETH Zürich, Switzerland

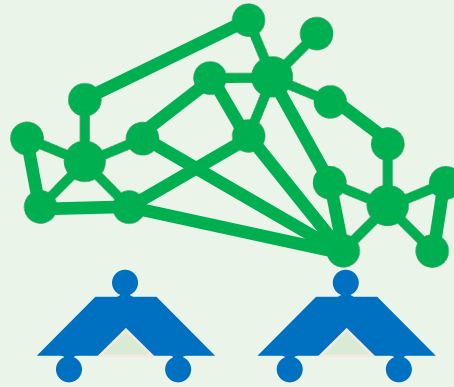
NECTARIOS KOZIRIS, National Technical University of Athens, Greece

GEORGIOS GOUMAS, National Technical University of Athens, Greece

ONUR MUTLU, ETH Zürich, Switzerland

<https://arxiv.org/pdf/2201.05072.pdf>

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



# SparseP

Towards Efficient Sparse Matrix Vector Multiplication  
on Real Processing-In-Memory Architectures

Christina Giannoula

Ivan Fernandez, Juan Gomez-Luna,

Nectarios Koziris, Georgios Goumas, Onur Mutlu

**SAFARI** **ETH** zürich

 **CSLab**  
National Technical University of Athens



UNIVERSIDAD  
DE MÁLAGA

# SparseP: Key Contributions

## 1. Efficient SpMV kernels for current & future PIM systems

- SparseP library = 25 SpMV kernels
  - Compression, data types, data partitioning, synchronization, load balancing

SparseP is Open-Source

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

## 2. Comprehensive analysis of SpMV on the first commercially-available real PIM system



- 26 sparse matrices
- Comparisons to state-of-the-art CPU and GPU systems
- Recommendations for software, system and hardware designers

Recommendations for Architects and Programmers

Full Paper: <https://arxiv.org/pdf/2201.05072.pdf>

# SparseP Talk Video



The screenshot shows a YouTube video player with a presentation slide. The slide features the SparseP logo, which consists of a green network graph above two blue stylized human figures. Below the logo, the title "SparseP" is written in a large blue font, followed by the subtitle "Towards Efficient Sparse Matrix Vector Multiplication on Real Processing-In-Memory Architectures" in a smaller blue font. The presenter's name, "Christina Giannoula", is listed in red, followed by the names of other contributors: "Ivan Fernandez, Juan Gomez-Luna, Nectarios Koziris, Georgios Goumas, Onur Mutlu". At the bottom of the slide, there are logos for "SAFARI ETH zürich", "CSLab", and several university seals including "National Technical University of Athens", "UNIVERSITÄT DUISBURG ESSEN", and "UNIVERSIDAD DE MÁLAGA". A small video inset in the top right corner shows the presenter, Christina Giannoula. The video player interface includes a progress bar at 0:02 / 55:25, a volume icon, and various control buttons like play, pause, and full screen.

Processing-in-Memory Course: Lecture 11: SpMV on a Real PIM Architecture - Spring 2022

149 views • Streamed live on May 19, 2022

👍 12    🗑 DISLIKE    ➦ SHARE    ⬇ DOWNLOAD    ✂ CLIP    ≡+ SAVE    ...



Onur Mutlu Lectures  
25K subscribers

ANALYTICS

EDIT VIDEO

# More on SparseP

---

Christina Giannoula, Ivan Fernandez, Juan Gomez-Luna, Nectarios Koziris, Georgios Goumas, and Onur Mutlu,

## **"SparseP: Towards Efficient Sparse Matrix Vector Multiplication on Real Processing-In-Memory Architectures"**

*Proceedings of the ACM International Conference on Measurement and Modeling of Computer Systems (**SIGMETRICS**), Mumbai, India, June 2022.*

[\[Extended arXiv Version\]](#)

[\[Abstract\]](#)

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

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

[\[SparseP Source Code\]](#)

[\[Talk Video \(16 minutes\)\]](#)

[\[Long Talk Video \(55 minutes\)\]](#)

## **SparseP: Towards Efficient Sparse Matrix Vector Multiplication on Real Processing-In-Memory Systems**

CHRISTINA GIANNOULA, ETH Zürich, Switzerland and National Technical University of Athens, Greece

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

JUAN GÓMEZ-LUNA, ETH Zürich, Switzerland

NECTARIOS KOZIRIS, National Technical University of Athens, Greece

GEORGIOS GOUMAS, National Technical University of Athens, Greece

ONUR MUTLU, ETH Zürich, Switzerland

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

# SpMV Multiplication on Real PIM Systems

---

- Appears at SIGMETRICS 2022

## ***SparseP*: Towards Efficient Sparse Matrix Vector Multiplication on Real Processing-In-Memory Systems**

CHRISTINA GIANNOULA, ETH Zürich, Switzerland and National Technical University of Athens, Greece

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

JUAN GÓMEZ-LUNA, ETH Zürich, Switzerland

NECTARIOS KOZIRIS, National Technical University of Athens, Greece

GEORGIOS GOUMAS, National Technical University of Athens, Greece

ONUR MUTLU, ETH Zürich, Switzerland

<https://arxiv.org/pdf/2201.05072.pdf>

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

# Transcendental Functions on Real PIM Systems

---

- Maurus Item, Juan Gómez Luna, Yuxin Guo, Geraldo F. Oliveira, Mohammad Sadrosadati, and Onur Mutlu,

## **"TransPimLib: Efficient Transcendental Functions for Processing-in-Memory Systems"**

*Proceedings of the 2023 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*, Raleigh, North Carolina, USA, April 2023.

[[arXiv version](#)]

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

[[TransPimLib Source Code](#)]

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

## **TransPimLib: Efficient Transcendental Functions for Processing-in-Memory Systems**

Maurus Item  
Geraldo F. Oliveira

Juan Gómez-Luna  
Mohammad Sadrosadati

Yuxin Guo  
Onur Mutlu

ETH Zürich

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



# Sequence Alignment on Real PIM Systems

---

- Safaa Diab, Amir Nassereldine, Mohammed Alser, Juan Gómez Luna, Onur Mutlu, and Izzat El Hajj,  
**"A Framework for High-throughput Sequence Alignment using Real Processing-in-Memory Systems"**  
**Bioinformatics**, [published online on] 27 March 2023.  
[[Online link at Bioinformatics Journal](#)]  
[[arXiv preprint](#)]  
[[AiM Source Code](#)]

## A Framework for High-throughput Sequence Alignment using Real Processing-in-Memory Systems

Safaa Diab<sup>1</sup> Amir Nassereldine<sup>1</sup> Mohammed Alser<sup>2</sup> Juan Gómez Luna<sup>2</sup>  
Onur Mutlu<sup>2</sup> Izzat El Hajj<sup>1</sup>

<sup>1</sup>American University of Beirut <sup>2</sup>ETH Zürich

<https://github.com/CMU-SAFARI/alignment-in-memory>



## Summary

- Sequence alignment on traditional systems is limited by the **memory bandwidth bottleneck**
- **Processing-in-memory (PIM)** overcomes this bottleneck by placing cores near the memory
- Our framework, **Alignment-in-Memory (AIM)**, is a PIM framework that supports multiple alignment algorithms (NW, SWG, GenASM, WFA)
  - Implemented on UPMEM, the first real PIM system
- Results show **substantial speedups over both CPUs (1.8X-28X) and GPUs (1.2X-2.7X)**
- AIM is available at:
  - <https://github.com/CMU-SAFARI/alignment-in-memory>

# 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



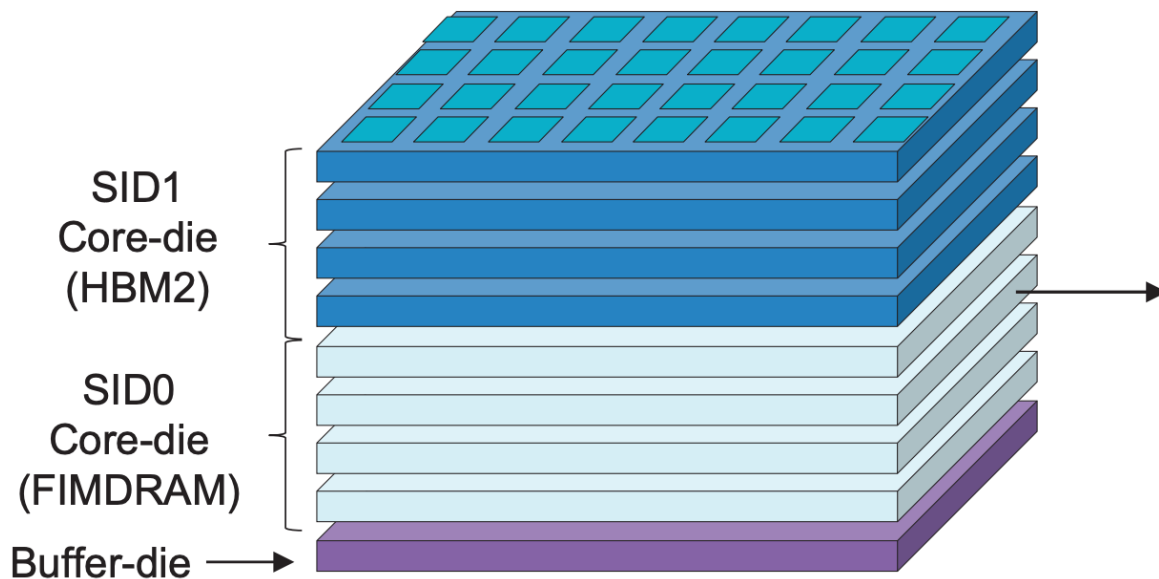
*The new architecture will deliver over twice the system performance and reduce energy consumption by more than 70%*

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 Kwon<sup>1</sup>, Suk Han Lee<sup>1</sup>, Jaehoon Lee<sup>1</sup>, Sang-Hyuk Kwon<sup>1</sup>, Je Min Ryu<sup>1</sup>, Jong-Pil Son<sup>1</sup>, Seongil O<sup>1</sup>, Hak-Soo Yu<sup>1</sup>, Haesuk Lee<sup>1</sup>, Soo Young Kim<sup>1</sup>, Youngmin Cho<sup>1</sup>, Jin Guk Kim<sup>1</sup>, Jongyoon Choi<sup>1</sup>, Hyun-Sung Shin<sup>1</sup>, Jin Kim<sup>1</sup>, BengSeng Phuah<sup>1</sup>, HyoungMin Kim<sup>1</sup>, Myeong Jun Song<sup>1</sup>, Ahn Choi<sup>1</sup>, Daeho Kim<sup>1</sup>, SooYoung Kim<sup>1</sup>, Eun-Bong Kim<sup>1</sup>, David Wang<sup>2</sup>, Shinhaeng Kang<sup>1</sup>, Yuhwan Ro<sup>3</sup>, Seungwoo Seo<sup>3</sup>, JoonHo Song<sup>3</sup>, Jaeyoun Youn<sup>1</sup>, Kyomin Sohn<sup>1</sup>, Nam Sung Kim<sup>1</sup>

<sup>1</sup>Samsung Electronics, Hwaseong, Korea

<sup>2</sup>Samsung Electronics, San Jose, CA

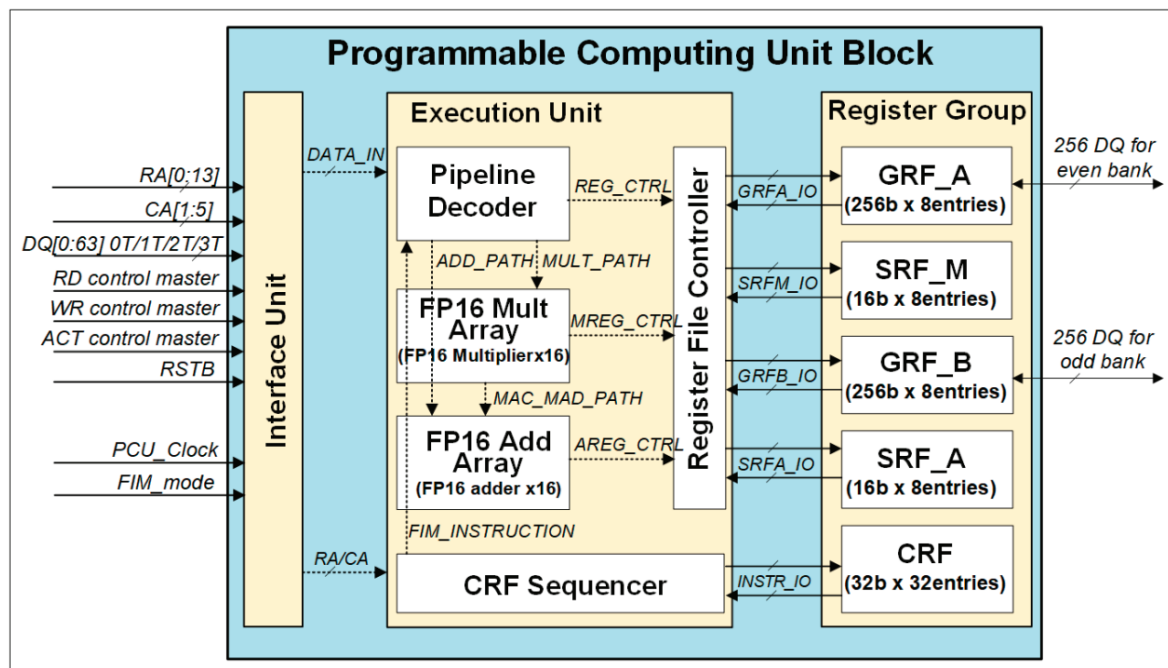
<sup>3</sup>Samsung Electronics, Suwon, Korea

# 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 Kwon<sup>1</sup>, Suk Han Lee<sup>1</sup>, Jaehoon Lee<sup>1</sup>, Sang-Hyuk Kwon<sup>1</sup>, Je Min Ryu<sup>1</sup>, Jong-Pil Son<sup>1</sup>, Seongil O<sup>1</sup>, Hak-Soo Yu<sup>1</sup>, Haesuk Lee<sup>1</sup>, Soo Young Kim<sup>1</sup>, Youngmin Cho<sup>1</sup>, Jin Guk Kim<sup>1</sup>, Jongyoon Choi<sup>1</sup>, Hyun-Sung Shin<sup>1</sup>, Jin Kim<sup>1</sup>, BengSeng Phuah<sup>1</sup>, HyungMin Kim<sup>1</sup>, Myeong Jun Song<sup>1</sup>, Ahn Choi<sup>1</sup>, Daeho Kim<sup>1</sup>, SooYoung Kim<sup>1</sup>, Eun-Bong Kim<sup>1</sup>, David Wang<sup>2</sup>, Shinhaeng Kang<sup>3</sup>, Yuhwan Ro<sup>3</sup>, Seungwoo Seo<sup>3</sup>, JoonHo Song<sup>3</sup>, Jaeyoun Youn<sup>1</sup>, Kyomin Sohn<sup>1</sup>, Nam Sung Kim<sup>1</sup>

<sup>1</sup>Samsung Electronics, Hwaseong, Korea  
<sup>2</sup>Samsung Electronics, San Jose, CA  
<sup>3</sup>Samsung 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 Kwon<sup>1</sup>, Suk Han Lee<sup>1</sup>, Jaehoon Lee<sup>1</sup>, Sang-Hyuk Kwon<sup>1</sup>, Je Min Ryu<sup>1</sup>, Jong-Pil Son<sup>1</sup>, Seongil O<sup>1</sup>, Hak-Soo Yu<sup>1</sup>, Haesuk Lee<sup>1</sup>, Soo Young Kim<sup>1</sup>, Youngmin Cho<sup>1</sup>, Jin Guk Kim<sup>1</sup>, Jongyoon Choi<sup>1</sup>, Hyun-Sung Shin<sup>1</sup>, Jin Kim<sup>1</sup>, BengSeng Phuah<sup>1</sup>, HyungMin Kim<sup>1</sup>, Myeong Jun Song<sup>1</sup>, Ahn Choi<sup>1</sup>, Daeho Kim<sup>1</sup>, SooYoung Kim<sup>1</sup>, Eun-Bong Kim<sup>1</sup>, David Wang<sup>2</sup>, Shinhaeng Kang<sup>1</sup>, Yuhwan Ro<sup>1</sup>, Seungwoo Seo<sup>1</sup>, JoonHo Song<sup>1</sup>, Jaeyoun Youn<sup>1</sup>, Kyomin Sohn<sup>1</sup>, Nam Sung Kim<sup>1</sup>

<sup>1</sup>Samsung Electronics, Hwaseong, Korea

<sup>2</sup>Samsung Electronics, San Jose, CA

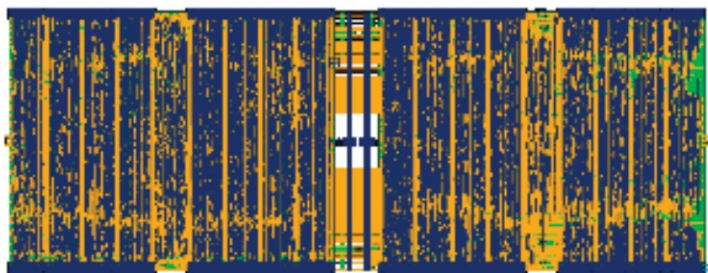
<sup>3</sup>Samsung Electronics, Suwon, Korea



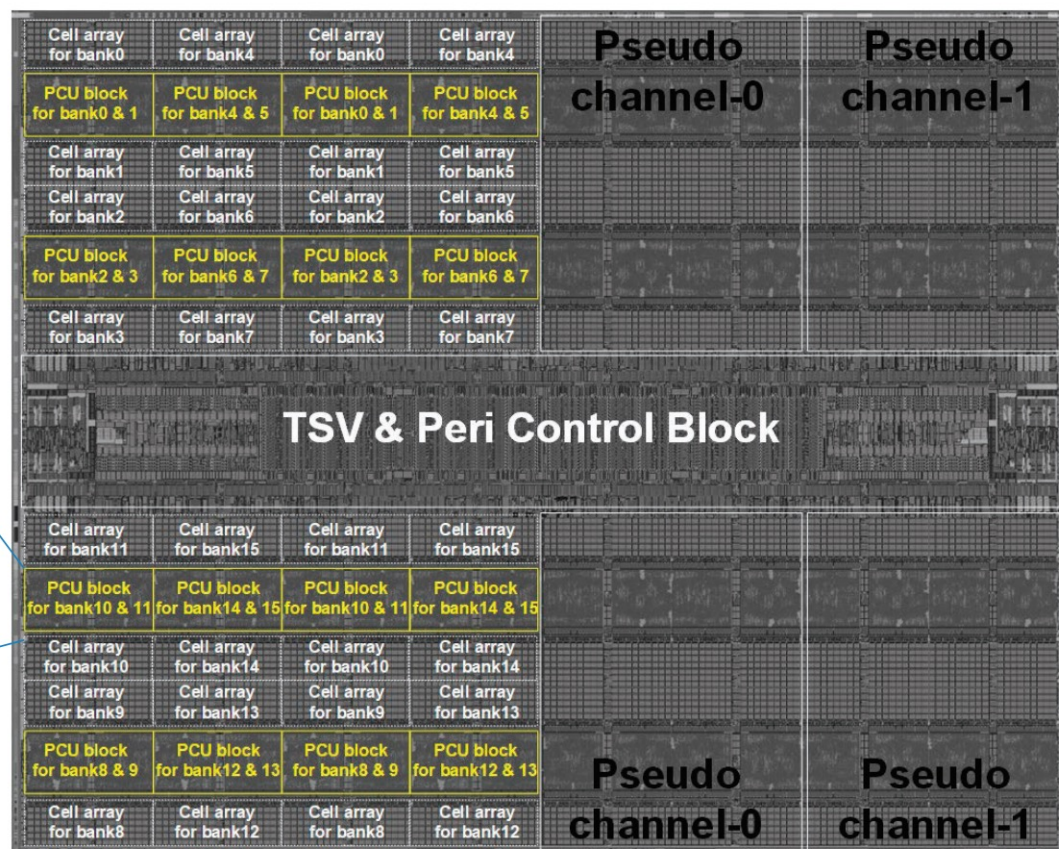
# Samsung Function-in-Memory DRAM (2021)

## 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

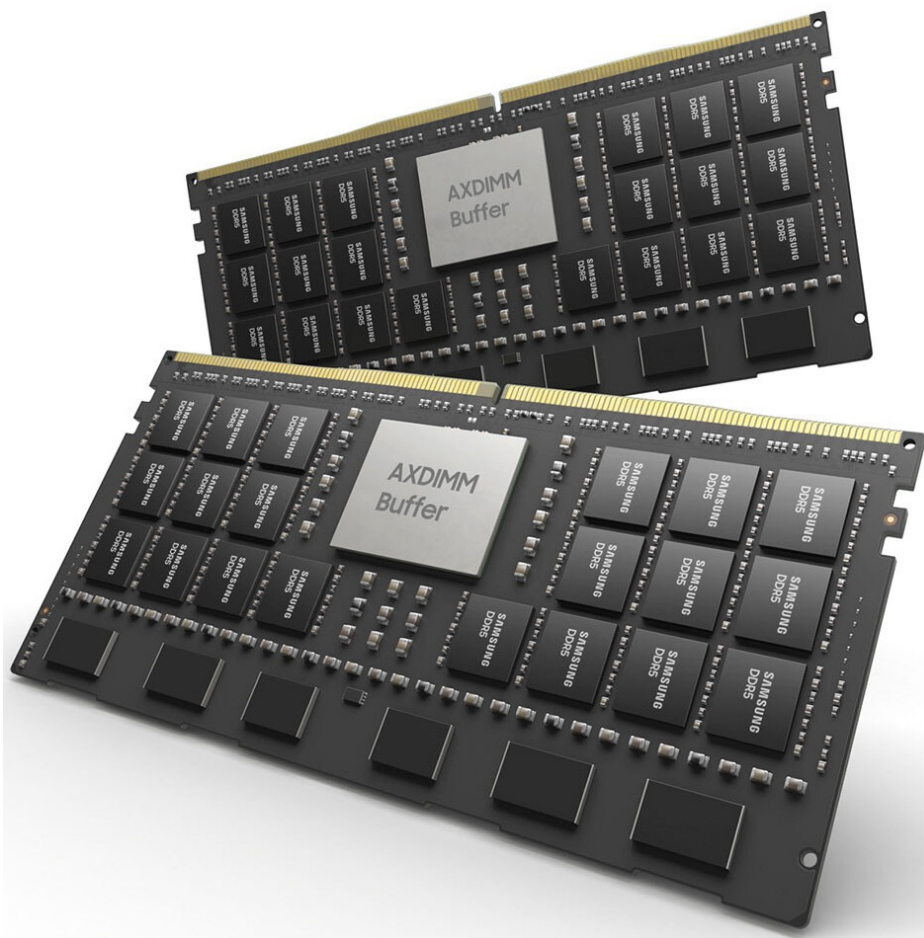
Young-Cheon Kwon<sup>1</sup>, Suk Han Lee<sup>1</sup>, Jaehoon Lee<sup>1</sup>, Sang-Hyuk Kwon<sup>1</sup>, Je Min Ryu<sup>1</sup>, Jong-Pil Son<sup>1</sup>, Seongil O<sup>1</sup>, Hak-Soo Yu<sup>1</sup>, Haesuk Lee<sup>1</sup>, Soo Young Kim<sup>1</sup>, Youngmin Cho<sup>1</sup>, Jin Guk Kim<sup>1</sup>, Jongyeon Choi<sup>1</sup>, Hyun-Sung Shim<sup>1</sup>, Jin Kim<sup>1</sup>, BengSeng Phuah<sup>1</sup>, HyounMin Kim<sup>1</sup>, Myeong Jun Song<sup>1</sup>, Ahn Chai<sup>1</sup>, Daeho Kim<sup>1</sup>, SooYoung Kim<sup>1</sup>, Eun-Bong Kim<sup>1</sup>, David Wang<sup>2</sup>, Shintae Kang<sup>3</sup>, Yulwan Ro<sup>3</sup>, Seungwoo Seo<sup>3</sup>, JoonHo Song<sup>3</sup>, Jaeyoun Yoon<sup>1</sup>, Kyomin Sohn<sup>1</sup>, Nam Sung Kim<sup>1</sup>

<sup>1</sup>Samsung Electronics, Hwaseong, Korea  
<sup>2</sup>Samsung Electronics, San Jose, CA  
<sup>3</sup>Samsung Electronics, Suwon, Korea

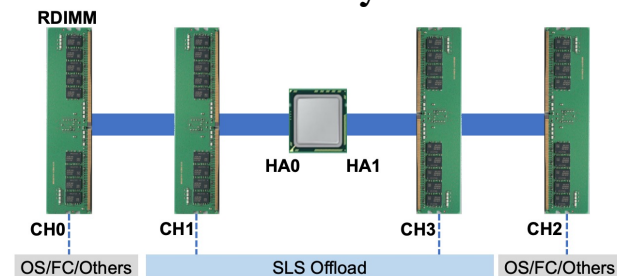


# Samsung AxDIMM (2021)

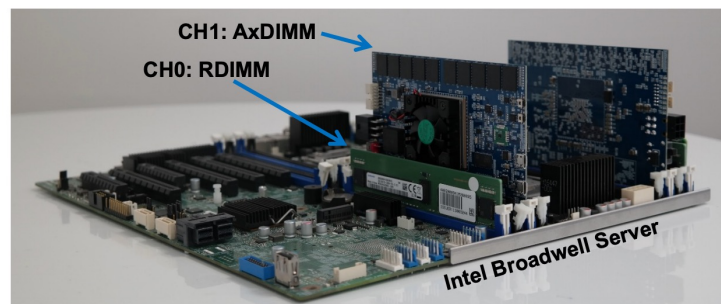
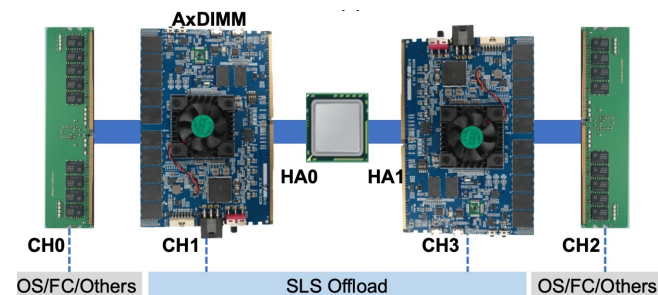
- DDRx-PIM
  - DLRM recommendation system



Baseline System



AxDIMM System



# SK Hynix Accelerator-in-Memory (2022)

## SK hynix Develops PIM, Next-Generation AI Accelerator

February 16, 2022



Seoul, February 16, 2022

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

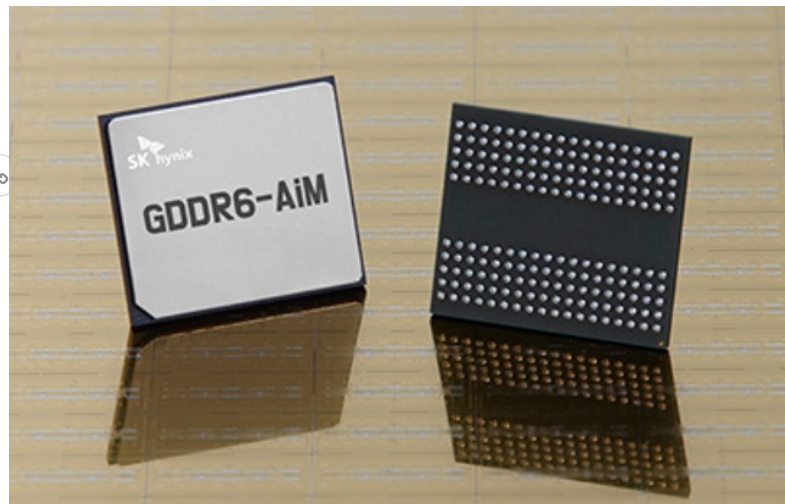
*\*PIM(Processing In Memory): A next-generation technology that provides a solution for data congestion issues for AI and big data by adding computational functions to semiconductor memory*

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

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

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

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



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

Seongju Lee, SK hynix, Icheon, Korea

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

# SK Hynix Accelerator-in-Memory (2022)

**System Architecture and Software Stack for GDDR6-AiM**

Yongkee Kwon and Chanwook Park  
SK hynix inc.

5:42 / 6:27:38

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



Onur Mutlu Lectures  
32.1K subscribers

Analytics

Edit video

33



Share

Download

Clip

Save



1,146 views Streamed live on Mar 26, 2023 Livestream - Data-Centric Architectures: Fundamentally Improving Performance and Energy (Spring 2023)

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

<https://events.safari.ethz.ch/asplos-...>

<https://www.youtube.com/watch?v=oYCaLcT0Kmo>



# AliBaba PIM Recommendation System (2022)

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

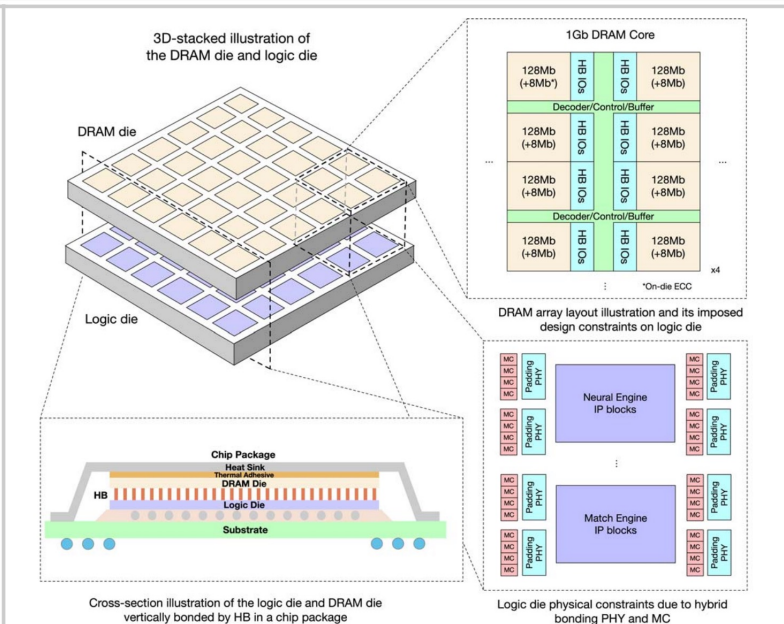


Figure 29.1.2: Illustration of 3D-stacked chip, cross-illustration of package, DRAM array layout and design blocks on logic die.

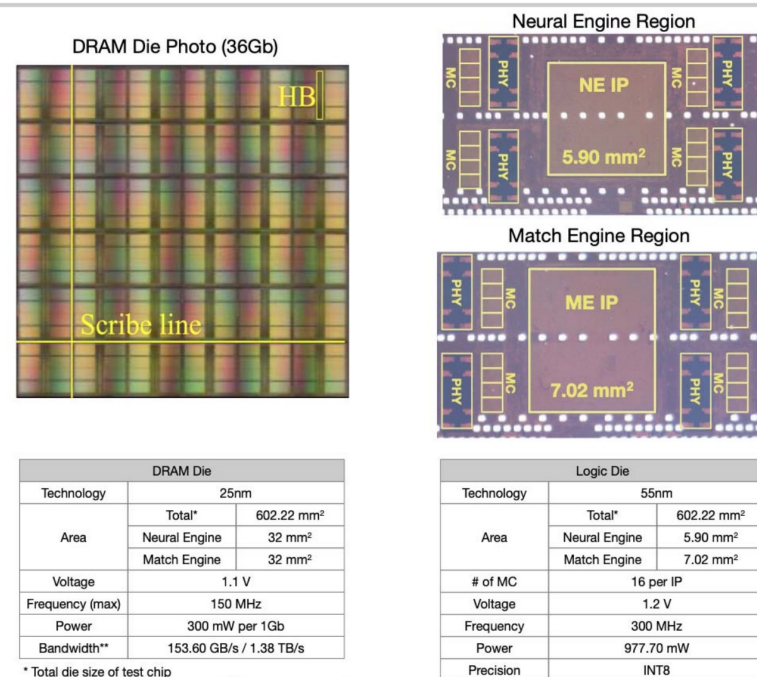


Figure 29.1.7: Die micrographs of DRAM die, NE and ME. Detailed specifications of DRAM die and logic die.

## 29.1 184QPS/W 64Mb/mm<sup>2</sup> 3D Logic-to-DRAM Hybrid Bonding with Process-Near-Memory Engine for Recommendation System

Dimin Niu<sup>1</sup>, Shuangchen Li<sup>1</sup>, Yuhao Wang<sup>1</sup>, Wei Han<sup>1</sup>, Zhe Zhang<sup>2</sup>, Yijin Guan<sup>2</sup>, Tianchan Guan<sup>3</sup>, Fei Sun<sup>1</sup>, Fei Xue<sup>1</sup>, Lide Duan<sup>1</sup>, Yuanwei Fang<sup>1</sup>, Hongzhong Zheng<sup>1</sup>, Xiping Jiang<sup>4</sup>, Song Wang<sup>4</sup>, Fengguo Zuo<sup>4</sup>, Yubing Wang<sup>4</sup>, Bing Yu<sup>4</sup>, Qiwei Ren<sup>4</sup>, Yuan Xie<sup>1</sup>

# SK Hynix CXL Processing Near Memory (2023)

IEEE COMPUTER ARCHITECTURE LETTERS, VOL. 22, NO. 1, JANUARY-JUNE

## Computational CXL-Memory Solution for Accelerating Memory-Intensive Applications

Joonseop Sim<sup>ID</sup>, Soohong Ahn<sup>ID</sup>, Taeyoung Ahn<sup>ID</sup>,  
Seungyong Lee<sup>ID</sup>, Myunghyun Rhee, Jooyoung Kim<sup>ID</sup>,  
Kwangsik Shin, Donguk Moon<sup>ID</sup>,  
Euseok Kim, and Kyoung Park<sup>ID</sup>

**Abstract**—CXL interface is the up-to-date technology that enables effective memory expansion by providing a memory-sharing protocol in configuring heterogeneous devices. However, its limited physical bandwidth can be a significant bottleneck for emerging data-intensive applications. In this work, we propose a novel CXL-based memory disaggregation architecture with a real-world prototype demonstration, which overcomes the bandwidth limitation of the CXL interface using near-data processing. The experimental results demonstrate that our design achieves up to  $1.9\times$  better performance/power efficiency than the existing CPU system.

**Index Terms**—Compute express link (CXL), near-data-processing (NDP)

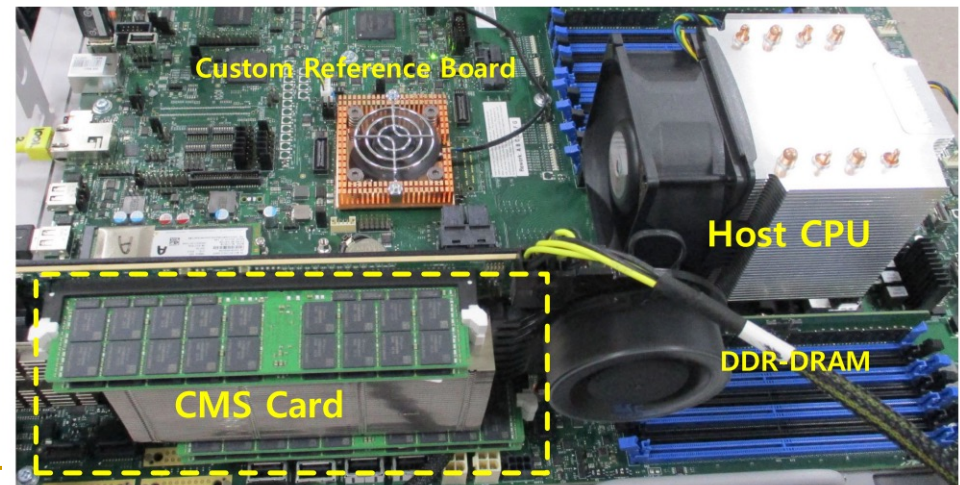


Fig. 6. FPGA prototype of proposed CMS card.

# Samsung CXL Processing Near Memory (2023)

## Samsung Processing in Memory Technology at Hot Chips 2023

By Patrick Kennedy - August 28, 2023



Samsung PIM PNM For Transformer Based AI HC35\_Page\_24

## Processing-in-Memory in the Real World



# 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

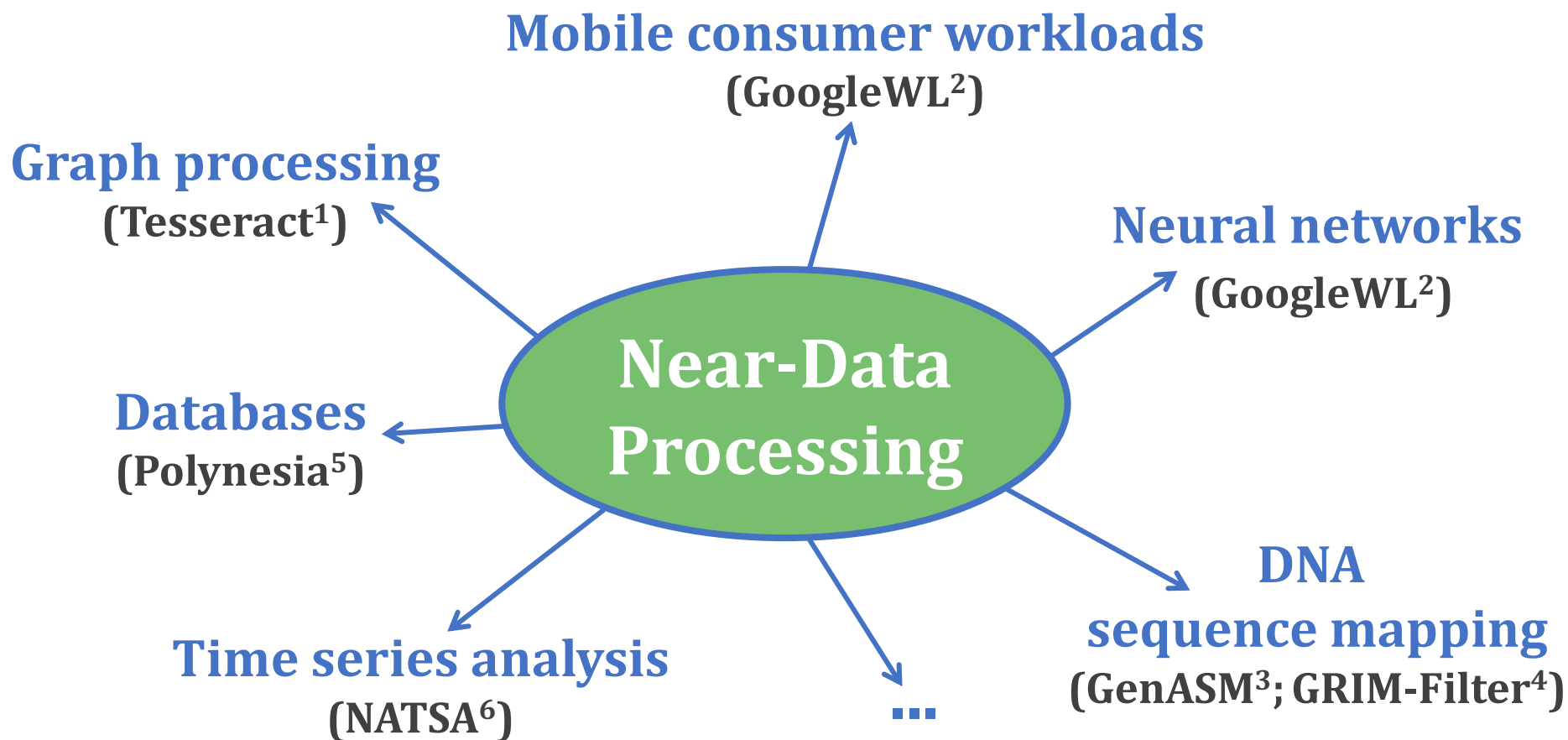
MOHAMMAD SADROSADATI, Institute for Research in Fundamental Sciences (IPM), Iran & ETH Zürich, Switzerland

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>.

# When to Employ Near-Data Processing?



[1] Ahn+, "A Scalable Processing-in-Memory Accelerator for Parallel Graph Processing," ISCA, 2015

[2] Boroumand+, "Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks," ASPLOS, 2018

[3] Cali+, "GenASM: A High-Performance, Low-Power Approximate String Matching Acceleration Framework for Genome Sequence Analysis," MICRO, 2020

[4] Kim+, "GRIM-Filter: Fast Seed Location Filtering in DNA Read Mapping Using Processing-in-Memory Technologies," BMC Genomics, 2018

[5] Boroumand+, "Polynesia: Enabling Effective Hybrid Transactional/Analytical Databases with Specialized Hardware/Software Co-Design," arXiv:2103.00798 [cs.AR], 2021

[6] Fernandez+, "NATSA: A Near-Data Processing Accelerator for Time Series Analysis," ICCD, 2020

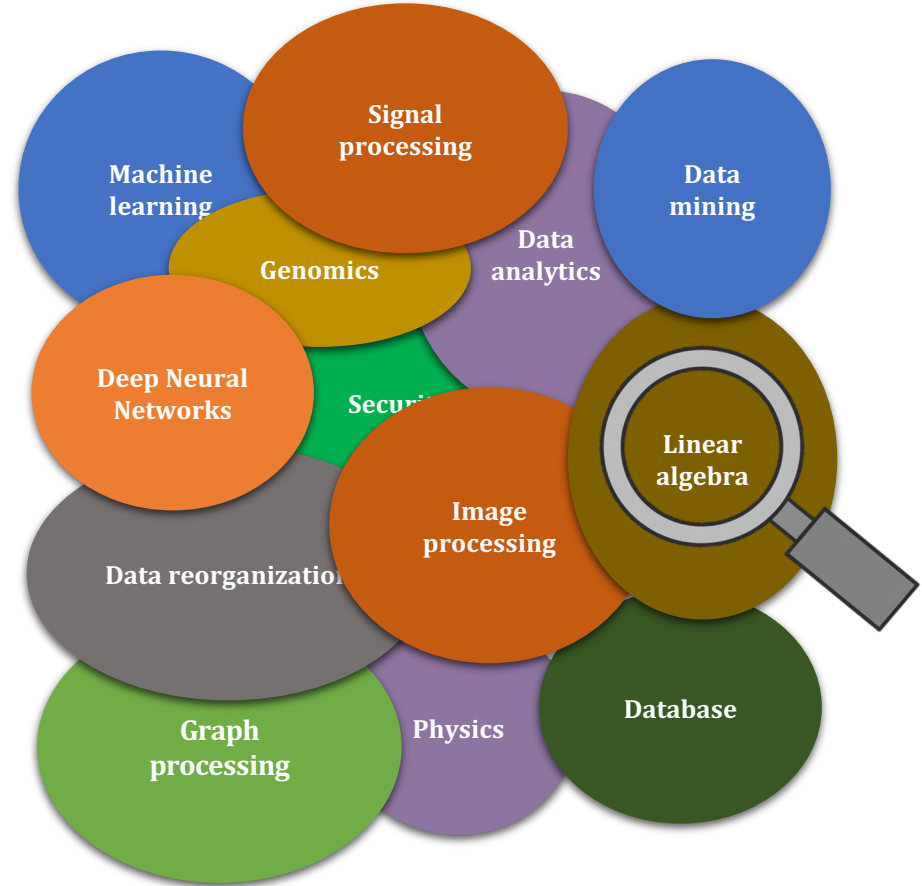
# Step 1: Application Profiling

- We analyze 345 applications from distinct domains:

- Graph Processing
- Deep Neural Networks
- Physics
- High-Performance Computing
- Genomics
- Machine Learning
- Databases
- Data Reorganization
- Image Processing
- Map-Reduce
- Benchmarking
- Linear Algebra

...

**SAFARI**



# Step 3: Memory Bottleneck Analysis

**Six classes of  
data movement bottlenecks:**

each class  $\leftrightarrow$  data movement  
mitigation mechanism

## Memory Bottleneck Class

1a: *DRAM  
Bandwidth*

1b: *DRAM Latency*

1c: *L1/L2  
Cache Capacity*

2a: *L3 Cache  
Contention*

2b: *L1 Cache  
Capacity*

2c: *Compute-Bound*

# 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

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>)

Readme

Releases

No releases published  
[Create a new release](#)

Packages

No packages published  
[Publish your first package](#)

Languages



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

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](#).

DAMOV-SIM

DAMOV  
Benchmarks

SAFARI

# 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

About

DAMOV is a benchmark suite and a

Get DAMOV at:

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

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

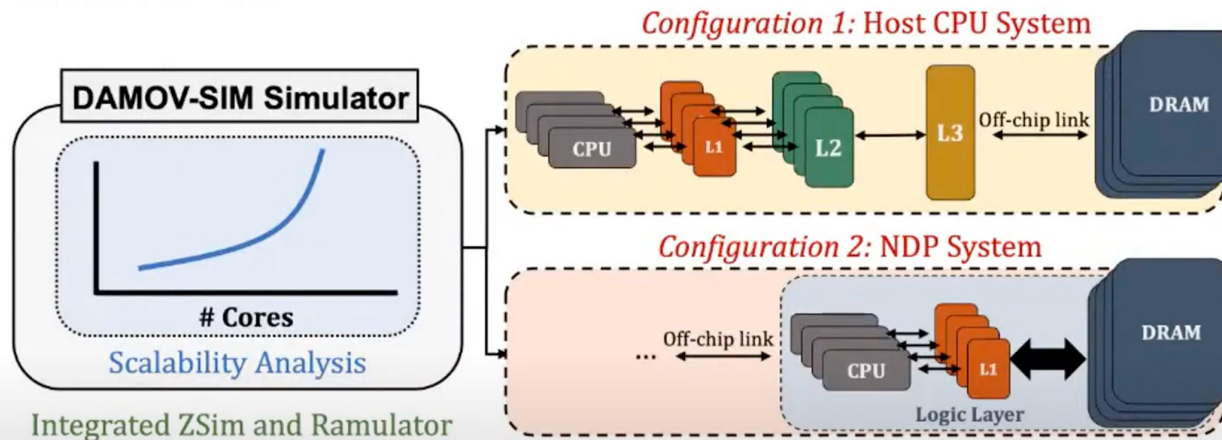
No packages published  
[Publish your first package](#)

### Languages

# More on DAMOV Analysis Methodology & Workloads

## Step 3: Memory Bottleneck Classification (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

SAFARI DAMOV-SIM: <https://github.com/CMU-SAFARI/DAMOV> 30

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

352 views • Streamed live on Jul 22, 2021

18 0 SHARE SAVE ...



Onur Mutlu Lectures  
17.7K subscribers

ANALYTICS

EDIT VIDEO

[https://www.youtube.com/watch?v=GWideVyo0nM&list=PL5Q2soXY2Zi\\_tOTAYm--dYByNPL7JhwR9&index=3](https://www.youtube.com/watch?v=GWideVyo0nM&list=PL5Q2soXY2Zi_tOTAYm--dYByNPL7JhwR9&index=3)



# 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

# Some Other Recent Papers (I)

---

- Jinfan Chen, Juan Gómez-Luna, Izzat El Hajj, Yuxin Guo, and Onur Mutlu,  
**"SimplePIM: A Software Framework for Productive and Efficient Processing in Memory"**  
*Proceedings of the 32nd International Conference on Parallel Architectures and Compilation Techniques (PACT)*, Vienna, Austria, October 2023.
- Harshita Gupta, Mayank Kabra, Juan Gómez-Luna, Konstantinos Kanellopoulos, and Onur Mutlu,  
**"Evaluating Homomorphic Operations on a Real-World Processing-In-Memory System"**  
*Proceedings of the 2023 IEEE International Symposium on Workload Characterization Poster Session (IISWC)*, Ghent, Belgium, October 2023.  
[[arXiv version](#)]  
[[Lightning Talk Slides \(pptx\)](#) ([pdf](#))]  
[[Poster \(pptx\)](#) ([pdf](#))]

# Some Other Recent Papers (II)

---

- Alain Denzler, Rahul Bera, Nastaran Hajinazar, Gagandeep Singh, Geraldo F. Oliveira, Juan Gómez-Luna, and Onur Mutlu,  
["Casper: Accelerating Stencil Computation using Near-cache Processing"](#)  
[IEEE Access](#), 3 March 2023.  
*Preprint in [arXiv](#)*, 28 December 2021.  
[[arXiv preprint](#)]  
[[IEEE Access version](#)]

# Ramulator 2.0 for PIM Systems

---

- Haocong Luo, Yahya Can Tugrul, F. Nisa Bostanci, Ataberk Olgun, A. Giray Yaglikci, and Onur Mutlu,  
**"Ramulator 2.0: A Modern, Modular, and Extensible DRAM Simulator"**  
*Preprint on **arxiv**, August 2023.*  
[[arXiv version](#)]  
[[Ramulator 2.0 Source Code](#)]

## Ramulator 2.0: A Modern, Modular, and Extensible DRAM Simulator

Haocong Luo, Yahya Can Tuğrul, F. Nisa Bostancı, Ataberk Olgun, A. Giray Yağlıkçı, and Onur Mutlu

<https://arxiv.org/pdf/2308.11030.pdf>

# Concluding Remarks

## Fundamentally Energy-Efficient **(Data-Centric)** Computing Architectures

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



# Computing Architectures with Minimal Data Movement

# Concluding Remarks

---

- We must design systems to be **balanced, high-performance, energy-efficient** (all at the same time) → intelligent systems
  - **Data-centric, data-driven, data-aware**
- Enable computation capability inside and close to memory
- This can
  - Lead to **orders-of-magnitude** improvements
  - **Enable new applications & computing platforms**
  - **Enable better understanding of nature**
  - ...
- Future of **truly memory-centric computing** is bright
  - We need to do research & design across the computing stack

**Data-centric**

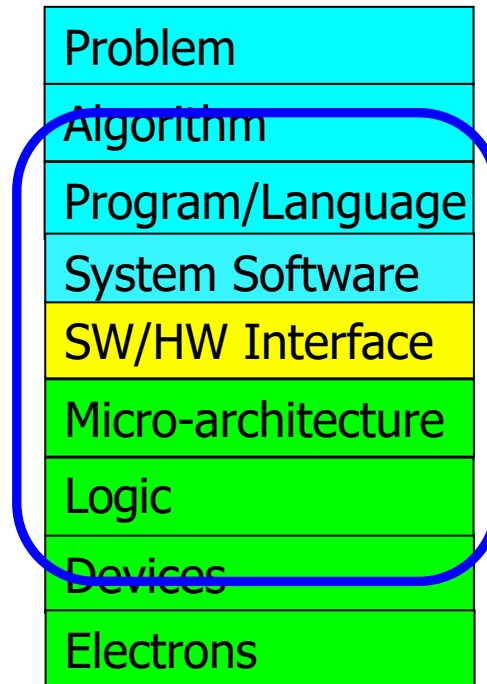
**Data-driven**

**Data-aware**



# We Need to Revisit the Entire Stack

---



**We can get there step by step**

# We Need to Exploit Good Principles

---

- Data-centric system design
- All components intelligent
- Better (cross-layer) communication, better interfaces
- Better-than-worst-case design
- Heterogeneity
- Flexibility, adaptability

**Open minds**

# A Blueprint for Fundamentally Better Architectures

---

- Onur Mutlu,  
**"Intelligent Architectures for Intelligent Computing Systems"**  
*Invited Paper in Proceedings of the Design, Automation, and Test in Europe Conference (**DATE**), Virtual, February 2021.*  
[[Slides \(pptx\)](#)] [[pdf](#)]  
[[IEDM Tutorial Slides \(pptx\)](#)] [[pdf](#)]  
[[Short DATE Talk Video](#) (11 minutes)]  
[[Longer IEDM Tutorial Video](#) (1 hr 51 minutes)]

## Intelligent Architectures for Intelligent Computing Systems

Onur Mutlu  
ETH Zurich  
omutlu@gmail.com



# Funding Acknowledgments

---

- Alibaba, AMD, ASML, Google, Facebook, Hi-Silicon, HP Labs, Huawei, IBM, Intel, Microsoft, Nvidia, Oracle, Qualcomm, Rambus, Samsung, Seagate, VMware, Xilinx
- NSF
- NIH
- GSRC
- SRC
- CyLab
- EFCL
- SNSF

Thank you!

# Acknowledgments

---

# SAFARI

*SAFARI Research Group*

*safari.ethz.ch*

Think BIG, Aim HIGH!

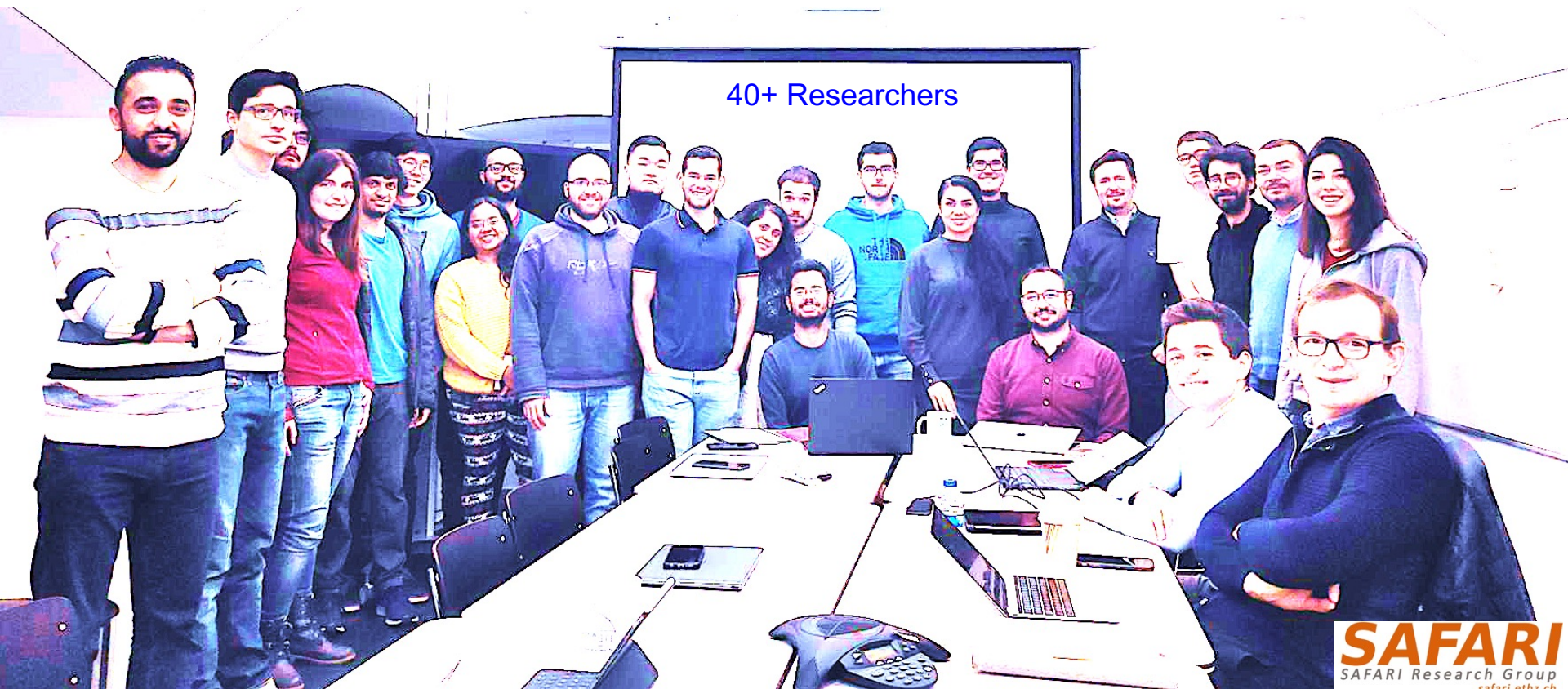
<https://safari.ethz.ch>

---

# Onur Mutlu's SAFARI Research Group

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

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



**SAFARI**  
SAFARI Research Group  
safari.ethz.ch

## Think BIG, Aim HIGH!

**SAFARI**

<https://safari.ethz.ch>

# SAFARI Newsletter December 2021 Edition

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

**SAFARI**  
SAFARI Research Group

*Think Big, Aim High*

**ETH** zürich



View in your browser  
December 2021





# SAFARI Newsletter June 2023 Edition

---

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

**SAFARI**  
SAFARI Research Group

*Think Big, Aim High*

**ETH** zürich



View in your browser  
June 2023



# SAFARI Introduction & Research

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



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



Onur Mutlu Lectures  
32.6K subscribers



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



THINK BIG, AIM HIGH!

SAFARI

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

# SAFARI PhD and Post-Doc Alumni

---

- <https://safari.ethz.ch/safari-alumni/>
- Hasan Hassan (Rivos), **EDAA Outstanding Dissertation Award 2023**; S&P 2020 Best Paper Award, 2020 Pwnie Award, IEEE Micro TP HM 2020
- Christina Giannoula (Univ. of Toronto), **NTUA Best Dissertation Award 2023**
- Minesh Patel (Rutgers, Asst. Prof.), **DSN Carter Award Best Thesis 2022**; ETH Medal 2023; MICRO'20 & DSN'20 Best Paper Awards; ISCA HoF 2021
- Damla Senol Cali (Bionano Genomics), **SRC TECHCON 2019 Best Student Presentation Award**; RECOMB-Seq 2018 Best Poster Award
- Nastaran Hajinazar (Intel)
- Gagandeep Singh (AMD/Xilinx), **FPL 2020 Best Paper Award Finalist**
- Amirali Boroumand (Stanford Univ → Google), **SRC TECHCON 2018 Best Presentation Award**
- Jeremie Kim (Apple), **EDAA Outstanding Dissertation Award 2020**; IEEE Micro Top Picks 2019; ISCA/MICRO HoF 2021
- Nandita Vijaykumar (Univ. of Toronto, Assistant Professor), **ISCA Hall of Fame 2021**
- Kevin Hsieh (Microsoft Research, Senior Researcher)
- Justin Meza (Facebook), **HiPEAC 2015 Best Student Presentation Award**; ICCD 2012 Best Paper Award
- Mohammed Alser (ETH Zurich), **IEEE Turkey Best PhD Thesis Award 2018**
- Yixin Luo (Google), **HPCA 2015 Best Paper Session**
- Kevin Chang (Facebook), **SRC TECHCON 2016 Best Student Presentation Award**
- Rachata Ausavarungnirun (KMUNTB, Assistant Professor), **NOCS 2015 and NOCS 2012 Best Paper Award Finalist**
- Gennady Pekhimenko (Univ. of Toronto, Assistant Professor), **ISCA Hall of Fame 2021**; ASPLOS 2015 SRC Winner
- Vivek Seshadri (Microsoft Research)
- 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)
- Lois Orosa (Galicia Supercomputing Center, Director)
- Jisung Park (POSTECH, Assistant Professor)
- Gagandeep Singh (AMD/Xilinx, Researcher)
- Juan Gomez-Luna (NVIDIA, Researcher), **ISPASS 2023 Best Paper Session**



# Referenced Papers, Talks, Artifacts

---

- All are available at

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

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

<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.

👤 241 followers 📍 ETH Zurich and Carnegie Mellon U... 🔗 <https://safari.ethz.ch/> ✉ [omutlu@gmail.com](mailto:omutlu@gmail.com)

🏠 Overview 📁 Repositories 80 📁 Projects 📁 Packages 👤 People 13

### 📁 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++ ☆ 468 🍴 201

### 📁 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 ☆ 107 🍴 43

### 📁 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++ ☆ 231 🍴 131

### 📁 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](http://users.ece.cmu.edu/~omutlu/pub/dram-row-hammer_isca14.pdf).

● C ☆ 208 🍴 43

### 📁 SoftMC Public

SoftMC is an experimental FPGA-based memory controller design that can be used to develop tests for DDR3 SODIMMs using a C++ based API. The design, the interface, and its capabilities and limitatio...

● Verilog ☆ 105 🍴 26

### 📁 Pythia Public

A customizable hardware prefetching framework using online reinforcement learning as described in the MICRO 2021 paper by Bera et al. (<https://arxiv.org/pdf/2109.12021.pdf>).

● C++ ☆ 91 🍴 28

# Memory-Centric Computing

Onur Mutlu

[omutlu@gmail.com](mailto:omutlu@gmail.com)

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

31 January 2024

UT-Austin ECE Colloquium

**SAFARI**

**ETH** zürich

# More Resources

# Special Research Sessions & Courses

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



The image shows a YouTube video player interface. The video title is "In-Memory Processing ISVLSI 2022 Special Session". Below the title, it says "IEEE Computer Society Annual Symposium on VLSI". The video is from the "Onur Mutlu Lectures" channel, which has 26.9K subscribers. The video has 1,286 views and was premiered on Aug 9, 2022. The video player shows a thumbnail with the text "In-Memory Processing ISVLSI 2022 Special Session" and "IEEE Computer Society Annual Symposium on VLSI". The video is currently at 0:04 / 3:36:35. The video player includes standard YouTube controls like play, pause, volume, and full screen. The video player also includes a small inset video in the top right corner showing a person speaking at a podium.

In-Memory Processing  
ISVLSI 2022 Special Session

IEEE Computer Society Annual Symposium on VLSI

ISVLSI 2022

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

0:04 / 3:36:35 • Dr. Juan Gómez-Luna, "Introduction to the ISVLSI 2022 Special Session on Processing-in-Memory" >

ISVLSI 2022 Special Session on Processing-in-Memory

1,286 views • Premiered Aug 9, 2022

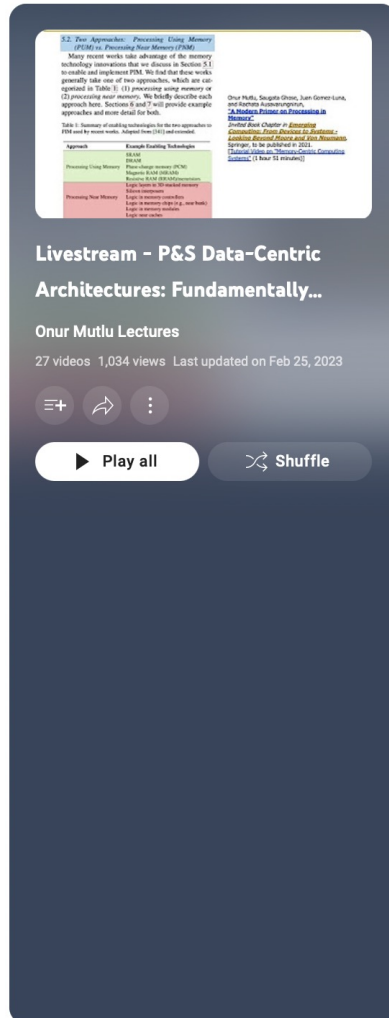
61 DISLIKE SHARE DOWNLOAD CLIP SAVE ...

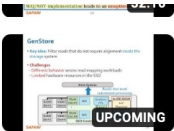
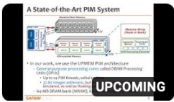
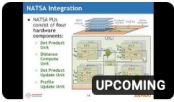
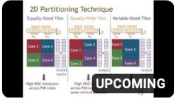
Onur Mutlu Lectures  
26.9K subscribers

ANALYTICS EDIT VIDEO

# Special Research Sessions & Courses (II)

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



- 19  **GenStore: In-Storage Filtering for High-Performance and Energy-Efficient Genome Analysis**  
Onur Mutlu Lectures • Premieres 3/12/23, 7:00 PM
- 20  **Introduction to the ISVLSI 2022 Special Session on Processing-in-Memory**  
Onur Mutlu Lectures • 286 views • 2 days ago
- 21  **Heterogeneous Data-Centric Architectures for Data-Intensive Applications: Case Studies in ML and DB**  
Onur Mutlu Lectures • 2 waiting • Premieres 3/10/23, 7:00 PM
- 22  **Machine Learning Training on a Real Processing-In-Memory System**  
Onur Mutlu Lectures • Premieres 3/14/23, 7:00 PM
- 23  **Exploiting Near-Data Processing to Accelerate Time Series Analysis**  
Onur Mutlu Lectures • Premieres 3/11/23, 7:00 PM
- 24  **PiDRAM: An FPGA-Based Framework for End-To-End Evaluation of Processing-In-DRAM Techniques**  
Onur Mutlu Lectures • Premieres 3/9/23, 7:00 PM
- 25  **The Road to Widely Deploying Processing-In-Memory: Challenges and Opportunities**  
Onur Mutlu Lectures • 399 views • 1 day ago
- 26  **SparseP: Efficient Sparse Matrix Vector Multiplication on Real Processing-In-Memory Architectures**  
Onur Mutlu Lectures • 1 waiting • Premieres 3/13/23, 7:00 PM
- 27  **HPCA 2023 Tutorial: Real-World Processing-in-Memory Architectures**  
Onur Mutlu Lectures • 1.6K views • Streamed 10 days ago

# 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\\_5EFgo&list=PL5Q2soXY2Zi-Mnk1PxjEIG32HAGILkTOF](https://www.youtube.com/watch?v=4yfkM_5EFgo&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>

Computer Architecture - Fall 2021

Recent Changes Media Manager Sitemap

Trace: readings start schedule

Home

Announcements

Materials

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

Resources

- 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

Livestream Lecture Playlist

2.560-DPU Processing in Memory System

Watch on YouTube

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

Recorded Lecture Playlist

TESLA Full Self-Driving Computer (2020)

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

Watch on YouTube

### Fall 2021 Lectures & Schedule

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



# DDCA (Spring 2022)

## Spring 2022 Edition:

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

## Spring 2021 Edition:

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

## Youtube Livestream (Spring 2022):

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

## Youtube Livestream (Spring 2021):

- [https://www.youtube.com/watch?v=LbC0EZY8yw4&list=PL5Q2soXY2Zi\\_uej3aY39YB5pfW4SJ7LIN](https://www.youtube.com/watch?v=LbC0EZY8yw4&list=PL5Q2soXY2Zi_uej3aY39YB5pfW4SJ7LIN)

## Bachelor's course

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

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



Trace: - schedule

Home

Announcements

Materials

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

Resources

- 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

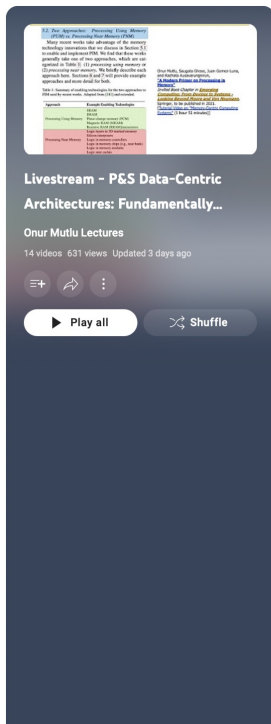
Recorded Lecture Playlist

## Spring 2021 Lectures/Schedule

| Week | Date       | Livestream   | Lecture                                                           | Readings                           | Lab | HW |
|------|------------|--------------|-------------------------------------------------------------------|------------------------------------|-----|----|
| W1   | 25.02 Thu. | YouTube Live | L1: Introduction and Basics<br>G2a (PDF) G2a (PPT)                | Required<br>Suggested<br>Mentioned |     |    |
|      | 26.02 Fri. | YouTube Live | L2a: Tradeoffs, Metrics, Mindset<br>G2a (PDF) G2a (PPT)           | Required                           |     |    |
|      |            |              | L2b: Mysteries in Computer Architecture<br>G2a (PDF) G2a (PPT)    | Required<br>Mentioned              |     |    |
| W2   | 04.03 Thu. | YouTube Live | L3a: Mysteries in Computer Architecture II<br>G2a (PDF) G2a (PPT) | Required<br>Suggested<br>Mentioned |     |    |

# Processing-in-Memory Course (Fall 2022)

- Short weekly lectures
- Hands-on projects



1

**PIM Course: Lecture 1: Data-Centric Architectures: Improving Performance & Energy - Fall 2022**  
Onur Mutlu Lectures • 1K views • 3 months ago

2

**PIM Course: Lecture 2: How to Evaluate Data Movement Bottlenecks - Fall 2022**  
Onur Mutlu Lectures • 678 views • 2 months ago

3

**PIM Course: Lecture 3: Real-world PIM: UPMEM PIM - Fall 2022**  
Onur Mutlu Lectures • 455 views • 2 months ago

4

**PIM Course: Lecture 4: Real-world PIM: Microbenchmarking of UPMEM PIM - Fall 2022**  
Onur Mutlu Lectures • 275 views • 2 months ago

5

**PIM Course: Lecture 5: Real-world PIM: Samsung HBM-PIM - Fall 2022**  
Onur Mutlu Lectures • 725 views • 2 months ago

6

**PIM Course: Lecture 6: Real-world PIM: SK Hynix AiM - Fall 2022**  
Onur Mutlu Lectures • 1K views • 2 months ago

7

**PIM Course: Lecture 7: Real-world PIM: Samsung AxDIMM - Fall 2022**  
Onur Mutlu Lectures • 767 views • 1 month ago

8

**PIM Course: Lecture 8: Real-world PIM: Alibaba HB-PNM - Fall 2022**  
Onur Mutlu Lectures • 383 views • 1 month ago

9

**PIM Course: Lecture 9: Programming PIM Architectures - Fall 2022**  
Onur Mutlu Lectures • 367 views • 1 month ago

design

**SAFARI Project & Seminars Courses (Fall 2022)**

Trace: • heterogeneous\_systems • processing\_in\_memory

Home

Courses

- SoftMC
- Ramulator
- Accelerating Genomics
- Mobile Genomics
- **Processing-in-Memory**
- Heterogeneous Systems
- Modern SSDs

processing\_in\_memory

**Table of Contents**

- Data-Centric Architectures: Fundamentally Improving Performance and Energy (227-0085-37L)
  - Course Description
  - Mentors
  - Lecture Video Playlist on YouTube
  - Spring 2022 Meetings/Schedule
  - Past Lecture Video Playlists on YouTube
  - Learning Materials
  - Assignments

**Data-Centric Architectures: Fundamentally Improving Performance and Energy (227-0085-37L)**

Edit

**Course Description**

Data movement between the memory units and the compute units of current computing systems is a major performance and energy bottleneck. From large-scale servers to mobile devices, data movement costs dominate computation costs in terms of both performance and energy consumption. For example, data movement between the main memory and the processing cores accounts for 62% of the total system energy in consumer applications. As a result, the data movement bottleneck is a huge burden that greatly limits the energy efficiency and performance of modern computing systems. This phenomenon is an undesired effect of the dichotomy between memory and the processor, which leads to the data movement bottleneck.

Many modern and important workloads such as machine learning, computational biology, graph processing, databases, video analytics, and real-time data analytics suffer greatly from the data movement bottleneck. These workloads are exemplified by irregular memory accesses, relatively low data reuse, low cache line utilization, low arithmetic intensity (i.e., ratio of operations per accessed byte), and large datasets that greatly exceed the main memory size. The computation in these workloads cannot usually compensate for the data movement costs. In order to alleviate this data movement bottleneck, we need a paradigm shift from the traditional processor-centric design, where all computation takes place in the compute units, to a more data-centric design where processing elements are placed closer to or inside where the data resides. This paradigm of computing is known as Processing-in-Memory (PIM).

This is your perfect P&S if you want to become familiar with the main PIM technologies, which represent “the next big thing” in Computer Architecture. You will work hands-on with the first real-world PIM architecture, will explore different PIM architecture designs for important workloads, and will develop tools to enable research of future PIM systems. Projects in this course span software and hardware as well as the software/hardware interface. You can potentially work on developing and optimizing new workloads for the first real-world PIM hardware or explore new PIM designs in simulators, or do something else that can forward our understanding of the PIM paradigm.

[https://safari.ethz.ch/projects\\_and\\_seminars/fall2022/doku.php?id=processing\\_in\\_memory](https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=processing_in_memory)

<https://youtube.com/playlist?list=PL5Q2soXY2Zi8KzG2CQYRNQOVD0GOBrnKy>

# PIM Course (Fall 2022)

## Fall 2022 Edition:

- [https://safari.ethz.ch/projects\\_and\\_seminars/fall2022/doku.php?id=processing\\_in\\_memory](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](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):

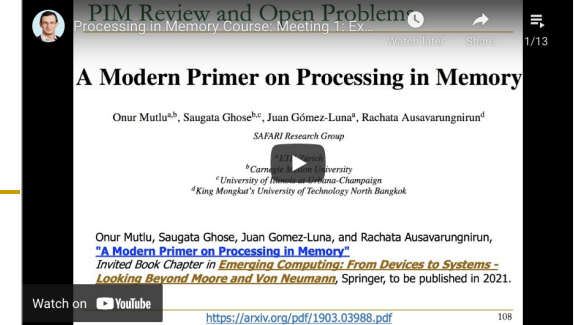
- <https://www.youtube.com/watch?v=9e4Chnwdovo&list=PL5Q2soXY2Zi-841fUYYUK9EsXKhQKRPyX>

## 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>

**SAFARI**

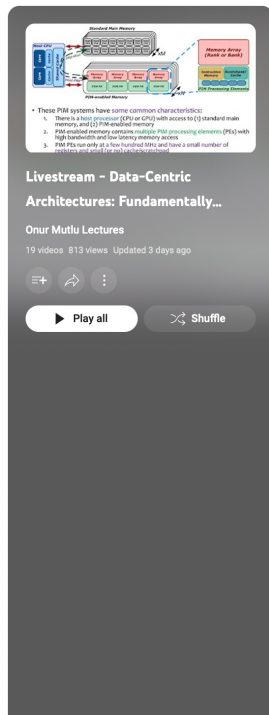


### Spring 2022 Meetings/Schedule

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

# Processing-in-Memory Course (Spring 2023)

- Short weekly lectures
- Hands-on projects



- PIM Course: Lecture 1: Data-Centric Architectures: Improving Performance & Energy (Spring 2023)**  
Onur Mutlu Lectures • 1.1K views • Streamed 3 months ago  
1:14:16
- PIM Course: Lecture 2: How to Evaluate Data Movement Bottlenecks (Spring 2023)**  
Onur Mutlu Lectures • 332 views • 2 months ago  
16:37
- ASPLOS 2023 Tutorial: Real-world Processing-in-Memory Systems for Modern Workloads**  
Onur Mutlu Lectures • 1.5K views • Streamed 2 months ago  
6:27:39
- PIM Course: Lecture 3: Real-world PIM: UPMEM PIM (Spring 2023)**  
Onur Mutlu Lectures • 411 views • 2 months ago  
19:43
- PIM Course: Lecture 4: Real-world PIM: Microbenchmarking of UPMEM PIM (Spring 2023)**  
Onur Mutlu Lectures • 188 views • 2 months ago  
24:10
- Análisis Experimental de una Arquitectura PIM - Juan Gómez Luna - Lecture in Spanish @ U. de Córdoba**  
Onur Mutlu Lectures • 169 views • 2 months ago  
2:27:12
- PIM Course: Lecture 5: Real-world PIM: Samsung HBM-PIM (Spring 2023)**  
Onur Mutlu Lectures • 483 views • 2 months ago  
24:08
- PIM Course: Lecture 6: Real-world PIM: SK Hynix AIM (Spring 2023)**  
Onur Mutlu Lectures • 573 views • 1 month ago  
35:50
- PIM Course: Lecture 7: Real-world PIM: Samsung AxDIMM (Spring 2023)**  
Onur Mutlu Lectures • 325 views • 1 month ago  
21:32

[https://www.youtube.com/playlist?list=PL5Q2soXY2zi\\_EObuoAZVSq\\_o6UySWQHvz](https://www.youtube.com/playlist?list=PL5Q2soXY2zi_EObuoAZVSq_o6UySWQHvz)

**SAFARI Project & Seminars Courses**  
(Spring 2023)

Search

Recent Changes Media Manager Sitemap

Trace: • heterogeneous\_systems • processing\_in\_memory

Home

Courses

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

processing\_in\_memory

**Data-Centric Architectures: Fundamentally Improving Performance and Energy (227-0085-37L)**

Course Description

Data movement between the memory units and the compute units of current computing systems is a major performance and energy bottleneck. From large-scale servers to mobile devices, data movement costs dominate computation costs in terms of both performance and energy consumption. For example, data movement between the main memory and the processing cores accounts for 62% of the total system energy in consumer applications. As a result, the data movement bottleneck is a huge burden that greatly limits the energy efficiency and performance of modern computing systems. This phenomenon is an undesired effect of the dichotomy between memory and the processor, which leads to the data movement bottleneck.

Many modern and important workloads such as machine learning, computational biology, graph processing, databases, video analytics, and real-time data analytics suffer greatly from the data movement bottleneck. These workloads are exemplified by irregular memory accesses, relatively low data reuse, low cache line utilization, low arithmetic intensity (i.e., ratio of operations per accessed byte), and large datasets that greatly exceed the main memory size. The computation in these workloads cannot usually compensate for the data movement costs. In order to alleviate this data movement bottleneck, we need a paradigm shift from the traditional processor-centric design, where all computation takes place in the compute units, to a more data-centric design where processing elements are placed closer to or inside where the data resides. This paradigm of computing is known as Processing-in-Memory (PIM).

This is your perfect P&S if you want to become familiar with the main PIM technologies, which represent “the next big thing” in Computer Architecture. You will work hands-on with the first real-world PIM architecture, will explore different PIM architecture designs for important workloads, and will develop tools to enable research of future PIM systems. Projects in this course span software and hardware as well as the software/hardware interface. You can potentially work on developing and optimizing new workloads for the first real-world PIM hardware or explore new PIM designs in simulators, or do something else that can forward our understanding of the PIM paradigm.

Table of Contents

- Data-Centric Architectures: Fundamentally Improving Performance and Energy (227-0085-37L)
- Course Description
- Mentors
- Lecture Video Playlist on YouTube
- Spring 2023 Meetings/Schedule
- Past Lecture Video Playlists on YouTube
- Learning Materials
- Assignments

Prerequisites of the course:

- Digital Design and Computer Architecture (or equivalent course).
- Familiarity with C/C++ programming.
- Interest in future computer architectures and computing paradigms.
- Interest in discovering why things do or do not work and solving problems
- Interest in making systems efficient and usable

[https://safari.ethz.ch/projects\\_and\\_seminars/spring2023/doku.php?id=processing\\_in\\_memory](https://safari.ethz.ch/projects_and_seminars/spring2023/doku.php?id=processing_in_memory)



# Real PIM Tutorials [MICRO'23, ISCA'23, ASPLOS'23, HPCA'23]

- June, March, Feb : Lectures + Hands-on labs + Invited talks



## ISCA 2023 Real-World PIM Tutorial

Search

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

Trace: • [start](#)

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

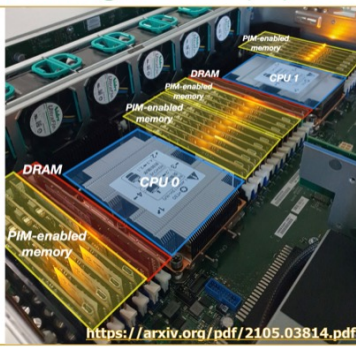
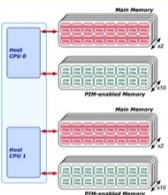
#### Tutorial Description

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.

#### Table of Contents

- [Real-world Processing-in-Memory Systems for Modern Workloads](#)
- [Tutorial Description](#)
- [Organizers](#)
- [Agenda \(June 18, 2023\)](#)
- [Lectures \(tentative\)](#)
- [Hands-on Labs \(tentative\)](#)
- [Learning Materials](#)

<https://events.safari.ethz.ch/isca-pim-tutorial/>

# Real PIM Tutorial [ISCA 2023]

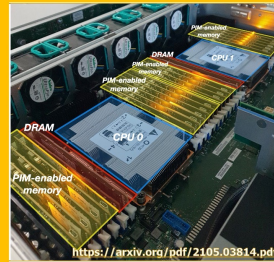
## ■ June 18: Lectures + Hands-on labs + Invited talks

### 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



International Symposium on Computer Architecture (ISCA)

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

<https://www.youtube.com/live/GIb5EgSrWk0?feature=share>

Room: Magnolia 16  
Marriott World Center Orlando  
Orlando, FL, USA  
July 18th, 2023

**SAFARI** zoom

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

Onur Mutlu Lectures  
33.9K subscribers

Subscribed

57

Share

Download

Clip

...

1,687 views Streamed live on Jun 18, 2023 Livestream - Data-Centric Architectures: Fundamentally Improving Performance and Energy (Spring 2023)

[https://www.youtube.com/  
live/GIb5EgSrWk0](https://www.youtube.com/live/GIb5EgSrWk0)

[https://events.safari.ethz.ch/  
isca-pim-tutorial/](https://events.safari.ethz.ch/isca-pim-tutorial/)

### Tutorial Materials

| Time            | Speaker                             | Title                                                                                                               | Materials                                                             |
|-----------------|-------------------------------------|---------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------|
| 8:55am-9:00am   | Dr. Juan Gómez Luna                 | Welcome & Agenda                                                                                                    | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 9:00am-10:20am  | Prof. Onur Mutlu                    | Memory-Centric Computing                                                                                            | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 10:20am-11:00am | Dr. Juan Gómez Luna                 | Processing-Near-Memory: Real PNM Architectures / Programming General-purpose PIM                                    | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 11:20am-11:50am | Prof. Izzat El Hajj                 | High-throughput Sequence Alignment using Real Processing-in-Memory Systems                                          | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 11:50am-12:30pm | Dr. Christina Giannoula             | SparseP: Towards Efficient Sparse Matrix Vector Multiplication for Real Processing-In-Memory Systems                | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 2:00pm-2:45pm   | Dr. Sukhan Lee                      | Introducing Real-world HBM-PIM Powered System for Memory-bound Applications                                         | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 2:45pm-3:30pm   | Dr. Juan Gómez Luna / Ataberk Olgun | Processing-Using-Memory: Exploiting the Analog Operational Properties of Memory Components / PUM Prototypes: PiDRAM | <a href="#">(PDF)</a> <a href="#">(PPT)</a> <a href="#">(PPT)</a>     |
| 4:00pm-4:40pm   | Dr. Juan Gómez Luna                 | Accelerating Modern Workloads on a General-purpose PIM System                                                       | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 4:40pm-5:20pm   | Dr. Juan Gómez Luna                 | Adoption Issues: How to Enable PIM?                                                                                 | <a href="#">(PDF)</a> <a href="#">(PPT)</a>                           |
| 5:20pm-5:30pm   | Dr. Juan Gómez Luna                 | Hands-on Lab: Programming and Understanding a Real Processing-in-Memory Architecture                                | <a href="#">(Handout)</a> <a href="#">(PDF)</a> <a href="#">(PPT)</a> |

# Real PIM Tutorial [ASPLOS 2023]

## ■ March 26: Lectures + Hands-on labs + Invited talks

ASPLOS 2023 Real-World PIM Tutorial

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

Important note about registration

**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) assess realistic strategies for PIM kernels, and (3)

### Tutorial Materials

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

ASPLOS 2023 Tutorial

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

**Accelerating Modern Workloads on a General-purpose PIM System**

Dr. Juan Gómez Luna  
Professor Onur Mutlu

ETH Zürich SAFARI

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

Sunday, March 26, 2023

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

Onur Mutlu Lectures  
32.1K subscribers

Subscribed

33

Share

Clip

Save

...

views Streamed 7 days ago Livestream - Data-Centric Architectures: Fundamentally Improving Performance and Energy (Spring 2023)

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

<https://events.safari.ethz.ch/asplos-2023/>

<https://www.youtube.com/watch?v=oYCaLcT0Kmo>

<https://events.safari.ethz.ch/asplos-pim-tutorial/>



# Real PIM Tutorial [HPCA 2023]

## ■ February 26: Lectures + Hands-on labs + Invited Talks

**Real-world Processing-in-Memory Architectures**

**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, Mythic) are already commercializing real PIM hardware, each with its own design approach and target applications. Several major vendors (e.g., Samsung, SK Hynix, Allbaba) have presented real PIM chip prototypes in the last two years.

**2,560-DPU Processing-in-Memory System**

Most of these architectures have in common that they place compute units near the memory arrays. But, there is more to come: Academia and Industry are actively exploring other types of PIM by, e.g., exploiting the analog operation of DRAM, SRAM, flash memory and emerging non-volatile memories.

PIM can provide large improvements in both performance and energy consumption, thereby enabling a commercially viable way of dealing with huge amounts of data that is bottlenecking our computing systems. Yet, it is critical to examine and research adoption issues of PIM using especially learnings from real PIM systems that are available today.

This tutorial focuses on the latest advances in PIM technology. 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 using real PIM systems, and (4) shed light on how to enable the adoption of PIM in future computing systems.

| Time            | Speaker                 | Title                                                                                                     | Materials                                                       |
|-----------------|-------------------------|-----------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------|
| 8:00am-8:40am   | Prof. Onur Mutlu        | Memory-Centric Computing                                                                                  | <a href="#">PDF</a> <a href="#">PPT</a>                         |
| 8:40am-10:00am  | Dr. Juan Gómez Luna     | Processing-Near-Memory: Real PNM Architectures Programming General-purpose PIM                            | <a href="#">PDF</a> <a href="#">PPT</a>                         |
| 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 | <a href="#">PDF</a> <a href="#">PPT</a>                         |
| 1:30pm-2:10pm   | Dr. Juan Gómez Luna     | Processing-Using-Memory: Exploiting the Analog Operational Properties of Memory Components                | <a href="#">PDF</a> <a href="#">PPT</a>                         |
| 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?                                                          | <a href="#">PDF</a> <a href="#">PPT</a>                         |
| 3:40pm-5:40pm   | Dr. Juan Gómez Luna     | Hands-on Lab: Programming and Understanding a Real Processing-in-Memory Architecture                      | <a href="#">Handout</a> <a href="#">PDF</a> <a href="#">PPT</a> |

**Goal: Processing Inside Memory**

Processor Core

Memory

Database

Graphs

Media

Query

Results

Interconnect

■ Many questions ... How do we design the:

- compute-capable memory & controllers?
- processors & communication units?
- software & hardware interfaces?
- system software, compilers, languages?
- algorithms & theoretical foundations?

HPCA 2023 Tutorial: Real-World Processing-in-Memory Architectures

Onur Mutlu Lectures

32.1K subscribers

50

Share

Clip

Save

1.8K views · Streamed 1 month ago · Livestream - P&S Data-Centric Architectures: Fundamentally Improving Performance and Energy (Fall 2022)

HPCA 2023 Tutorial: Real-World Processing-in-Memory Architectures

<https://events.safari.ethz.ch/real-pi...>

<https://www.youtube.com/watch?v=f5-nT1tbz5w>

<https://events.safari.ethz.ch/real-pim-tutorial/>

- **October 29:** Lectures + Hands-on labs + Invited talks

[illegible]

<https://events.safari.ethz.ch/micro-pim-tutorial>

## 300

# SSD Course (Spring 2023)

## Spring 2023 Edition:

- [https://safari.ethz.ch/projects\\_and\\_seminars/spring2023/doku.php?id=modern\\_ssd](https://safari.ethz.ch/projects_and_seminars/spring2023/doku.php?id=modern_ssd)

## Fall 2022 Edition:

- [https://safari.ethz.ch/projects\\_and\\_seminars/fall2022/doku.php?id=modern\\_ssd](https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=modern_ssd)

## Youtube Livestream (Spring 2023):

- [https://www.youtube.com/watch?v=4VTwOMmsnJY&list=PL5Q2soXY2Zi\\_8qOM5Icpp8hB2SHtm4z57&pp=iAQB](https://www.youtube.com/watch?v=4VTwOMmsnJY&list=PL5Q2soXY2Zi_8qOM5Icpp8hB2SHtm4z57&pp=iAQB)

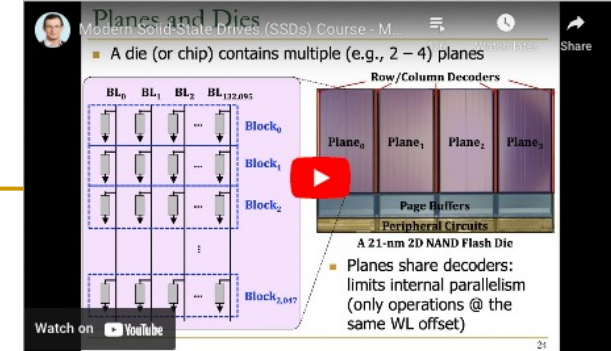
## Youtube Livestream (Fall 2022):

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

## Project course

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

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



## Fall 2022 Meetings/Schedule

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



# Genomics Course (Fall 2022)

## ■ Fall 2022 Edition:

- [https://safari.ethz.ch/projects\\_and\\_seminars/fall2022/doku.php?id=bioinformatics](https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=bioinformatics)

## ■ Spring 2022 Edition:

- [https://safari.ethz.ch/projects\\_and\\_seminars/spring2022/doku.php?id=bioinformatics](https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=bioinformatics)

## ■ Youtube Livestream (Fall 2022):

- [https://www.youtube.com/watch?v=nA41964-9r8&list=PL5Q2soXY2Zi8tFIQvdxOdizD\\_EhVAMVQV](https://www.youtube.com/watch?v=nA41964-9r8&list=PL5Q2soXY2Zi8tFIQvdxOdizD_EhVAMVQV)

## ■ Youtube Livestream (Spring 2022):

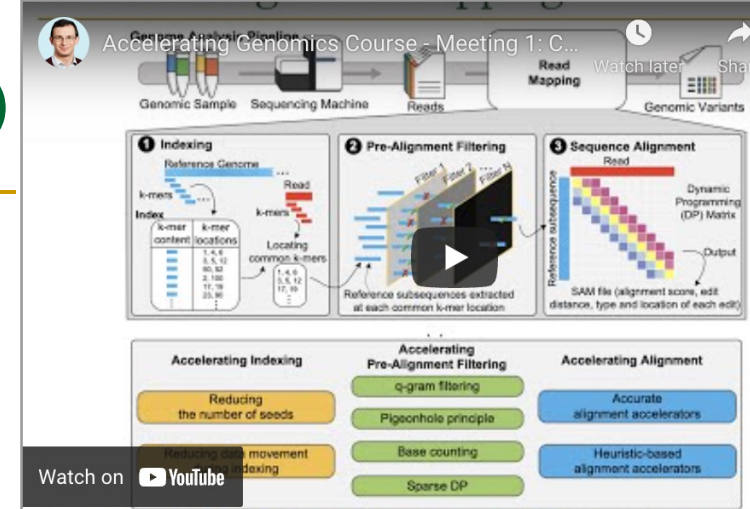
- [https://www.youtube.com/watch?v=DEL\\_5A\\_Y3TI&list=PL5Q2soXY2Zi8NrPDgOR1yRU\\_Cxxjw-u18](https://www.youtube.com/watch?v=DEL_5A_Y3TI&list=PL5Q2soXY2Zi8NrPDgOR1yRU_Cxxjw-u18)

## ■ Project course

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

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

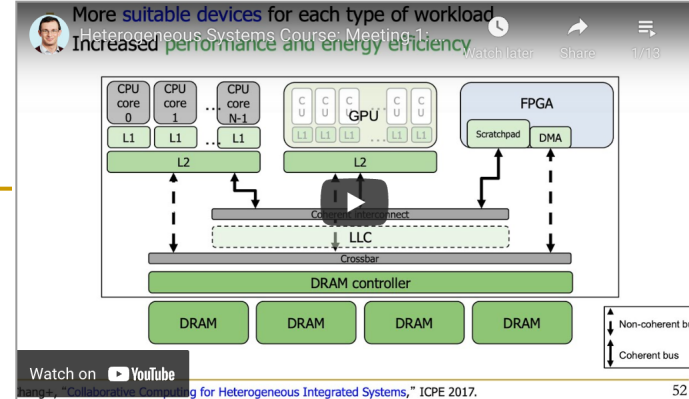
**SAFARI**



## 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<br>(PDF) (PPT) | Required Materials<br>Recommended Materials |
| W2   | 18.3 Fri.  | YouTube Live     | M2: Introduction to Sequencing<br>(PDF) (PPT)                                        |                                             |
| W3   | 25.3 Fri.  | YouTube Premiere | M3: Read Mapping<br>(PDF) (PPT)                                                      |                                             |
| W4   | 01.04 Fri. | YouTube Premiere | M4: GateKeeper<br>(PDF) (PPT)                                                        |                                             |
| W5   | 08.04 Fri. | YouTube Premiere | M5: MAGNET & Shouji<br>(PDF) (PPT)                                                   |                                             |
| W6   | 15.4 Fri.  | YouTube Premiere | M6: SneakySnake<br>(PDF) (PPT)                                                       |                                             |
| W7   | 29.4 Fri.  | YouTube Premiere | M7: GenStore<br>(PDF) (PPT)                                                          |                                             |
| W8   | 06.05 Fri. | YouTube Premiere | M8: GRIM-Filter<br>(PDF) (PPT)                                                       |                                             |
| W9   | 13.05 Fri. | YouTube Premiere | M9: Genome Assembly<br>(PDF) (PPT)                                                   |                                             |
| W10  | 20.05 Fri. | YouTube Live     | M10: Genomic Data Sharing Under Differential Privacy<br>(PDF) (PPT)                  |                                             |
| W11  | 10.06 Fri. | YouTube Premiere | M11: Accelerating Genome Sequence Analysis<br>(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=PL5Q2soXY2Zi9XrgXR38IM\_FTjmY6h7Gzm

## 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>

## Spring 2022 Meetings/Schedule

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

# HW/SW Co-Design (Spring 2022)

## Spring 2022 Edition:

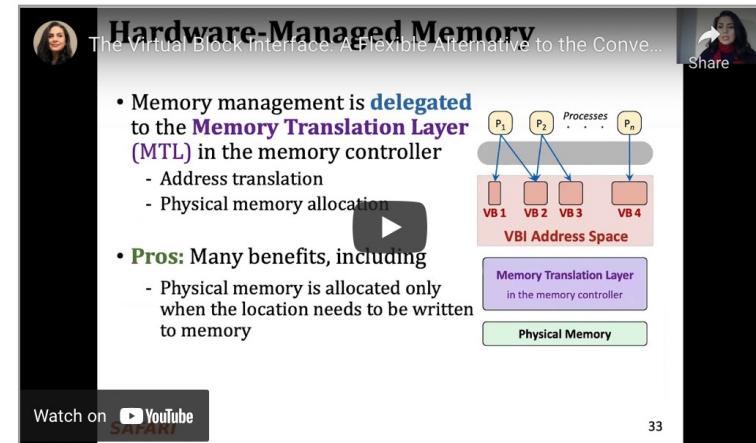
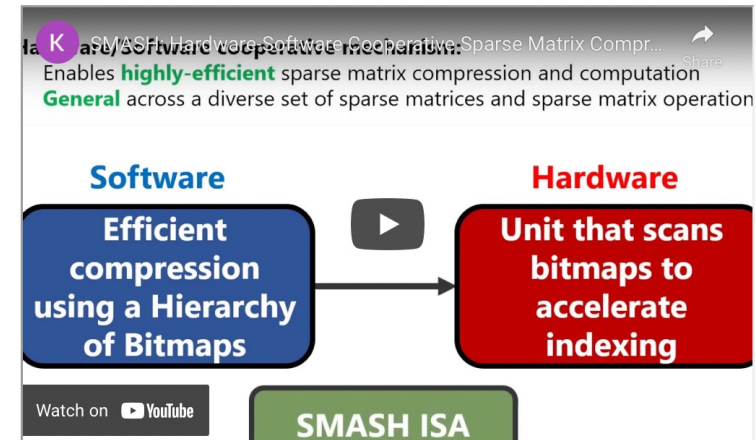
- https://safari.ethz.ch/projects\_and\_seminars/spring2022/doku.php?id=hw\_sw\_co\_design

## 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



## 2022 Meetings/Schedule (Tentative)

| Week | Date  | Livestream                   | Meeting                                                                       | Materials | Assignments |
|------|-------|------------------------------|-------------------------------------------------------------------------------|-----------|-------------|
| W0   | 16.03 | <a href="#">YouTube Live</a> | <a href="#">Intro to HW/SW Co-Design</a> (PPTX) <a href="#">Q&amp;A</a> (PDF) | Required  | HW 0 Out    |
| W1   | 23.03 |                              | <a href="#">Project selection</a>                                             | Required  |             |
| W2   | 30.03 | <a href="#">YouTube Live</a> | <a href="#">Virtual Memory (I)</a> (PPTX) <a href="#">Q&amp;A</a> (PDF)       |           |             |
| W3   | 13.04 | <a href="#">YouTube Live</a> | <a href="#">Virtual Memory (II)</a> (PPTX) <a href="#">Q&amp;A</a> (PDF)      |           |             |

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

# RowHammer & DRAM Exploration (Fall 2022)

## Fall 2022 Edition:

- ❑ [https://safari.ethz.ch/projects\\_and\\_seminars/fall2022/doku.php?id=softmc](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](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](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



### 2022 Meetings/Schedule (Tentative)

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



# Exploration of Emerging Memory Systems (Fall 2022)

## Fall 2022 Edition:

- ❑ [https://safari.ethz.ch/projects\\_and\\_seminars/fall2022/doku.php?id=ramulator](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](https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=ramulator)

## Youtube Livestream (Spring 2022):

- ❑ [https://www.youtube.com/watch?v=aM-lIXRQd3s&list=PL5Q2soXY2Zi\\_TlmlGw\\_Z8hBo2925ZApgV](https://www.youtube.com/watch?v=aM-lIXRQd3s&list=PL5Q2soXY2Zi_TlmlGw_Z8hBo2925ZApgV)

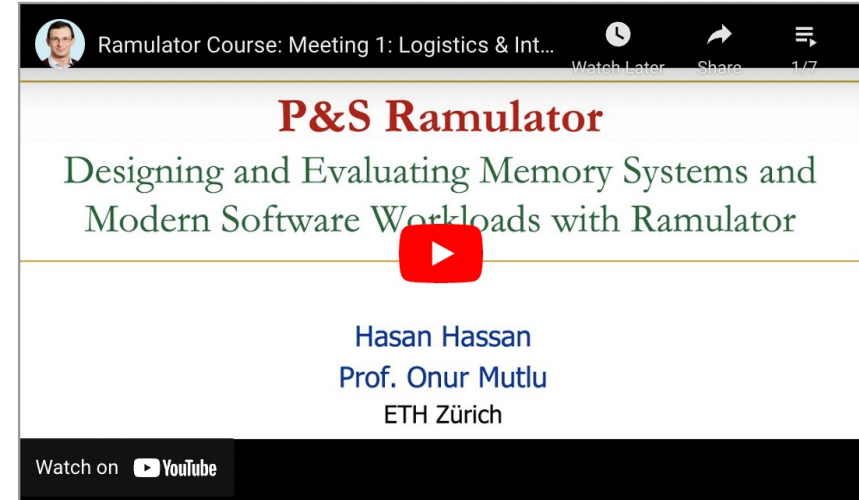
## Bachelor's course

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

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

### Lecture Video Playlist on YouTube

Lecture Playlist



### 2022 Meetings/Schedule (Tentative)

| Week | Date       | Livestream                                    | Meeting                                                                                                                              | Learning Materials | Assignments |
|------|------------|-----------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------|--------------------|-------------|
| W1   | 09.03 Wed. | <a href="#">YouTube</a> <a href="#">Video</a> | <b>M1: Logistics &amp; Intro to Simulating Memory Systems Using Ramulator</b><br><a href="#">PDF</a> (PDF) <a href="#">PPT</a> (PPT) |                    | HW0         |
| W2   | 16.03 Fri. | <a href="#">YouTube</a> <a href="#">Video</a> | <b>M2: Tutorial on Using Ramulator</b><br><a href="#">PDF</a> (PDF) <a href="#">PPT</a> (PPT)                                        |                    |             |
| W3   | 25.02 Fri. | <a href="#">YouTube</a> <a href="#">Video</a> | <b>M3: BlockHammer</b><br><a href="#">PDF</a> (PDF) <a href="#">PPT</a> (PPT)                                                        |                    |             |
| W4   | 01.04 Fri. | <a href="#">YouTube</a> <a href="#">Video</a> | <b>M4: CLR-DRAM</b><br><a href="#">PDF</a> (PDF) <a href="#">PPT</a> (PPT)                                                           |                    |             |
| W5   | 08.04 Fri. | <a href="#">YouTube</a> <a href="#">Video</a> | <b>M5: SIMDRAM</b><br><a href="#">PDF</a> (PDF) <a href="#">PPT</a> (PPT)                                                            |                    |             |
| W6   | 29.04 Fri. | <a href="#">YouTube</a> <a href="#">Video</a> | <b>M6: DAMOV</b><br><a href="#">PDF</a> (PDF) <a href="#">PPT</a> (PPT)                                                              |                    |             |
| W7   | 06.05 Fri. | <a href="#">YouTube</a> <a href="#">Video</a> | <b>M7: Synchron</b><br><a href="#">PDF</a> (PDF) <a href="#">PPT</a> (PPT)                                                           |                    |             |

# Backup Slides

# SAFARI PhD and Post-Doc Alumni

---

- <https://safari.ethz.ch/safari-alumni/>
- Hasan Hassan (Rivos), **EDAA Outstanding Dissertation Award 2023**; S&P 2020 Best Paper Award, 2020 Pwnie Award, IEEE Micro TP HM 2020
- Christina Giannoula (Univ. of Toronto)
- Minesh Patel (ETH Zurich), **DSN Carter Award for Best Thesis 2022**; ETH Medal 2023; MICRO'20 & DSN'20 Best Paper Awards; ISCA HoF 2021
- Damla Senol Cali (Bionano Genomics), **SRC TECHCON 2019 Best Student Presentation Award**; RECOMB-Seq 2018 Best Poster Award
- Nastaran Hajinazar (Intel)
- Gagandeep Singh (AMD/Xilinx), **FPL 2020 Best Paper Award Finalist**
- Amirali Boroumand (Stanford Univ → Google), **SRC TECHCON 2018 Best Presentation Award**
- Jeremie Kim (Apple), **EDAA Outstanding Dissertation Award 2020**; IEEE Micro Top Picks 2019; ISCA/MICRO HoF 2021
- Nandita Vijaykumar (Univ. of Toronto, Assistant Professor), **ISCA Hall of Fame 2021**
- Kevin Hsieh (Microsoft Research, Senior Researcher)
- Justin Meza (Facebook), **HiPEAC 2015 Best Student Presentation Award**; ICCD 2012 Best Paper Award
- Mohammed Alser (ETH Zurich), **IEEE Turkey Best PhD Thesis Award 2018**
- Yixin Luo (Google), **HPCA 2015 Best Paper Session**
- Kevin Chang (Facebook), **SRC TECHCON 2016 Best Student Presentation Award**
- Rachata Ausavarungrun (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)
- Lois Orosa (Galicia Supercomputing Center, Director)
- Jisung Park (POSTECH, Assistant Professor)
- Gagandeep Singh (AMD/Xilinx, Researcher)

# You Can Join Us!

---

■ <https://safari.ethz.ch/apply/>

## SAFARI Researcher Applications

Sign in

This is the application submission site to be considered for being a researcher in the [SAFARI Research Group](#), directed by [Professor Onur Mutlu](#) ([Publications and Teaching](#)).

If you are interested in doing research in the [SAFARI Research Group](#), please make sure you apply through this submissions site and supply as many of the requested documents and information as possible. Please read and follow the provided instructions and submit as complete an application as possible (given the position you are applying for).

We suggest studying the following materials before submission:

[SAFARI Publications and Courses](#)

[Onur Mutlu's Online Lectures and Course Materials](#)

We strongly recommend that you read and analyze critically as many recent papers from our group as possible. This is the best way to prepare for the application process. Our recommendation is that you use professor Mutlu's methodology for critically analyzing papers.

[Guide On Reviewing Papers](#)

Good luck!

Welcome to the SAFARI at ETH Zurich -- PhD, Postdoc, Internship, Visiting Researcher Applications (SAFARI Researcher Applications) submissions site.

# Data-Driven (Self-Optimizing) Architectures

# System Architecture Design Today

---

- Human-driven
  - Humans design the policies (how to do things)
- Many (too) simple, short-sighted policies all over the system
- No automatic data-driven policy learning
- (Almost) no learning: cannot take lessons from past actions

**Can we design  
fundamentally intelligent architectures?**

# An Intelligent Architecture

---

- Data-driven
  - Machine learns the “best” policies (how to do things)
- Sophisticated, workload-driven, changing, far-sighted policies
- Automatic data-driven policy learning
- All controllers are intelligent data-driven agents

**We need to rethink design  
(of all controllers)**



# Self-Optimizing Memory Controllers

---

- Engin Ipek, Onur Mutlu, José F. Martínez, and Rich Caruana,  
**"Self Optimizing Memory Controllers: A Reinforcement Learning Approach"**  
*Proceedings of the 35th International Symposium on Computer Architecture (ISCA)*, pages 39-50, Beijing, China, June 2008.  
***Selected to the ISCA-50 25-Year Retrospective Issue covering 1996-2020 in 2023 ([Retrospective \(pdf\)](#) [Full Issue](#)).***

## Self-Optimizing Memory Controllers: A Reinforcement Learning Approach

Engin İpek<sup>1,2</sup>   Onur Mutlu<sup>2</sup>   José F. Martínez<sup>1</sup>   Rich Caruana<sup>1</sup>

<sup>1</sup>Cornell University, Ithaca, NY 14850 USA

<sup>2</sup>Microsoft Research, Redmond, WA 98052 USA

# Self-Optimizing Memory Prefetchers

Rahul Bera, Konstantinos Kanellopoulos, Anant Nori, Taha Shahroodi, Sreenivas Subramoney, and Onur Mutlu,  
**"Pythia: A Customizable Hardware Prefetching Framework Using Online Reinforcement Learning"**  
*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](#) (20 minutes)]

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

[[Pythia Source Code](#) (Officially Artifact Evaluated with All Badges)]

[[arXiv version](#)]

***Officially artifact evaluated as available, reusable and reproducible.***



## Pythia: A Customizable Hardware Prefetching Framework Using Online Reinforcement Learning

Rahul Bera<sup>1</sup>

Konstantinos Kanellopoulos<sup>1</sup>

Anant V. Nori<sup>2</sup>

Taha Shahroodi<sup>3,1</sup>

Sreenivas Subramoney<sup>2</sup>

Onur Mutlu<sup>1</sup>

<sup>1</sup>ETH Zürich

<sup>2</sup>Processor Architecture Research Labs, Intel Labs

<sup>3</sup>TU Delft

<https://arxiv.org/pdf/2109.12021.pdf>

# Learning-Based Off-Chip Load Predictors

- Rahul Bera, Konstantinos Kanellopoulos, Shankar Balachandran, David Novo, Ataberk Olgun, Mohammad Sadrosadati, and Onur Mutlu,  
**"Hermes: Accelerating Long-Latency Load Requests via Perceptron-Based Off-Chip Load Prediction"**

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

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

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

[[Talk Video](#) (12 minutes)]

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

[[arXiv version](#)]

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

***Officially artifact evaluated as available, reusable and reproducible.***

***Best paper award at MICRO 2022.***



## Hermes: Accelerating Long-Latency Load Requests via Perceptron-Based Off-Chip Load Prediction

Rahul Bera<sup>1</sup>    Konstantinos Kanellopoulos<sup>1</sup>    Shankar Balachandran<sup>2</sup>    David Novo<sup>3</sup>  
Ataberk Olgun<sup>1</sup>    Mohammad Sadrosadati<sup>1</sup>    Onur Mutlu<sup>1</sup>

<sup>1</sup>ETH Zürich    <sup>2</sup>Intel Processor Architecture Research Lab    <sup>3</sup>LIRMM, Univ. Montpellier, CNRS

**<https://arxiv.org/pdf/2209.00188.pdf>**

# Self-Optimizing Hybrid SSD Controllers

---

Gagandeep Singh, Rakesh Nadig, Jisung Park, Rahul Bera, Nastaran Hajinazar, David Novo, Juan Gomez-Luna, Sander Stuijk, Henk Corporaal, and Onur Mutlu,

## **"Sibyl: Adaptive and Extensible Data Placement in Hybrid Storage Systems Using Online Reinforcement Learning"**

*Proceedings of the 49th International Symposium on Computer Architecture (ISCA), New York, June 2022.*

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

[[arXiv version](#)]

[[Sibyl Source Code](#)]

[[Talk Video](#) (16 minutes)]

## **Sibyl: Adaptive and Extensible Data Placement in Hybrid Storage Systems Using Online Reinforcement Learning**

|                              |                              |                            |                             |                                 |
|------------------------------|------------------------------|----------------------------|-----------------------------|---------------------------------|
| Gagandeep Singh <sup>1</sup> | Rakesh Nadig <sup>1</sup>    | Jisung Park <sup>1</sup>   | Rahul Bera <sup>1</sup>     | Nastaran Hajinazar <sup>1</sup> |
| David Novo <sup>3</sup>      | Juan Gómez-Luna <sup>1</sup> | Sander Stuijk <sup>2</sup> | Henk Corporaal <sup>2</sup> | Onur Mutlu <sup>1</sup>         |

<sup>1</sup>ETH Zürich

<sup>2</sup>Eindhoven University of Technology

<sup>3</sup>LIRMM, Univ. Montpellier, CNRS

## Data-Driven (Self-Optimizing) Computing Architectures

# Data-Characteristic-Aware Architectures

# Data-Aware Architectures

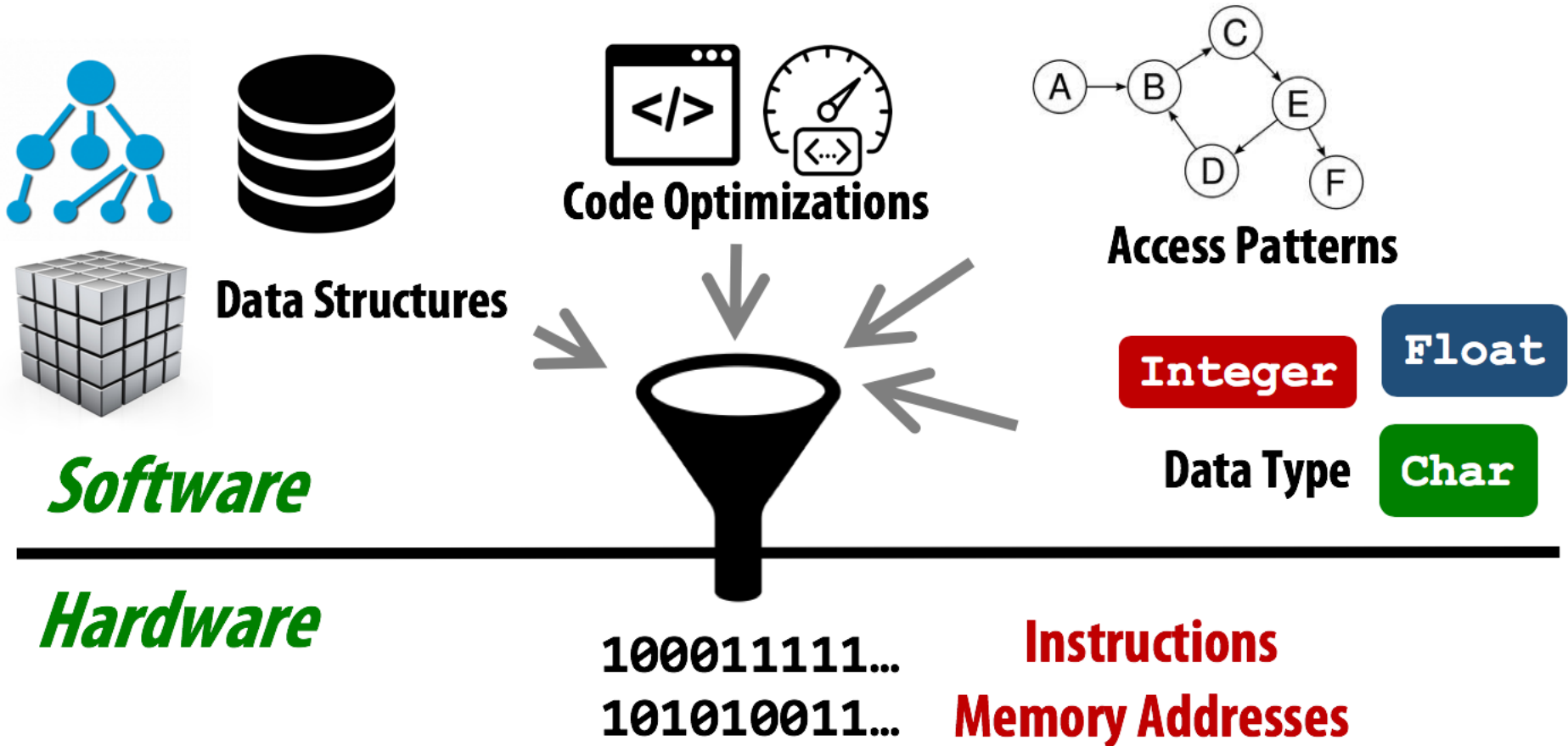
---

- A data-aware architecture understands what it can do with and to each piece of data
- It makes use of different properties of data to improve performance, efficiency and other metrics
  - Compressibility
  - Approximability
  - Locality
  - Sparsity
  - Criticality for Computation X
  - Access Semantics
  - ...



# One Problem: Limited Expressiveness

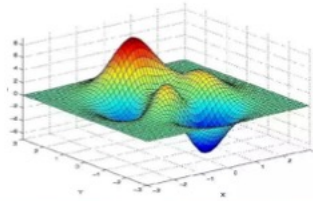
## Higher-level information is not visible to HW



# A Solution: More Expressive Interfaces

**Performance**

**Software**



**Functionality**



**ISA  
Virtual Memory**

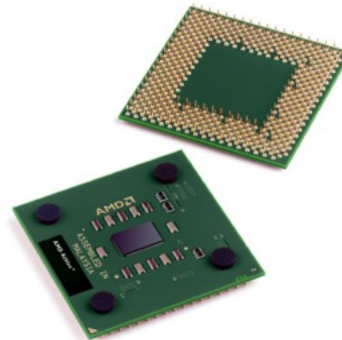
**Higher-level  
Program  
Semantics**

**Expressive  
Memory  
“XMem”**

**Hardware**



wiseGEEK



# 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<sup>†§</sup> Abhilasha Jain<sup>†</sup> Diptesh Majumdar<sup>†</sup> Kevin Hsieh<sup>†</sup> Gennady Pekhimenko<sup>‡</sup>  
Eiman Ebrahimi<sup>⌘</sup> Nastaran Hajinazar<sup>†</sup> Phillip B. Gibbons<sup>†</sup> Onur Mutlu<sup>§†</sup>

<sup>†</sup>Carnegie Mellon University

<sup>‡</sup>University of Toronto

<sup>⌘</sup>NVIDIA

<sup>†</sup>Simon Fraser University

<sup>§</sup>ETH Zürich

# 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 <sup>†§</sup>        | Eiman Ebrahimi <sup>‡</sup> | Kevin Hsieh <sup>†</sup> |
| Phillip B. Gibbons <sup>†</sup>         | Onur Mutlu <sup>§†</sup>    |                          |
| <sup>†</sup> Carnegie Mellon University | <sup>‡</sup> NVIDIA         | <sup>§</sup> ETH Zürich  |

# Open-Source Frameworks for Data-Aware Systems

---

- Nandita Vijaykumar, Ataberk Olgun, Konstantinos Kanellopoulos, F. Nisa Bostanci, Hasan Hassan, Mehrshad Lotfi, Phillip B. Gibbons, and Onur Mutlu,  
**"MetaSys: A Practical Open-source Metadata Management System to Implement and Evaluate Cross-layer Optimizations"**  
*ACM Transactions on Architecture and Code Optimization (TACO)*, June 2022.  
[arXiv version]  
Presented at the 18th HiPEAC Conference, Toulouse, France, January 2023.  
[Slides (pptx) (pdf)]  
[Preliminary Talk Video (14 minutes)]  
[SAFARI Live Seminar Video (1 hour 26 minutes)]  
[MetaSys Source Code]  
***Best paper award at HiPEAC 2023.***

## MetaSys: A Practical Open-Source Metadata Management System to Implement and Evaluate Cross-Layer Optimizations

Nandita Vijaykumar\*    Ataberk Olgun<sup>§</sup>    Konstantinos Kanellopoulos<sup>§</sup>    Hasan Hassan<sup>§</sup>  
Mehrshad Lotfi<sup>§</sup>    Phillip B. Gibbons<sup>†</sup>    Onur Mutlu<sup>§</sup>

\*University of Toronto

<sup>§</sup>ETH Zürich

<sup>†</sup>Carnegie Mellon University

# 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, bknessib, kvaid}@microsoft.com



# 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 Kanellopoulos<sup>1</sup> Nandita Vijaykumar<sup>2,1</sup> Christina Giannoula<sup>1,3</sup> Roknoddin Azizi<sup>1</sup>  
Skanda Koppula<sup>1</sup> Nika Mansouri Ghiasi<sup>1</sup> Taha Shahroodi<sup>1</sup> Juan Gomez Luna<sup>1</sup> Onur Mutlu<sup>1,2</sup>

<sup>1</sup>ETH Zürich

<sup>2</sup>Carnegie Mellon University

<sup>3</sup>National Technical University of Athens

# 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

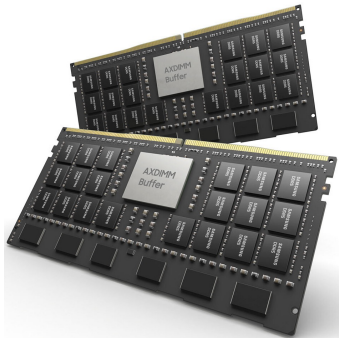
Nastaran Hajinazar<sup>\*†</sup> Pratyush Patel<sup>✕</sup> Minesh Patel<sup>\*</sup> Konstantinos Kanellopoulos<sup>\*</sup> Saugata Ghose<sup>‡</sup>  
Rachata Ausavarungnirun<sup>⊙</sup> Geraldo F. Oliveira<sup>\*</sup> Jonathan Appavoo<sup>◇</sup> Vivek Seshadri<sup>▽</sup> Onur Mutlu<sup>\*‡</sup>

<sup>\*</sup>ETH Zürich   <sup>†</sup>Simon Fraser University   <sup>✕</sup>University of Washington   <sup>‡</sup>Carnegie Mellon University  
<sup>⊙</sup>King Mongkut's University of Technology North Bangkok   <sup>◇</sup>Boston University   <sup>▽</sup>Microsoft Research India

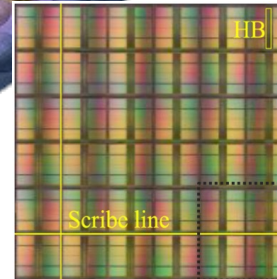
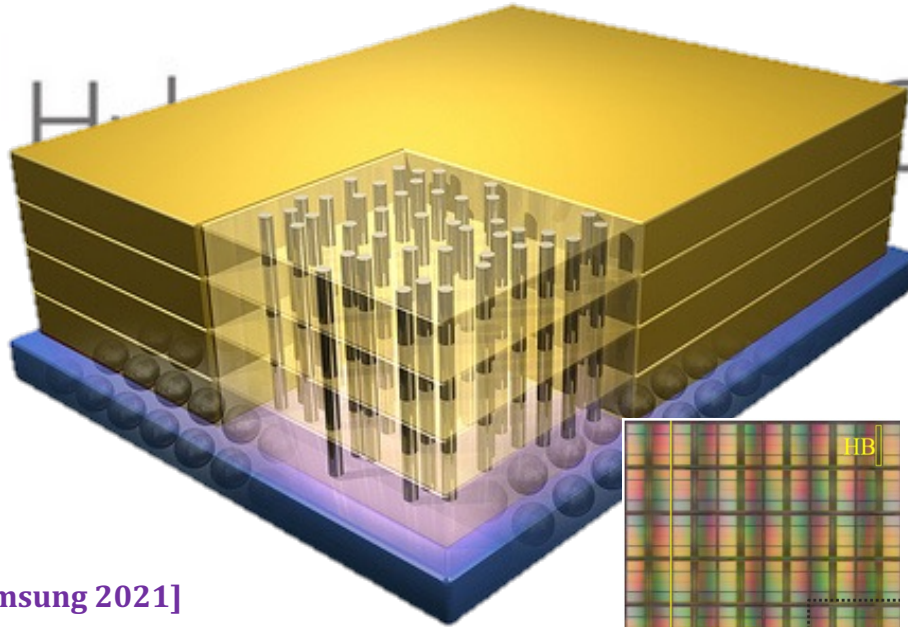
## Data-Characteristic-Aware Computing Architectures

# More Background Slides

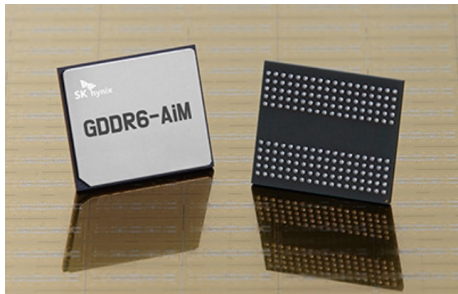
# Processing-in-Memory Landscape Today



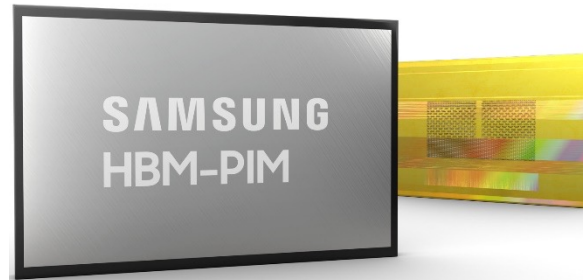
[Samsung 2021]



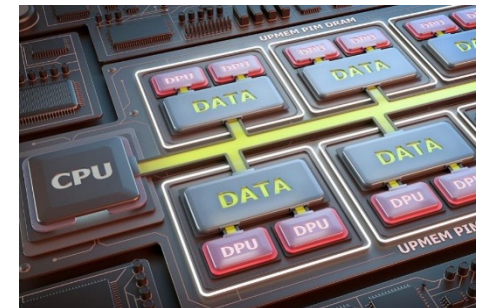
[Alibaba 2022]



[SK Hynix 2022]



[Samsung 2021]



[UPMEM 2019]

# Memory Scaling Issues **Are** Real

---

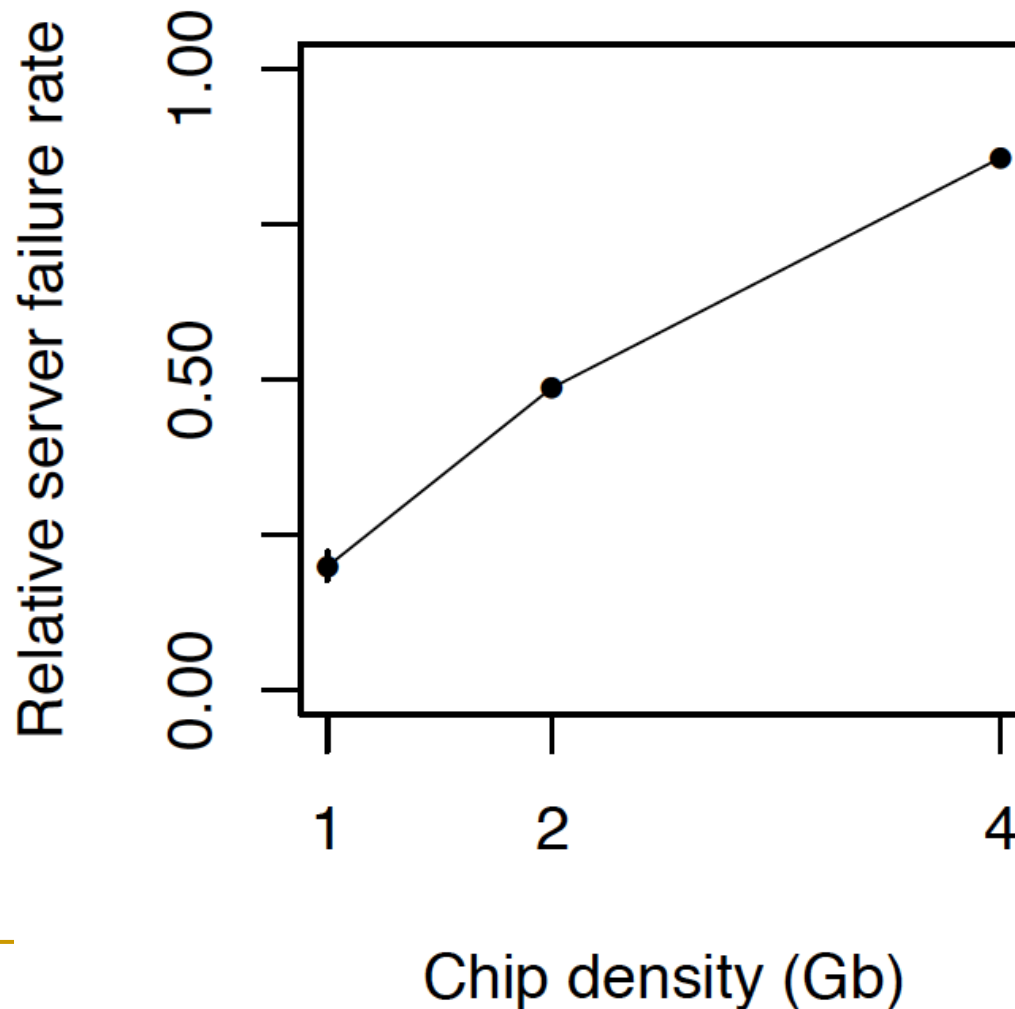
- Onur Mutlu,  
**"Memory Scaling: A Systems Architecture Perspective"**  
*Proceedings of the 5th International Memory Workshop (IMW)*, Monterey, CA, May 2013. Slides  
(pptx) (pdf)  
EETimes Reprint

## Memory Scaling: A Systems Architecture Perspective

Onur Mutlu  
Carnegie Mellon University  
onur@cmu.edu  
<http://users.ece.cmu.edu/~omutlu/>

# As Memory Scales, It Becomes Unreliable

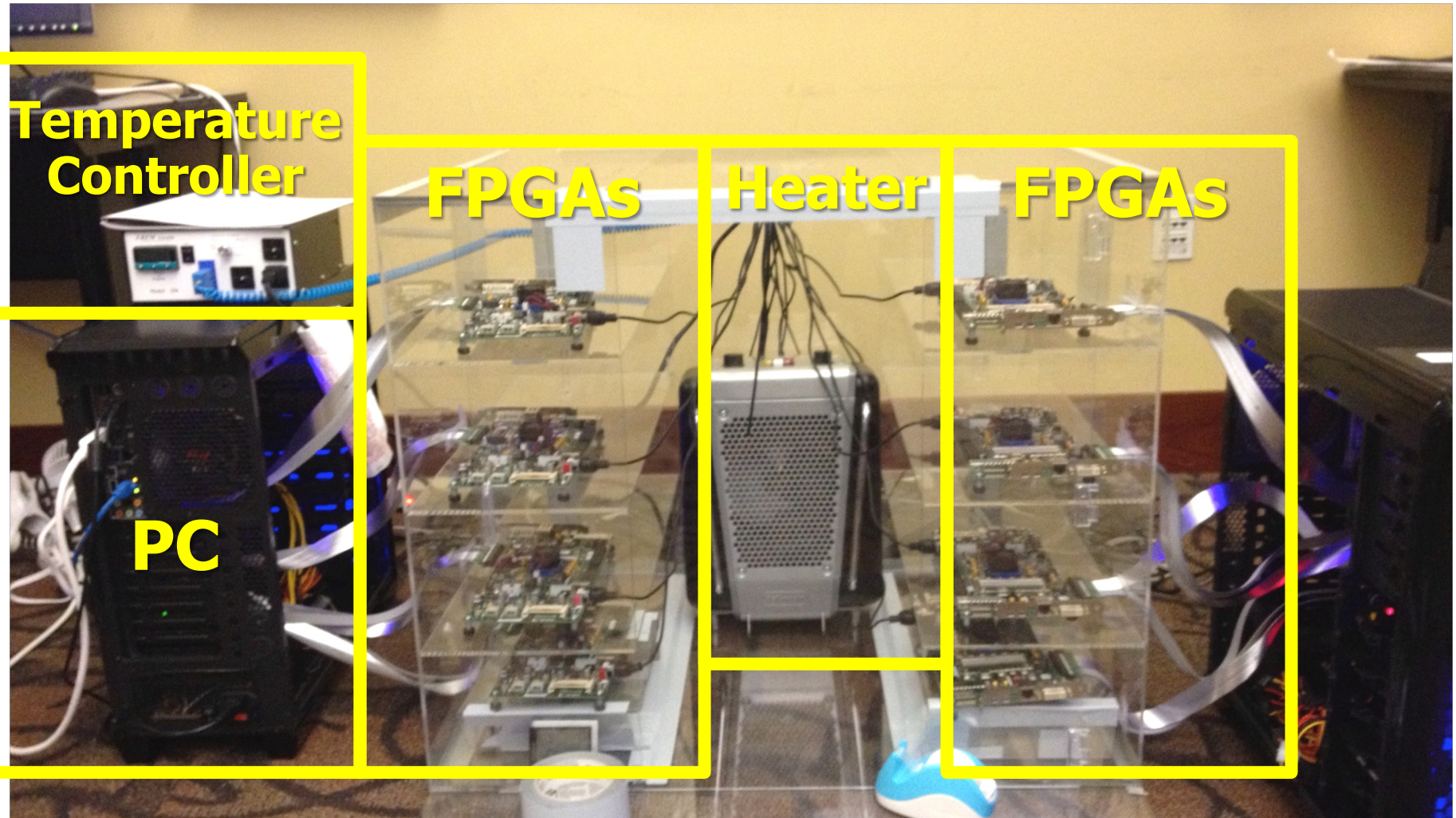
- Data from all of Facebook's servers worldwide
- Meza+, "Revisiting Memory Errors in Large-Scale Production Data Centers," DSN'15.



*Intuition:  
quadratic  
increase  
in  
capacity*



# Infrastructures to Understand Such Issues

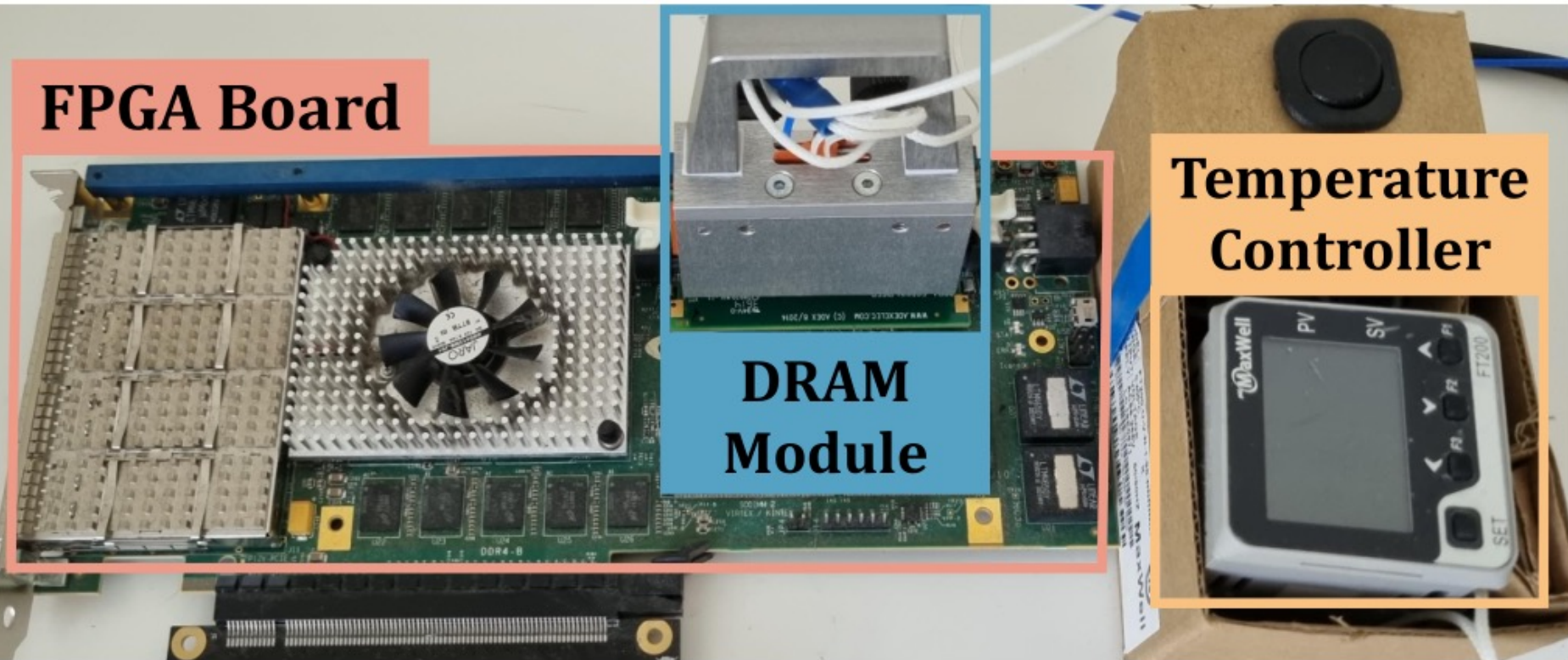


# Memory Testing Infrastructures

**FPGA Board**

**DRAM  
Module**

**Temperature  
Controller**

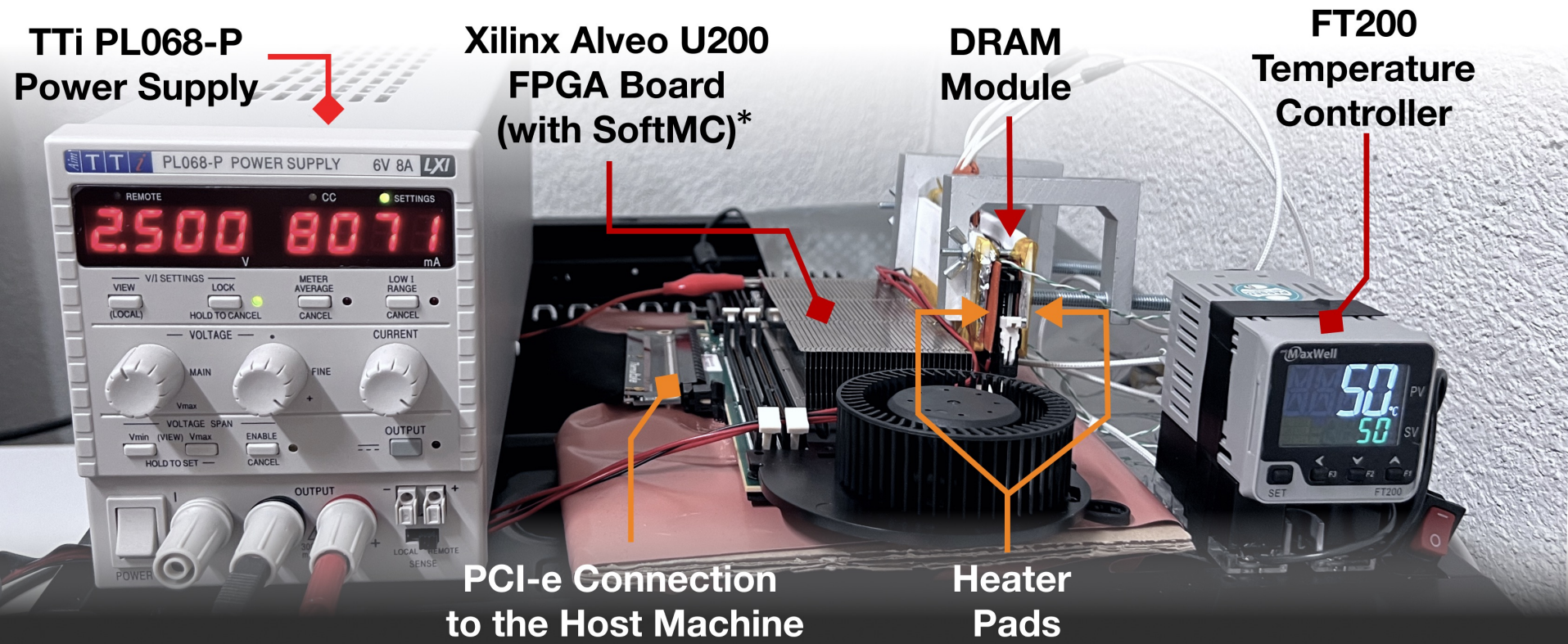


*\* SoftMC [Hassan+, HPCA'17] enhanced for DDR4*



# Updated Memory Testing Infrastructure

FPGA-based SoftMC (Xilinx Virtex UltraScale+ XCU200)



Fine-grained control over DRAM commands,  
**timing ( $\pm 1.5\text{ns}$ )**, **temperature ( $\pm 0.1^\circ\text{C}$ )**,  
and **voltage ( $\pm 1\text{mV}$ )**

# SoftMC: Open Source DRAM Infrastructure

---

- Hasan Hassan, Nandita Vijaykumar, Samira Khan, Saugata Ghose, Kevin Chang, Gennady Pekhimenko, Donghyuk Lee, Oguz Ergin, and Onur Mutlu,

**"SoftMC: A Flexible and Practical Open-Source Infrastructure for Enabling Experimental DRAM Studies"**

*Proceedings of the 23rd International Symposium on High-Performance Computer Architecture (HPCA), Austin, TX, USA, February 2017.*

[Slides (pptx) (pdf)] [Lightning Session Slides (pptx) (pdf)]

[Full Talk Lecture (39 minutes)]

[Source Code]

## SoftMC: A Flexible and Practical Open-Source Infrastructure for Enabling Experimental DRAM Studies

Hasan Hassan<sup>1,2,3</sup> Nandita Vijaykumar<sup>3</sup> Samira Khan<sup>4,3</sup> Saugata Ghose<sup>3</sup> Kevin Chang<sup>3</sup>  
Gennady Pekhimenko<sup>5,3</sup> Donghyuk Lee<sup>6,3</sup> Oguz Ergin<sup>2</sup> Onur Mutlu<sup>1,3</sup>

<sup>1</sup>ETH Zürich    <sup>2</sup>TOBB University of Economics & Technology    <sup>3</sup>Carnegie Mellon University  
<sup>4</sup>University of Virginia    <sup>5</sup>Microsoft Research    <sup>6</sup>NVIDIA Research

# DRAM Bender

---

- Ataberk Olgun, Hasan Hassan, A Giray Yağlıkçı, Yahya Can Tuğrul, Lois Orosa, Haocong Luo, Minesh Patel, Oğuz Ergin, and Onur Mutlu,  
**"DRAM Bender: An Extensible and Versatile FPGA-based Infrastructure to Easily Test State-of-the-art DRAM Chips"**  
*IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, 2023.  
[[Extended arXiv version](#)]  
[[DRAM Bender Source Code](#)]  
[[DRAM Bender Tutorial Video](#) (43 minutes)]

## DRAM Bender: An Extensible and Versatile FPGA-based Infrastructure to Easily Test State-of-the-art DRAM Chips

Ataberk Olgun<sup>§</sup>      Hasan Hassan<sup>§</sup>      A. Giray Yağlıkçı<sup>§</sup>      Yahya Can Tuğrul<sup>§†</sup>  
Lois Orosa<sup>§⊙</sup>      Haocong Luo<sup>§</sup>      Minesh Patel<sup>§</sup>      Oğuz Ergin<sup>†</sup>      Onur Mutlu<sup>§</sup>  
          <sup>§</sup>*ETH Zürich*      <sup>†</sup>*TOBB ETÜ*      <sup>⊙</sup>*Galician Supercomputing Center*

# A Curious Phenomenon [Kim et al., ISCA 2014]

---

One can  
predictably induce errors  
in most DRAM memory chips

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



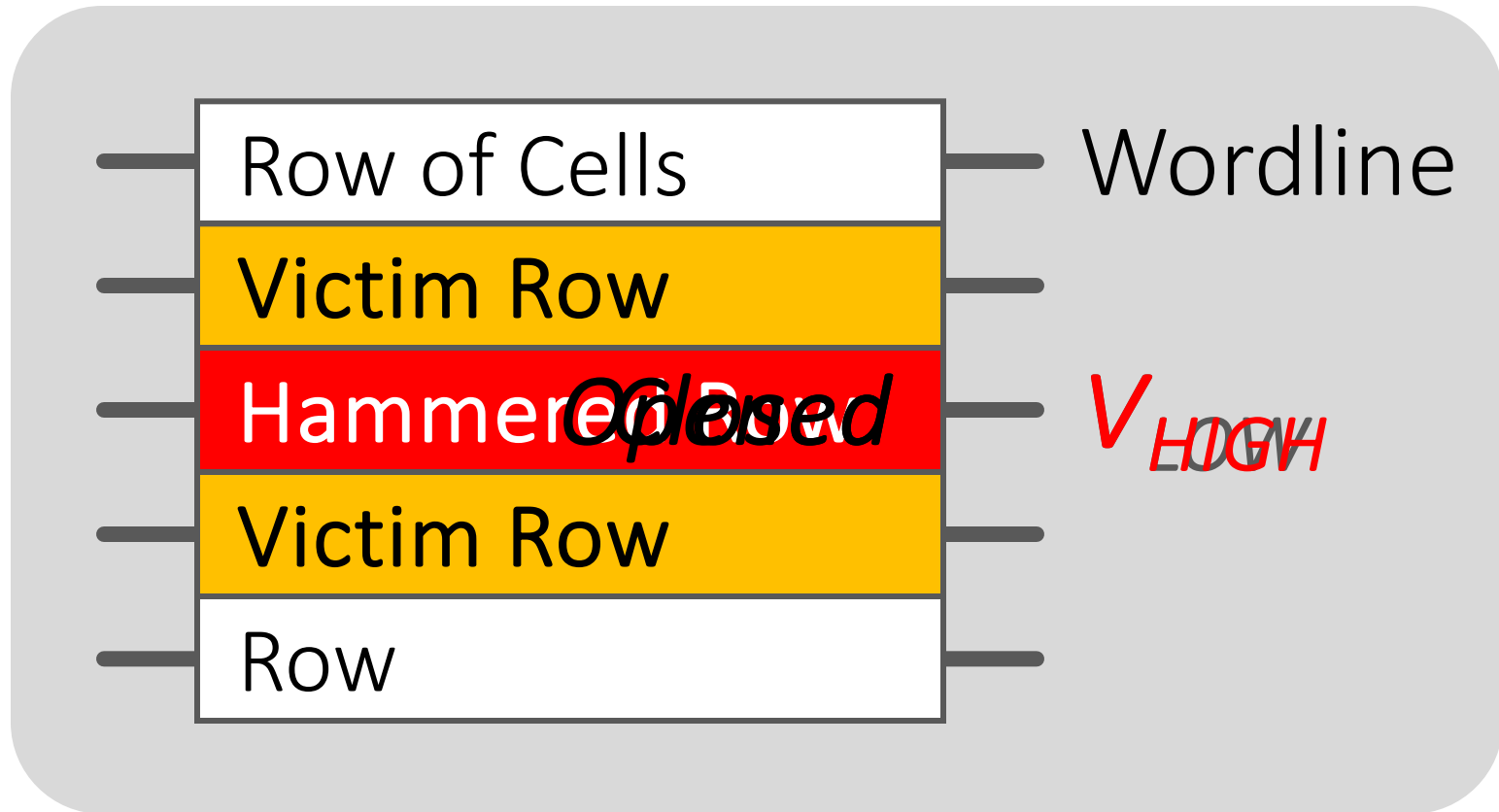


Rowhammer

---



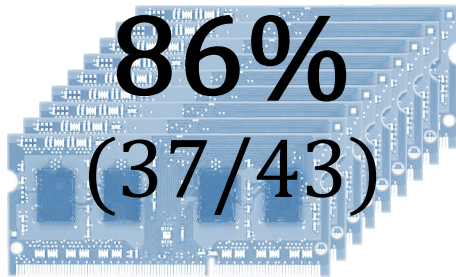
# Modern Memory is Prone to Disturbance Errors



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

# Most DRAM Modules Are Vulnerable

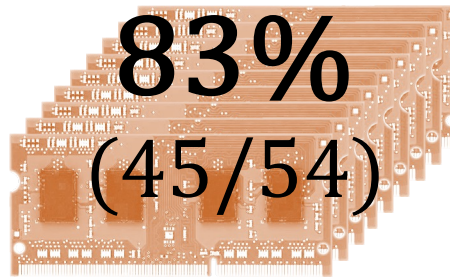
A company



Up to  
 $1.0 \times 10^7$

errors

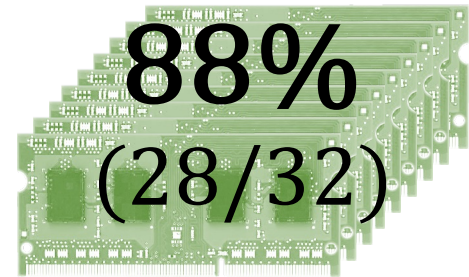
B company



Up to  
 $2.7 \times 10^6$

errors

C company



Up to  
 $3.3 \times 10^5$

errors

# The RowHammer Vulnerability

---

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

# RowHammer [ISCA 2014]

- 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 \(pptx\) \(pdf\)\]](#) [\[Source Code and Data\]](#) [\[Lecture Video\]](#) (1 hr 49 mins), 25 September 2020]  
*One of the 7 papers of 2012-2017 selected as Top Picks in Hardware and Embedded Security for IEEE TCAD ([link](#)).*  
*Selected to the ISCA-50 25-Year Retrospective Issue covering 1996-2020 in 2023 ([Retrospective \(pdf\)](#) [Full Issue](#)).*

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

Yoongu Kim<sup>1</sup> Ross Daly\* Jeremie Kim<sup>1</sup> Chris Fallin\* Ji Hye Lee<sup>1</sup>  
Donghyuk Lee<sup>1</sup> Chris Wilkerson<sup>2</sup> Konrad Lai Onur Mutlu<sup>1</sup>

<sup>1</sup>Carnegie Mellon University <sup>2</sup>Intel Labs

# Memory Scaling Issues **Are** Real

---

- Onur Mutlu,  
**"The RowHammer Problem and Other Issues We May Face as Memory Becomes Denser"**

*Invited Paper in Proceedings of the Design, Automation, and Test in Europe Conference (**DATE**), Lausanne, Switzerland, March 2017.*

*[Slides (pptx) (pdf)]*

## The RowHammer Problem and Other Issues We May Face as Memory Becomes Denser

Onur Mutlu  
ETH Zürich  
onur.mutlu@inf.ethz.ch  
<https://people.inf.ethz.ch/omutlu>

# Memory Scaling Issues **Are** Real

---

- 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<sup>§‡</sup>      Jeremie S. Kim<sup>‡§</sup>  
§ETH Zürich      ‡Carnegie Mellon University

# Memory Scaling Issues **Are** Real

---

- Onur Mutlu, Ataberk Olgun, and A. Giray Yaglikci,  
**"Fundamentally Understanding and Solving RowHammer"**  
*Invited Special Session Paper at the 28th Asia and South Pacific Design Automation Conference (ASP-DAC), Tokyo, Japan, January 2023.*  
[arXiv version]  
[Slides (pptx) (pdf)]  
[Talk Video (26 minutes)]

## Fundamentally Understanding and Solving RowHammer

Onur Mutlu  
onur.mutlu@safari.ethz.ch  
ETH Zürich  
Zürich, Switzerland

Ataberk Olgun  
ataberk.olgund@safari.ethz.ch  
ETH Zürich  
Zürich, Switzerland

A. Giray Yağlıkçı  
giray.yaglikci@safari.ethz.ch  
ETH Zürich  
Zürich, Switzerland



# The Story of RowHammer Tutorial ...

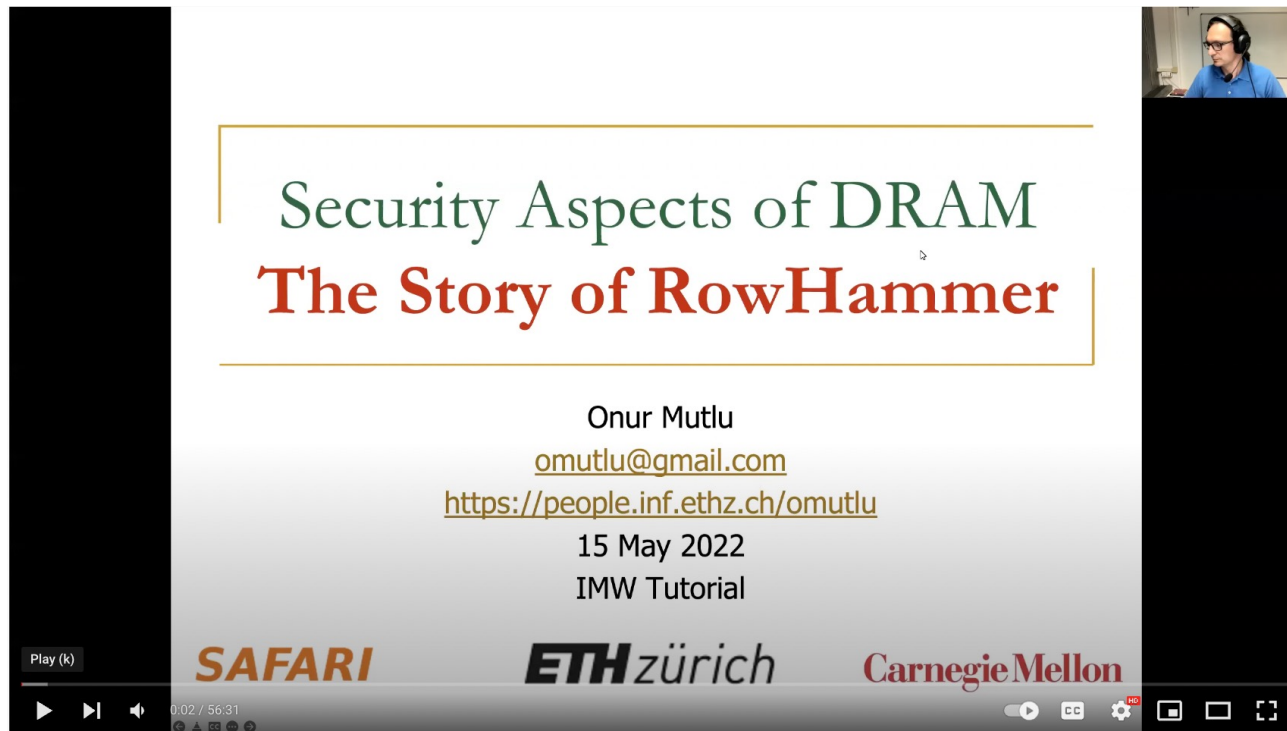
Onur Mutlu,

**"Security Aspects of DRAM: The Story of RowHammer"**

*Invited Tutorial at 14th IEEE Electron Devices Society International Memory Workshop (IMW), Dresden, Germany, May 2022.*

[Slides (pptx)(pdf)]

[Tutorial Video (57 minutes)]



Recent Premieres

The Story of RowHammer – Invited Tutorial at IMW 2022 (Intl. Memory Workshop) - Onur Mutlu

598 views • Premiered Jul 27, 2022

👍 19    🗨 DISLIKE    ➦ SHARE    ⬇ DOWNLOAD    🗂 CLIP    💾 SAVE    ...



Onur Mutlu Lectures  
27.6K subscribers

<https://www.youtube.com/watch?v=37hWqlkQRG0>

ANALYTICS

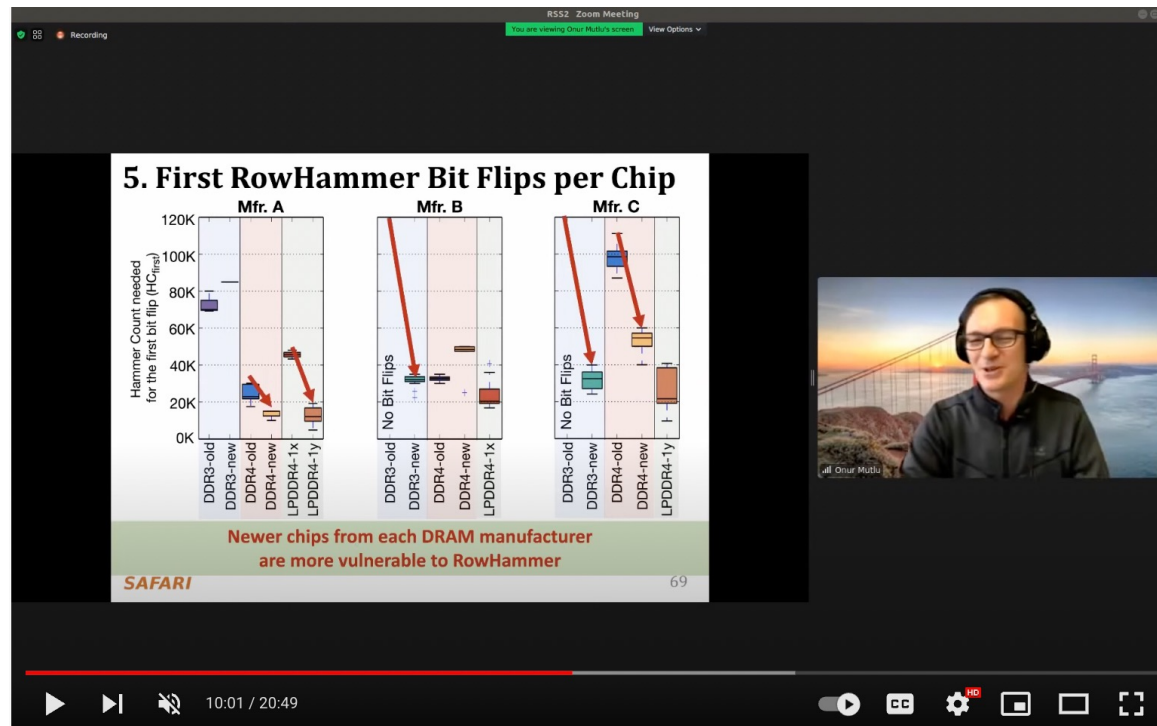
EDIT VIDEO

# 10 Years of RowHammer in 20 Minutes

- Onur Mutlu,  
**"The Story of RowHammer"**

*Invited Talk at the Workshop on Robust and Safe Software 2.0 (RSS2), held with the 27th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS), Virtual, 28 February 2022.*

[Slides (pptx) (pdf)]



The Story of RowHammer - Invited Talk in Robust & Safe Software Workshop (ASPLOS 2022) - Onur Mutlu

402 views • Premiered Apr 27, 2022

👍 17    🗑 DISLIKE    ➦ SHARE    ⬇ DOWNLOAD    ✂ CLIP    ⚙ SAVE    ...



Onur Mutlu Lectures  
24.5K subscribers

<https://www.youtube.com/watch?v=ctKTRyi96Bk>

SUBSCRIBED



## Main Memory Needs Intelligent Controllers

# Industry's Intelligent DRAM Controllers (I)

## ISSCC 2023 / SESSION 28 / HIGH-DENSITY MEMORIES /

### **28.8 A 1.1V 16Gb DDR5 DRAM with Probabilistic-Aggressor Tracking, Refresh-Management Functionality, Per-Row Hammer Tracking, a Multi-Step Precharge, and Core-Bias Modulation for Security and Reliability Enhancement**

Woongrae Kim, Chulmoon Jung, Seongnyuh Yoo, Duckhwa Hong, Jeongjin Hwang, Jungmin Yoon, Ohyong Jung, Joonwoo Choi, Sanga Hyun, Mankeun Kang, Sangho Lee, Dohong Kim, Sanghyun Ku, Donhyun Choi, Nogeun Joo, Sangwoo Yoon, Junseok Noh, Byeongyong Go, Cheolhoe Kim, Sunil Hwang, Mihyun Hwang, Seol-Min Yi, Hyungmin Kim, Sanghyuk Heo, Yeonsu Jang, Kyoungchul Jang, Shinho Chu, Yoonna Oh, Kwidong Kim, Junghyun Kim, Soohwan Kim, Jeongtae Hwang, Sangil Park, Junphyo Lee, Inchul Jeong, Joohwan Cho, Jonghwan Kim

SK hynix Semiconductor, Icheon, Korea





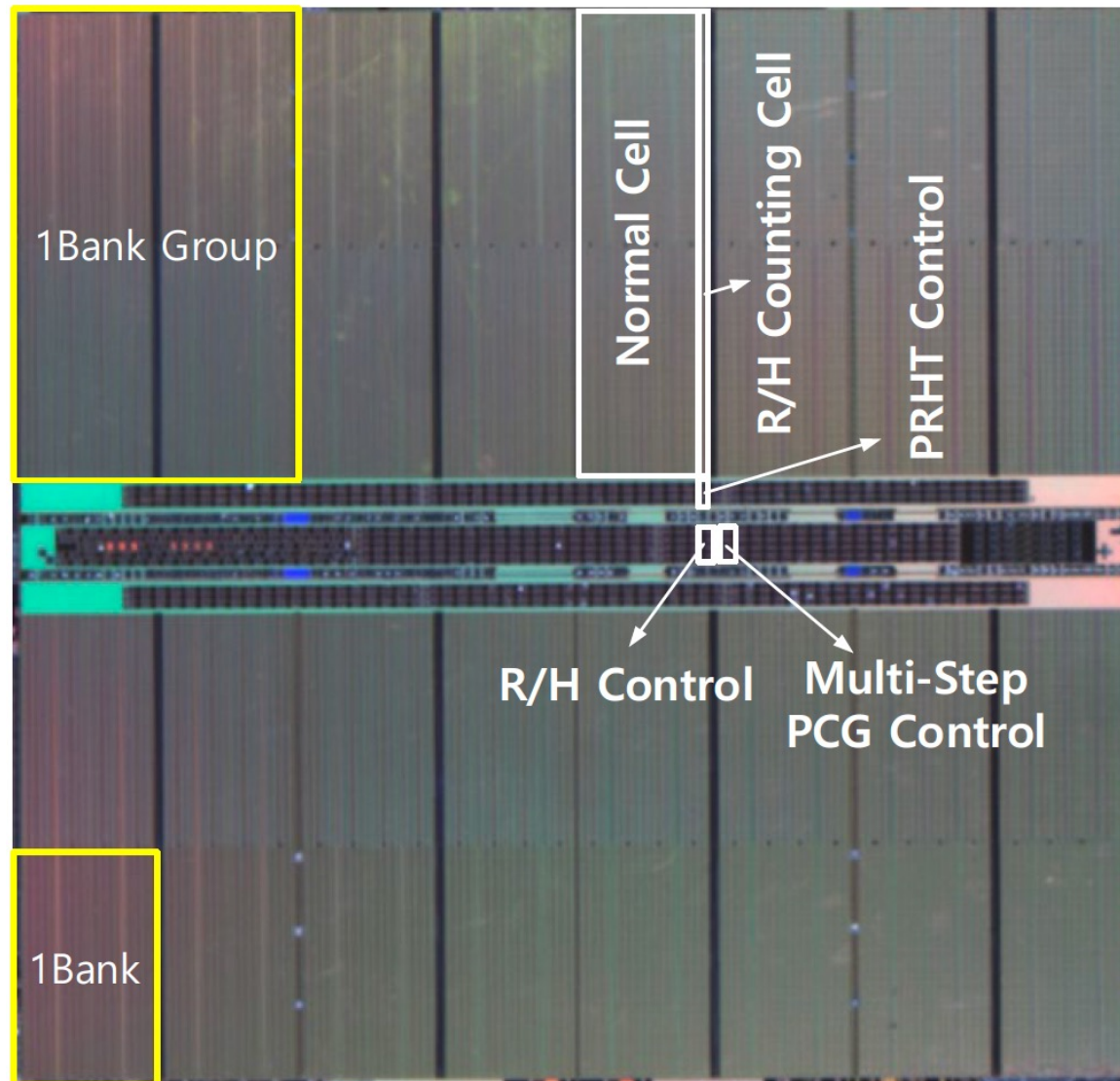
# Industry's Intelligent DRAM Controllers (II)

---

SK hynix Semiconductor, Icheon, Korea

DRAM products have been recently adopted in a wide range of high-performance computing applications: such as in cloud computing, in big data systems, and IoT devices. This demand creates larger memory capacity requirements, thereby requiring aggressive DRAM technology node scaling to reduce the cost per bit [1,2]. However, DRAM manufacturers are facing technology scaling challenges due to row hammer and refresh retention time beyond 1a-nm [2]. Row hammer is a failure mechanism, where repeatedly activating a DRAM row disturbs data in adjacent rows. Scaling down severely threatens reliability since a reduction of DRAM cell size leads to a reduction in the intrinsic row hammer tolerance [2,3]. To improve row hammer tolerance, there is a need to probabilistically activate adjacent rows with carefully sampled active addresses and to improve intrinsic row hammer tolerance [2]. In this paper, row-hammer-protection and refresh-management schemes are presented to guarantee DRAM security and reliability despite the aggressive scaling from 1a-nm to sub 10-nm nodes. The probabilistic-aggressor-tracking scheme with a refresh-management function (RFM) and per-row hammer tracking (PRHT) improve DRAM resilience. A multi-step precharge reinforces intrinsic row-hammer tolerance and a core-bias modulation improves retention time: even in the face of cell-transistor degradation due to technology scaling. This comprehensive scheme leads to a reduced probability of failure, due to row hammer attacks, by 93.1% and an improvement in retention time by 17%.

# Industry's Intelligent DRAM Controllers (III)



ISSCC 2023 / SESSION 28 / HIGH-DENSITY MEMORIES

**28.8 A 1.1V 16Gb DDR5 DRAM with Probabilistic-Aggressor Tracking, Refresh-Management Functionality, Per-Row Hammer Tracking, a Multi-Step Precharge, and Core-Bias Modulation for Security and Reliability Enhancement**

Woongrae Kim, Chulmoon Jung, Seongnyuh Yoo, Duckhwa Hong, Jeongjin Hwang, Jungmin Yoon, Ohyoung Jung, Joonwoo Choi, Sanga Hyun, Mankeun Kang, Sangho Lee, Dohong Kim, Sanghyun Ku, Donhyun Choi, Nogeun Joo, Sangwoo Yoon, Junseok Noh, Byeongyong Go, Cheolhoe Kim, Sunil Hwang, Mihyun Hwang, Seol-Min Yi, Hyungmin Kim, Sanghyuk Heo, Yeonsu Jang, Kyoungchul Jang, Shinho Chu, Yoonna Oh, Kwidong Kim, Junghyun Kim, Soohwan Kim, Jeongtae Hwang, Sangil Park, Junphyo Lee, Inchul Jeong, Joohwan Cho, Jonghwan Kim

SK hynix Semiconductor, Icheon, Korea

## DSAC: Low-Cost Rowhammer Mitigation Using In-DRAM Stochastic and Approximate Counting Algorithm

Seungki Hong   Dongha Kim   Jaehyung Lee   Reum Oh  
Changsik Yoo   Sangjoon Hwang   Jooyoung Lee

DRAM Design Team, Memory Division, Samsung Electronics

<https://arxiv.org/pdf/2302.03591v1.pdf>



# Intel Optane Persistent Memory (2019)

---

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



# Emerging Memories Also Need Intelligent Controllers

---

- 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<sup>†</sup> Engin Ipek<sup>†</sup> Onur Mutlu<sup>‡</sup> Doug Burger<sup>†</sup>

<sup>†</sup>Computer Architecture Group  
Microsoft Research  
Redmond, WA  
{blee, ipek, dburger}@microsoft.com

<sup>‡</sup>Computer Architecture Laboratory  
Carnegie Mellon University  
Pittsburgh, PA  
onur@cmu.edu

Intelligent  
Memory Controllers  
Can Avoid Many Failures  
& Enable Better Scaling

# Three Key Systems & Application Trends

---

## 1. Data access is the major bottleneck

- ▣ Applications are increasingly data hungry

## 2. Energy consumption is a key limiter

## 3. Data movement energy dominates compute

- ▣ Especially true for off-chip to on-chip movement

# Do We Want This?

---





# Or This?

---





High Performance,

Energy Efficient,

Sustainable

(All at the Same Time)



# The Problem

---

Data access is the major performance and energy bottleneck

Our current  
design principles  
cause great energy waste  
(and great performance loss)

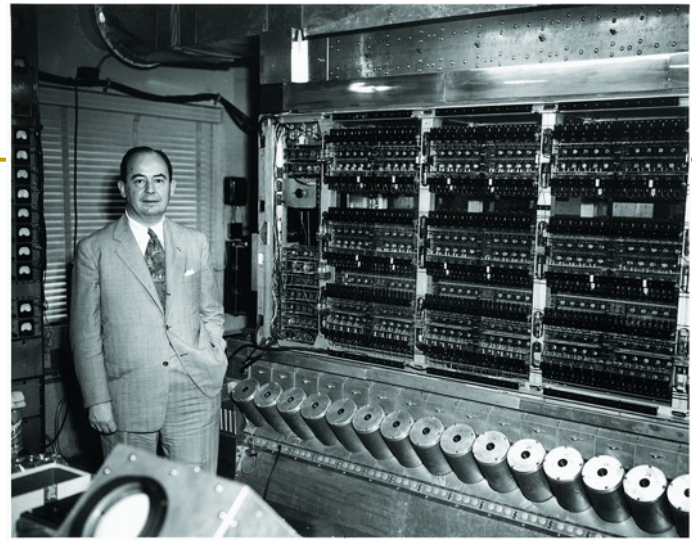
# The Problem

---

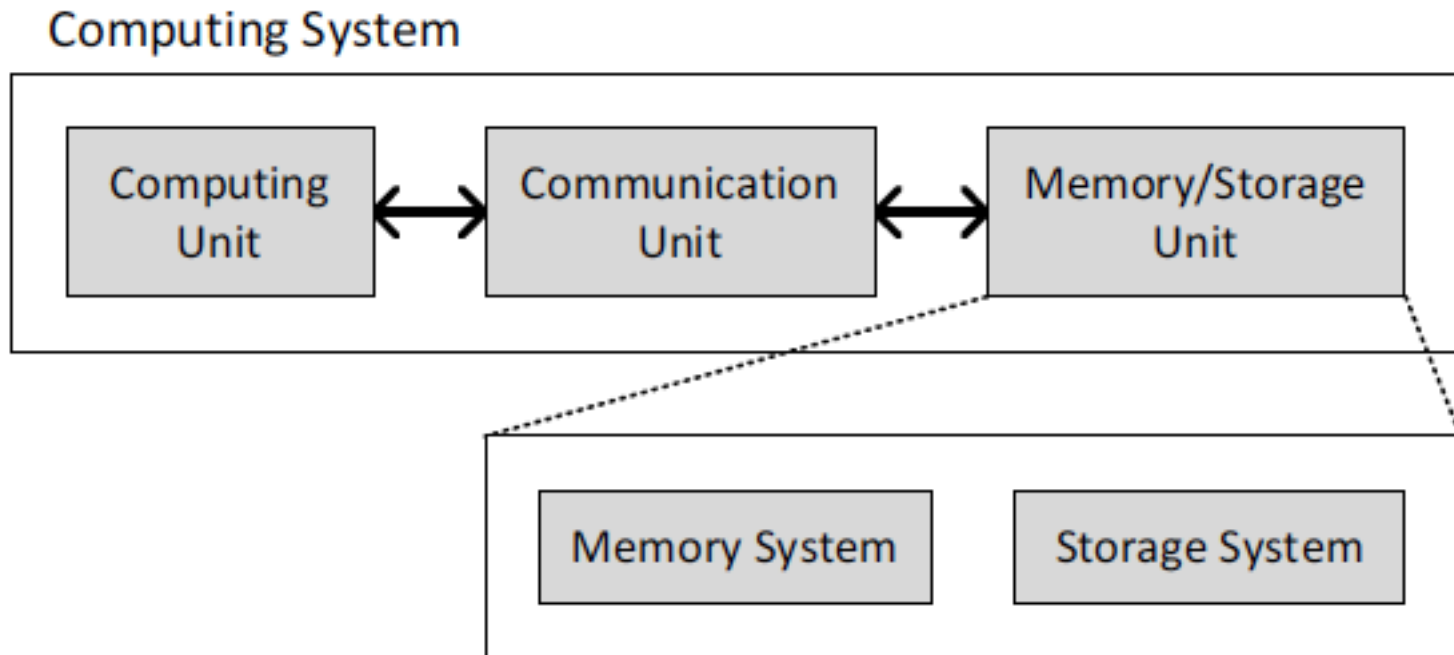
Processing of data  
is performed  
far away from the data

# A Computing System

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

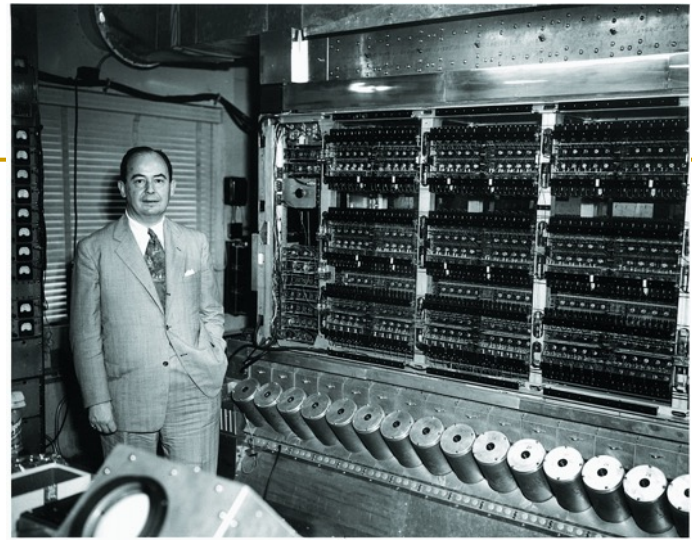


Burks, Goldstein, von Neumann, "Preliminary discussion of the logical design of an electronic computing instrument," 1946.



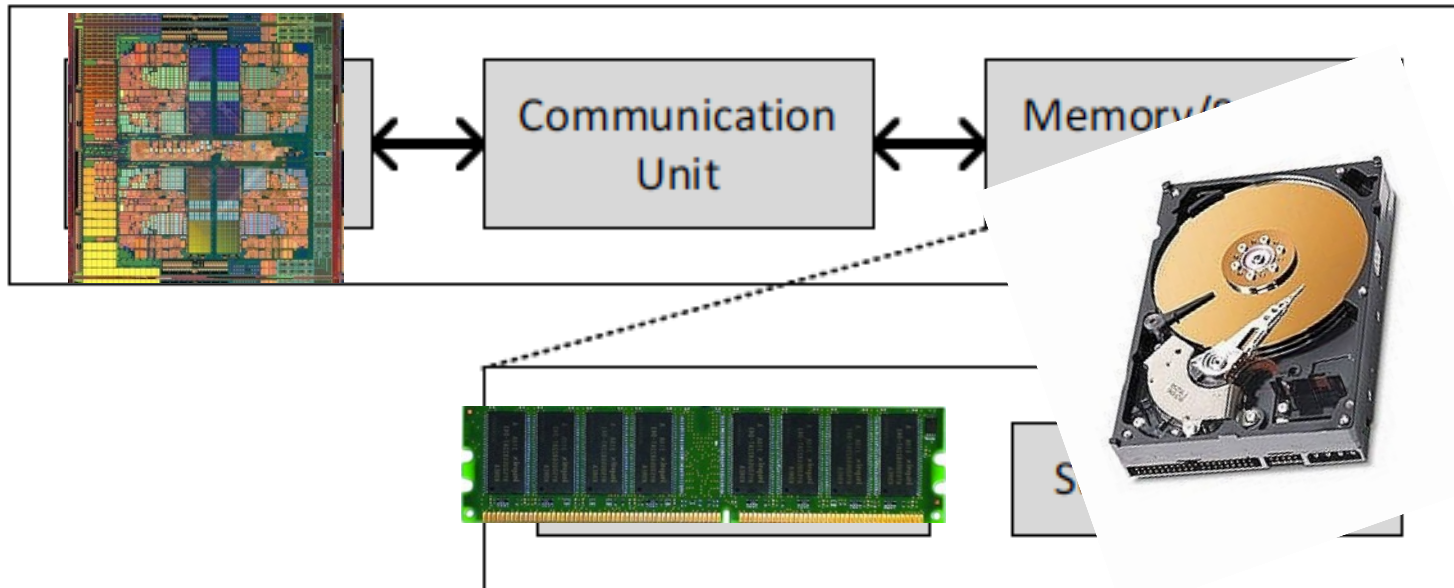
# A Computing System

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



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

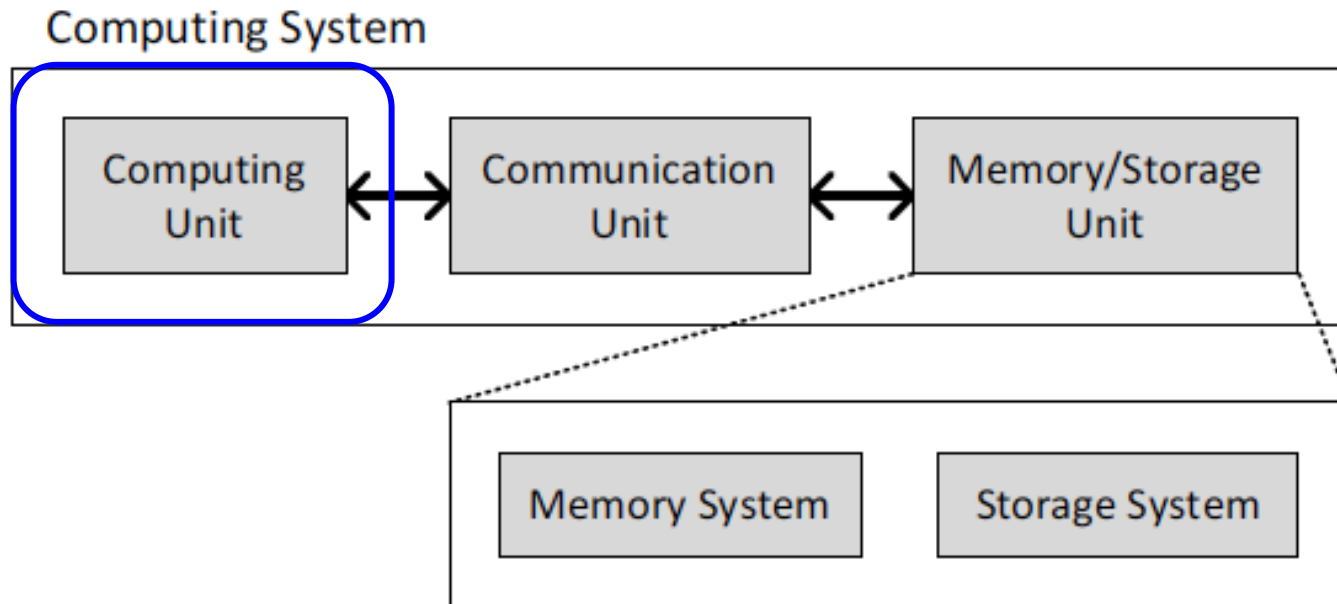
## Computing System



# Today's Computing Systems

---

- Processor centric
- All data processed in the processor → at great system cost



# It's the Memory, Stupid!

---

- **"It's the Memory, Stupid!"** (Richard Sites, MPR, 1996)

**RICHARD SITES**

## **It's the Memory, Stupid!**

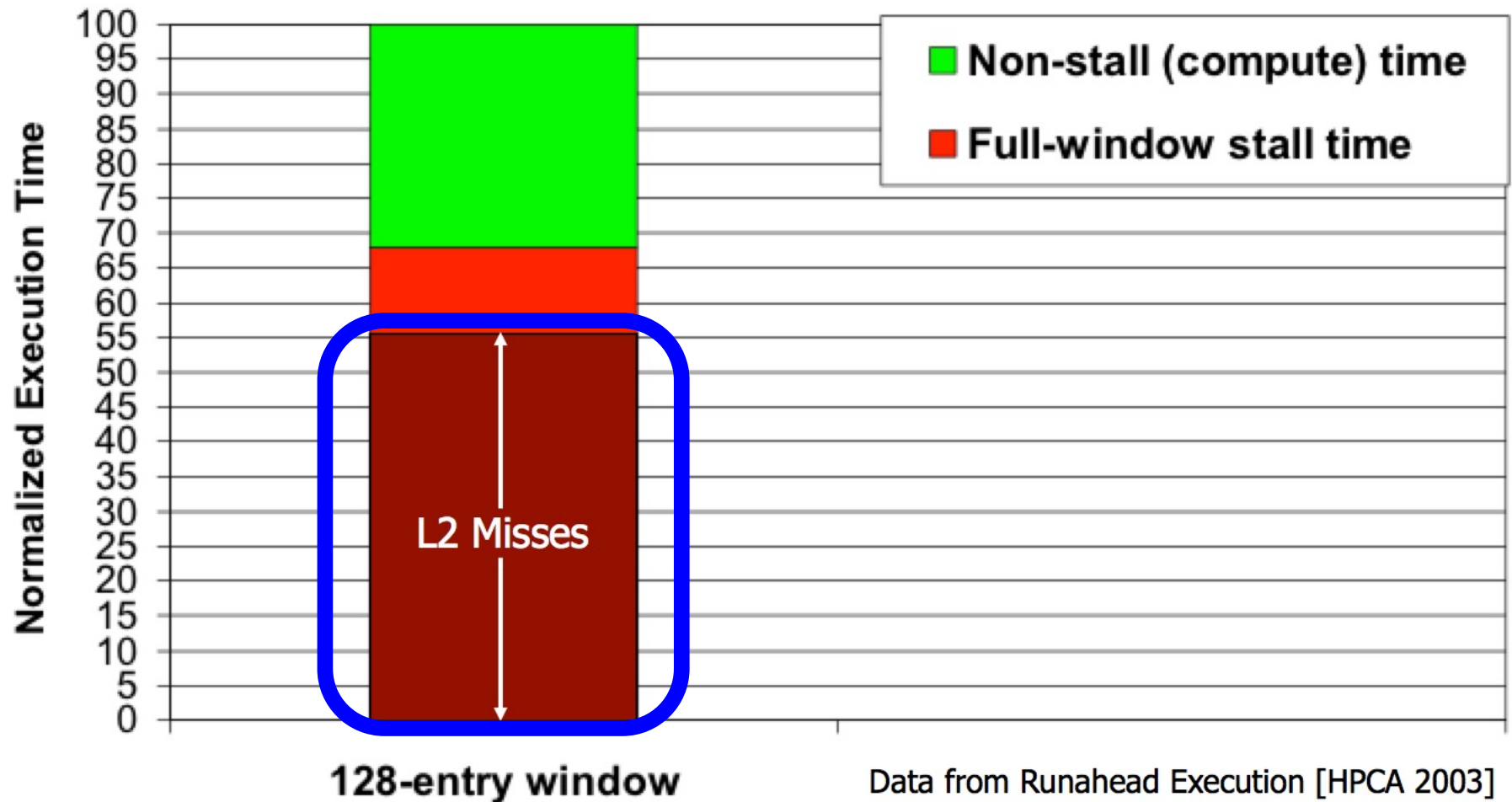
When we started the Alpha architecture design in 1988, we estimated a 25-year lifetime and a relatively modest 32% per year compounded performance improvement of implementations over that lifetime (1,000× total). We guestimated about 10× would come from CPU clock improvement, 10× from multiple instruction issue, and 10× from multiple processors.

5, 1996  MICROPROCESSOR REPORT

I expect that over the coming decade memory subsystem design will be the *only* important design issue for microprocessors.



# The Performance Perspective



# The Performance Perspective

---

- Onur Mutlu, Jared Stark, Chris Wilkerson, and Yale N. Patt,  
**"Runahead Execution: An Alternative to Very Large Instruction Windows for Out-of-order Processors"**  
*Proceedings of the 9th International Symposium on High-Performance Computer Architecture (HPCA)*, pages 129-140, Anaheim, CA, February 2003. [Slides \(pdf\)](#)  
***One of the 15 computer arch. papers of 2003 selected as Top Picks by IEEE Micro. HPCA Test of Time Award (awarded in 2021).***

## Runahead Execution: An Alternative to Very Large Instruction Windows for Out-of-order Processors

Onur Mutlu §    Jared Stark †    Chris Wilkerson ‡    Yale N. Patt §

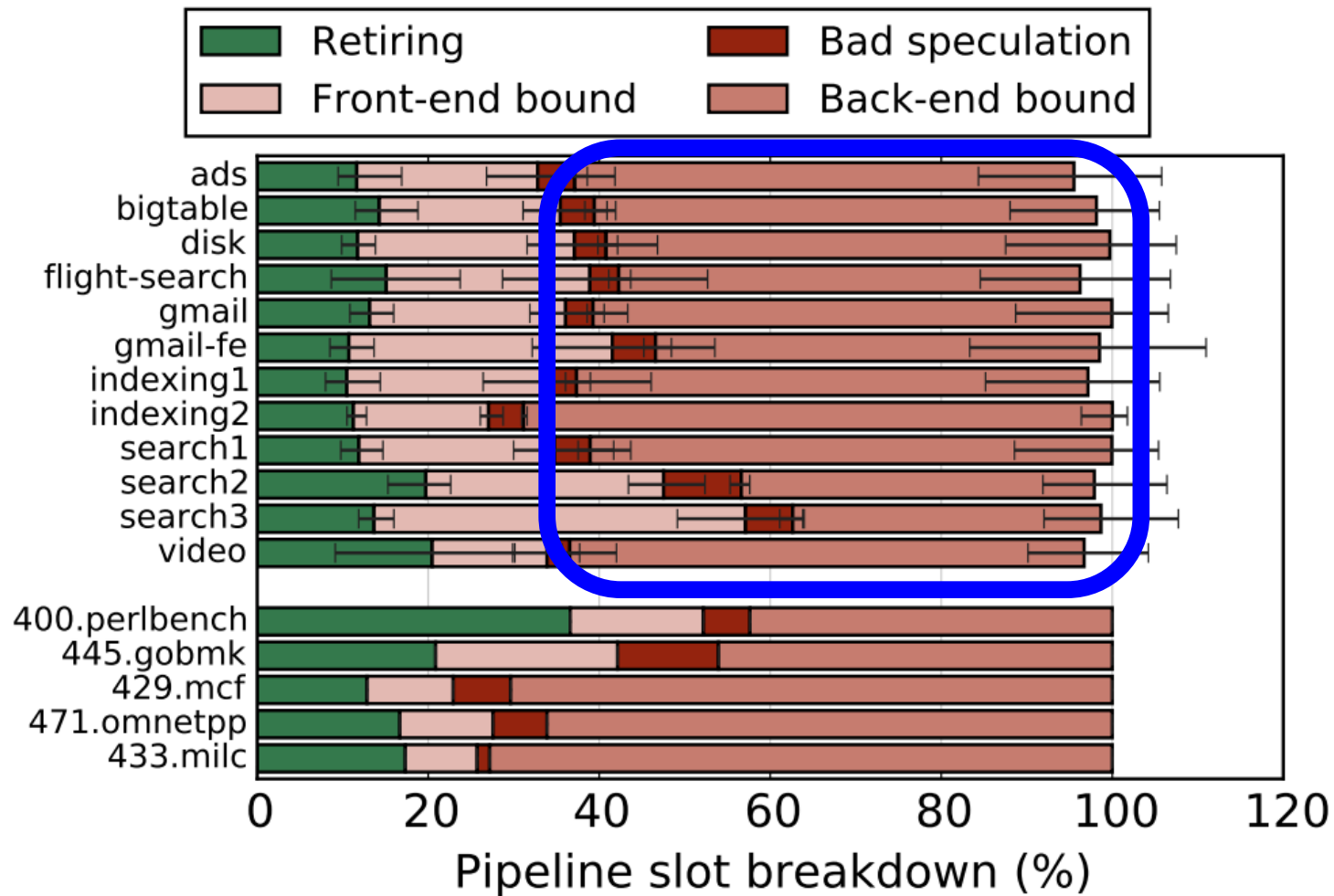
§ECE Department  
The University of Texas at Austin  
{onur,patt}@ece.utexas.edu

†Microprocessor Research  
Intel Labs  
jared.w.stark@intel.com

‡Desktop Platforms Group  
Intel Corporation  
chris.wilkerson@intel.com

# The Performance Perspective (Today)

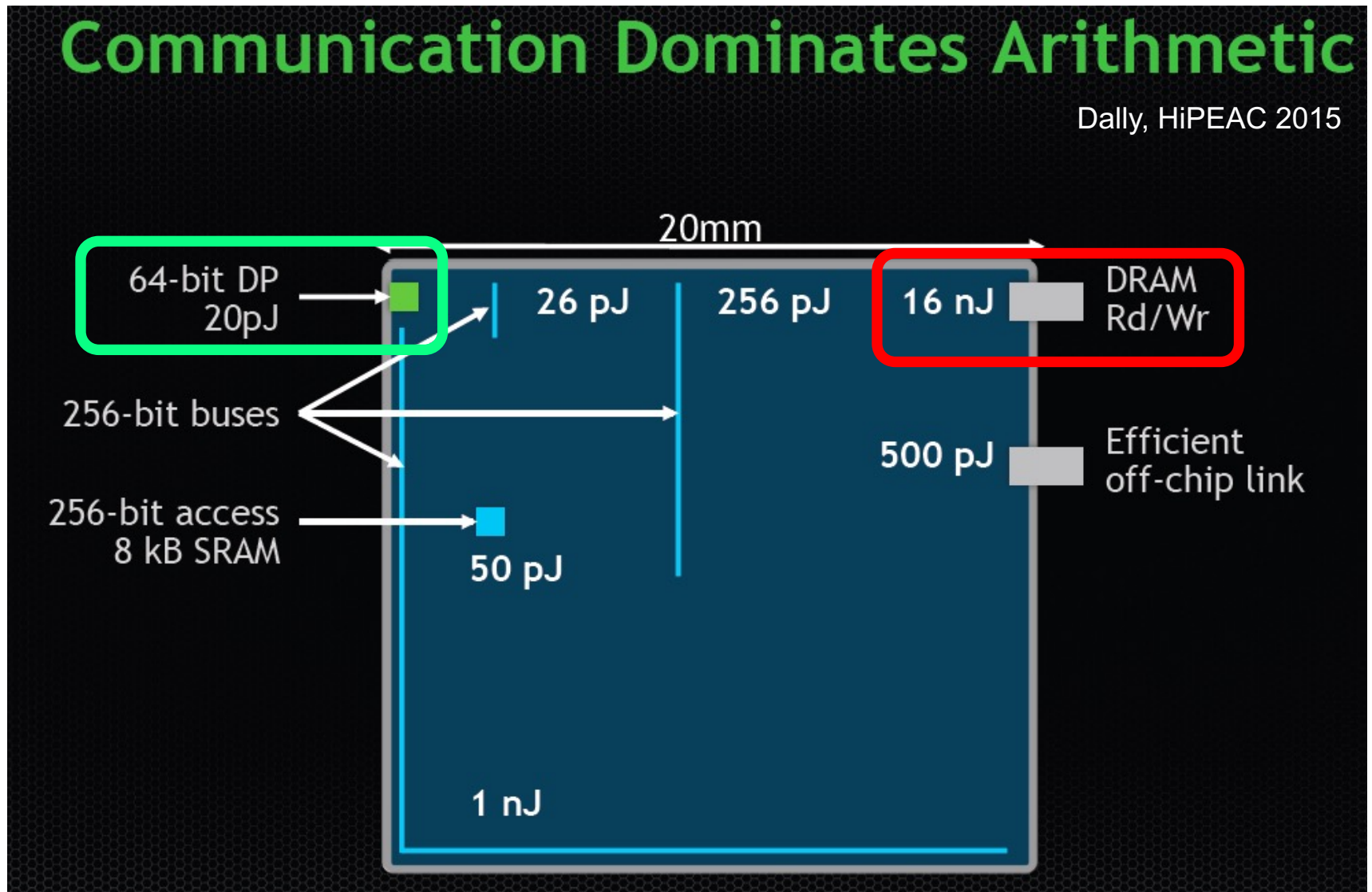
- All of Google's Data Center Workloads (2015):



# The Energy Perspective

## Communication Dominates Arithmetic

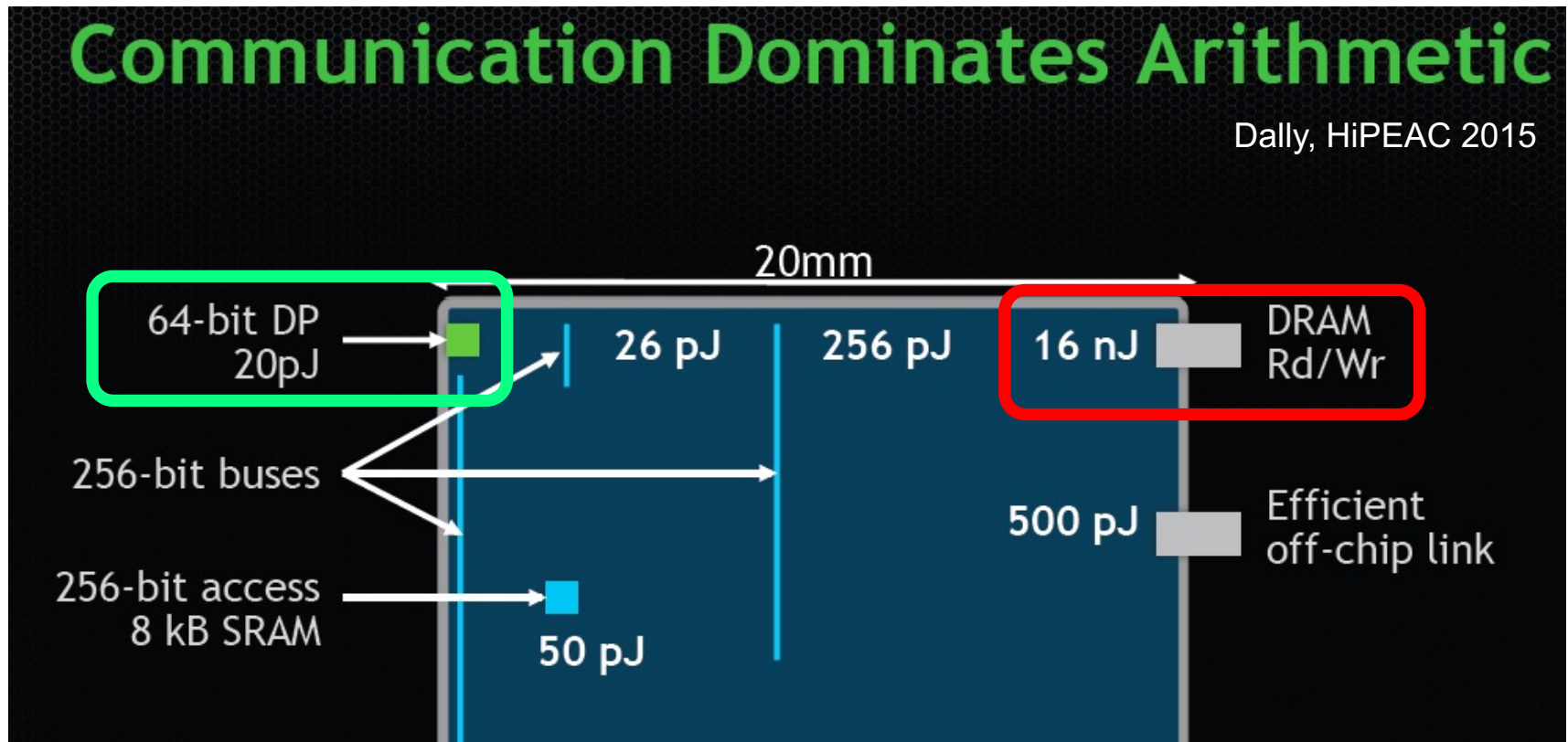
Dally, HiPEAC 2015



# Data Movement vs. Computation Energy

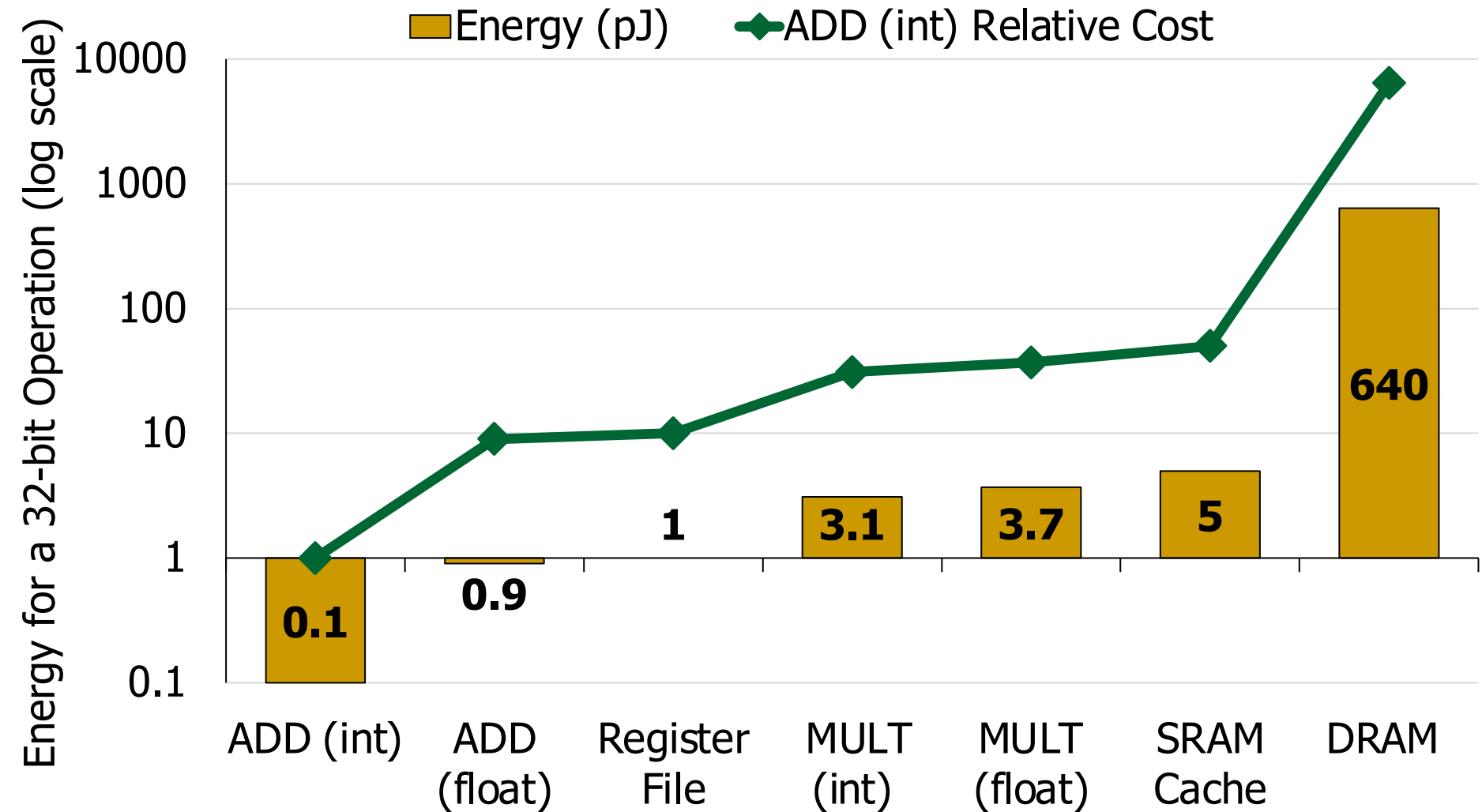
## Communication Dominates Arithmetic

Dally, HiPEAC 2015



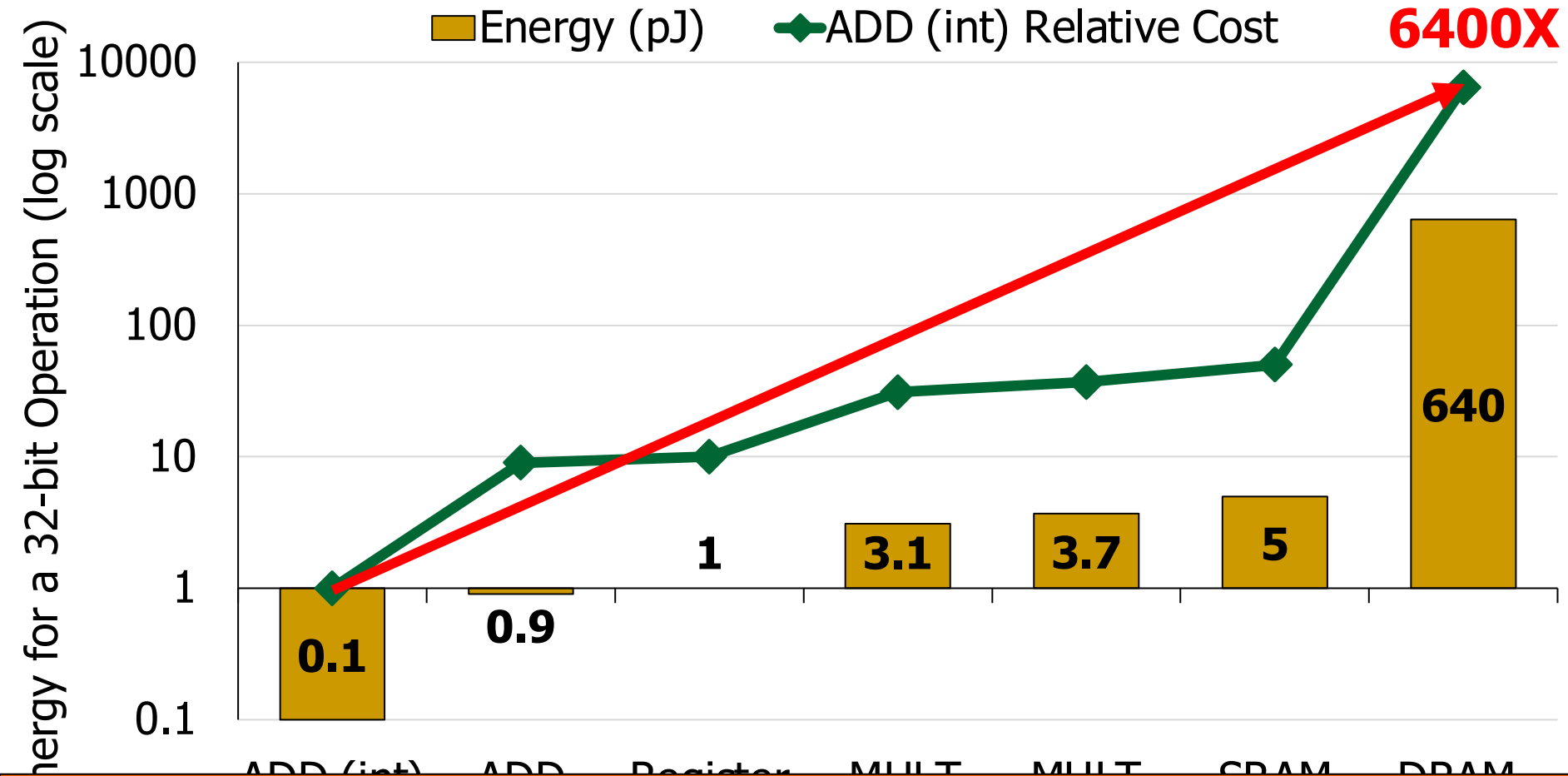
A memory access consumes  $\sim 100$ - $1000\times$  the energy of a complex addition

# Data Movement vs. Computation Energy





# Data Movement vs. Computation Energy



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

# Energy Waste in 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.

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

## Google Workloads for Consumer Devices: Mitigating Data Movement Bottlenecks

Amirali Boroumand<sup>1</sup>

Saugata Ghose<sup>1</sup>

Youngsok Kim<sup>2</sup>

Rachata Ausavarungnirun<sup>1</sup>

Eric Shiu<sup>3</sup>

Rahul Thakur<sup>3</sup>

Daehyun Kim<sup>4,3</sup>

Aki Kuusela<sup>3</sup>

Allan Knies<sup>3</sup>

Parthasarathy Ranganathan<sup>3</sup>

Onur Mutlu<sup>5,1</sup>

# Energy Waste in Accelerators

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

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

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

Amirali Boroumand<sup>†◇</sup>  
Geraldo F. Oliveira<sup>\*</sup>

Saugata Ghose<sup>‡</sup>  
Xiaoyu Ma<sup>§</sup>

Berkin Akin<sup>§</sup>  
Eric Shiu<sup>§</sup>

Ravi Narayanaswami<sup>§</sup>  
Onur Mutlu<sup>\*†</sup>

<sup>†</sup>Carnegie Mellon Univ.

<sup>◇</sup>Stanford Univ.

<sup>‡</sup>Univ. of Illinois Urbana-Champaign

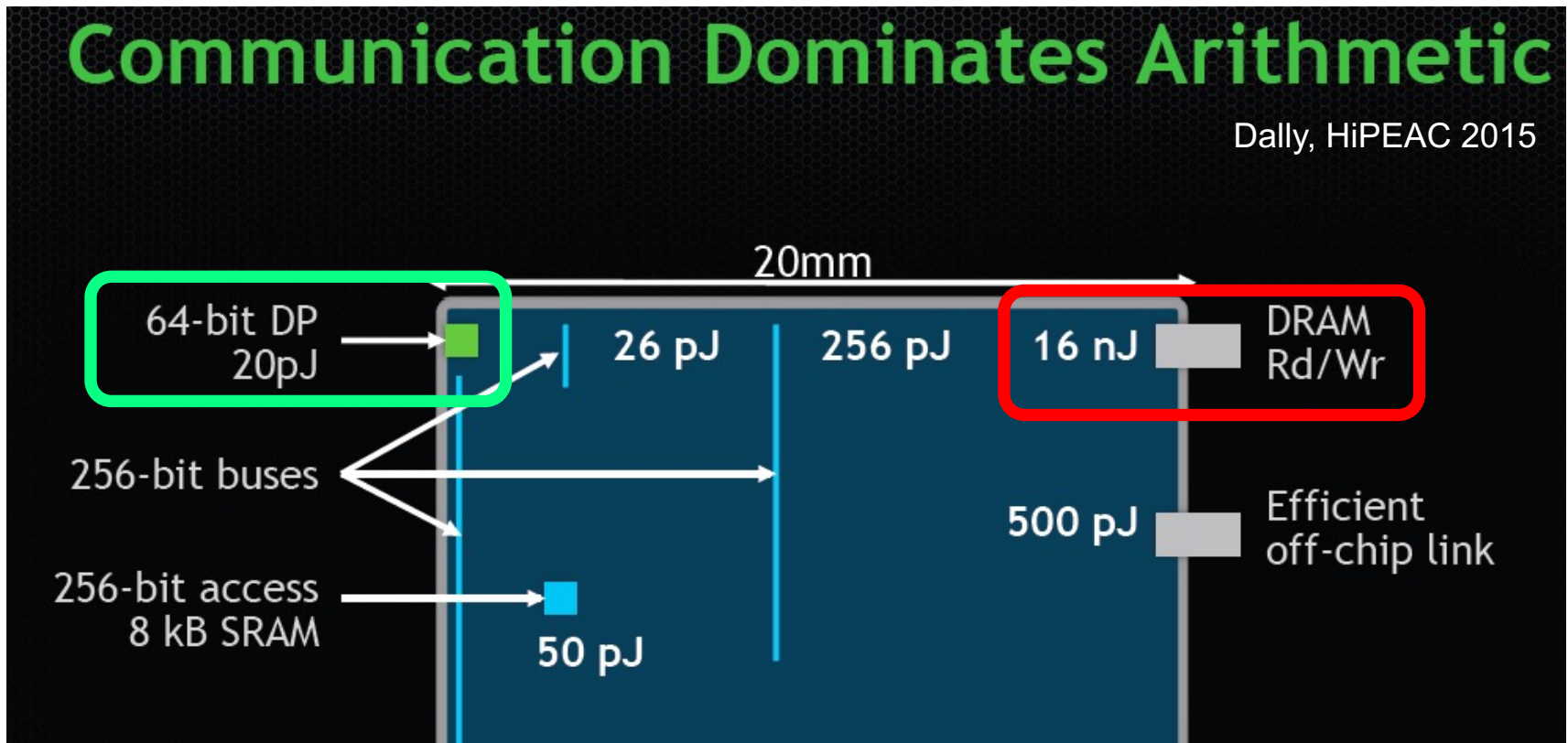
<sup>§</sup>Google

<sup>\*</sup>ETH Zürich

# We Do Not Want to Move Data!

## Communication Dominates Arithmetic

Dally, HiPEAC 2015



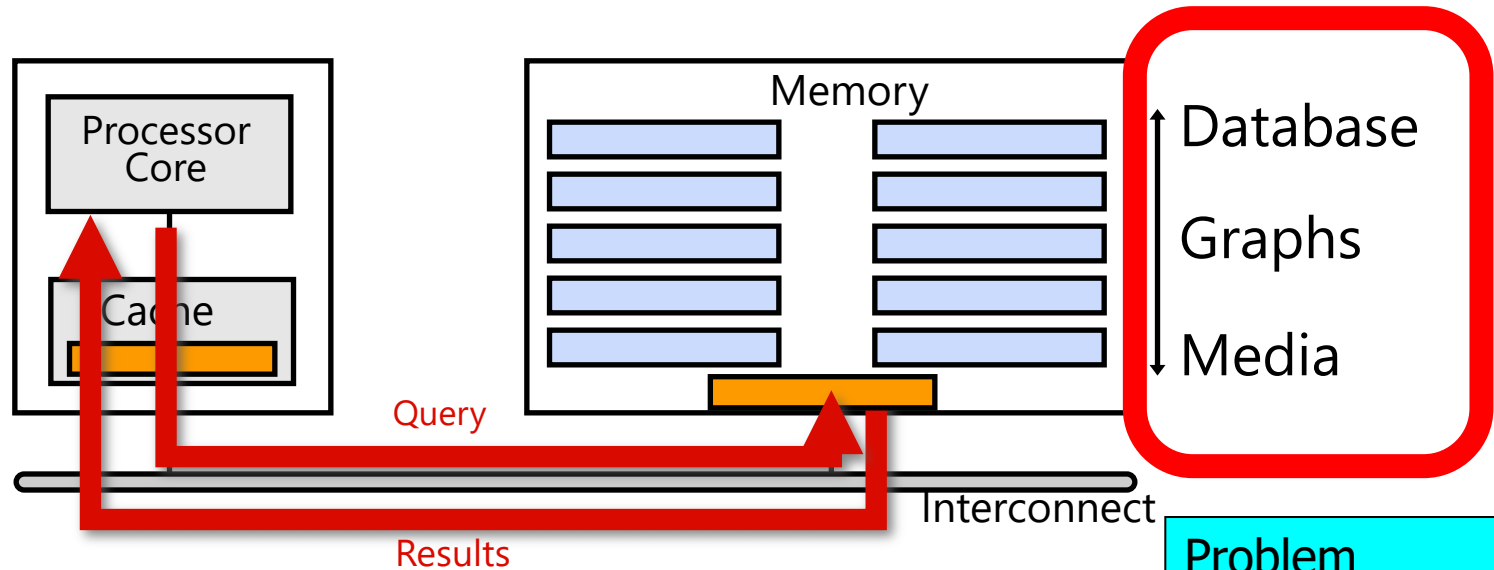
A memory access consumes  $\sim 100-1000X$  the energy of a complex addition

# We Need A Paradigm Shift To ...

---

- Enable computation with minimal data movement
- Compute where it makes sense (where data resides)
- Make computing architectures more data-centric

# Goal: Processing Inside Memory



- Many questions ... How do we design the:
  - ❑ compute-capable memory & controllers?
  - ❑ processors & communication units?
  - ❑ software & hardware interfaces?
  - ❑ system software, compilers, languages?
  - ❑ algorithms & theoretical foundations?

|                    |
|--------------------|
| Problem            |
| Algorithm          |
| Program/Language   |
| System Software    |
| SW/HW Interface    |
| Micro-architecture |
| Logic              |
| Devices            |
| Electrons          |



# PIM Review and Open Problems

---

## A Modern Primer on Processing in Memory

Onur Mutlu<sup>a,b</sup>, Saugata Ghose<sup>b,c</sup>, Juan Gómez-Luna<sup>a</sup>, Rachata Ausavarungnirun<sup>d</sup>

*SAFARI Research Group*

<sup>a</sup>*ETH Zürich*

<sup>b</sup>*Carnegie Mellon University*

<sup>c</sup>*University of Illinois at Urbana-Champaign*

<sup>d</sup>*King Mongkut's University of Technology North Bangkok*

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

**"A Modern Primer on Processing in Memory"**

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

# PIM Course (Fall 2022)

## ■ Fall 2022 Edition:

- [https://safari.ethz.ch/projects\\_and\\_seminars/fall2022/doku.php?id=processing\\_in\\_memory](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](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):

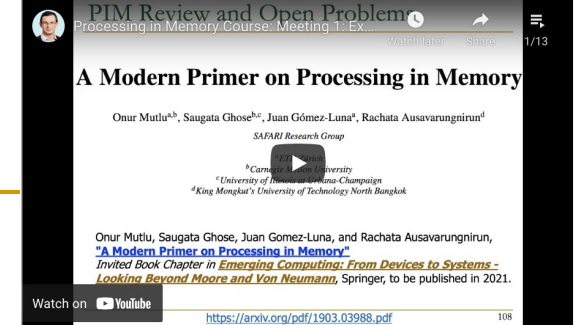
- <https://www.youtube.com/watch?v=9e4Chnwdovo&list=PL5Q2soXY2Zi-841fUYUK9EsXKhQKRPyX>

## ■ 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>

**SAFARI**

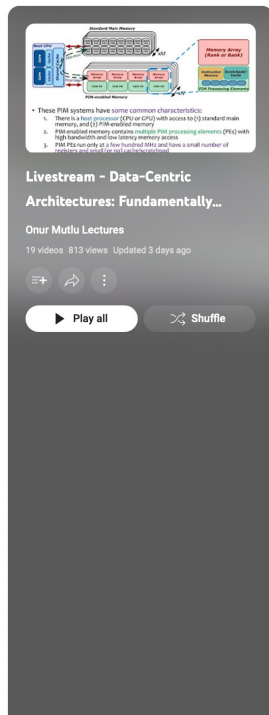


### Spring 2022 Meetings/Schedule

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

# Processing-in-Memory Course (Spring 2023)

- Short weekly lectures
- Hands-on projects



- PIM Course: Lecture 1: Data-Centric Architectures: Improving Performance & Energy (Spring 2023)**  
Onur Mutlu Lectures • 1.1K views • Streamed 3 months ago  
1:14:16
- PIM Course: Lecture 2: How to Evaluate Data Movement Bottlenecks (Spring 2023)**  
Onur Mutlu Lectures • 332 views • 2 months ago  
16:37
- ASPLOS 2023 Tutorial: Real-world Processing-in-Memory Systems for Modern Workloads**  
Onur Mutlu Lectures • 1.5K views • Streamed 2 months ago  
6:27:39
- PIM Course: Lecture 3: Real-world PIM: UPMEM PIM (Spring 2023)**  
Onur Mutlu Lectures • 411 views • 2 months ago  
19:43
- PIM Course: Lecture 4: Real-world PIM: Microbenchmarking of UPMEM PIM (Spring 2023)**  
Onur Mutlu Lectures • 188 views • 2 months ago  
24:10
- Análisis Experimental de una Arquitectura PIM - Juan Gómez Luna - Lecture in Spanish @ U. de Córdoba**  
Onur Mutlu Lectures • 169 views • 2 months ago  
2:27:12
- PIM Course: Lecture 5: Real-world PIM: Samsung HBM-PIM (Spring 2023)**  
Onur Mutlu Lectures • 483 views • 2 months ago  
24:08
- PIM Course: Lecture 6: Real-world PIM: SK Hynix AIM (Spring 2023)**  
Onur Mutlu Lectures • 573 views • 1 month ago  
35:50
- PIM Course: Lecture 7: Real-world PIM: Samsung AxDIMM (Spring 2023)**  
Onur Mutlu Lectures • 325 views • 1 month ago  
21:32

[https://www.youtube.com/playlist?list=PL5Q2soXY2zi\\_EObuoAZVSq\\_o6UySWQHvz](https://www.youtube.com/playlist?list=PL5Q2soXY2zi_EObuoAZVSq_o6UySWQHvz)

**SAFARI Project & Seminars Courses**  
(Spring 2023)

Search

Recent Changes Media Manager Sitemap

Trace: • heterogeneous\_systems • processing\_in\_memory

Home

Courses

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

processing\_in\_memory

**Data-Centric Architectures: Fundamentally Improving Performance and Energy (227-0085-37L)**

Course Description

Data movement between the memory units and the compute units of current computing systems is a major performance and energy bottleneck. From large-scale servers to mobile devices, data movement costs dominate computation costs in terms of both performance and energy consumption. For example, data movement between the main memory and the processing cores accounts for 62% of the total system energy in consumer applications. As a result, the data movement bottleneck is a huge burden that greatly limits the energy efficiency and performance of modern computing systems. This phenomenon is an undesired effect of the dichotomy between memory and the processor, which leads to the data movement bottleneck.

Many modern and important workloads such as machine learning, computational biology, graph processing, databases, video analytics, and real-time data analytics suffer greatly from the data movement bottleneck. These workloads are exemplified by irregular memory accesses, relatively low data reuse, low cache line utilization, low arithmetic intensity (i.e., ratio of operations per accessed byte), and large datasets that greatly exceed the main memory size. The computation in these workloads cannot usually compensate for the data movement costs. In order to alleviate this data movement bottleneck, we need a paradigm shift from the traditional processor-centric design, where all computation takes place in the compute units, to a more data-centric design where processing elements are placed closer to or inside where the data resides. This paradigm of computing is known as Processing-in-Memory (PIM).

This is your perfect P&S if you want to become familiar with the main PIM technologies, which represent “the next big thing” in Computer Architecture. You will work hands-on with the first real-world PIM architecture, will explore different PIM architecture designs for important workloads, and will develop tools to enable research of future PIM systems. Projects in this course span software and hardware as well as the software/hardware interface. You can potentially work on developing and optimizing new workloads for the first real-world PIM hardware or explore new PIM designs in simulators, or do something else that can forward our understanding of the PIM paradigm.

Table of Contents

- Data-Centric Architectures: Fundamentally Improving Performance and Energy (227-0085-37L)
- Course Description
- Mentors
- Lecture Video Playlist on YouTube
- Spring 2023 Meetings/Schedule
- Past Lecture Video Playlists on YouTube
- Learning Materials
- Assignments

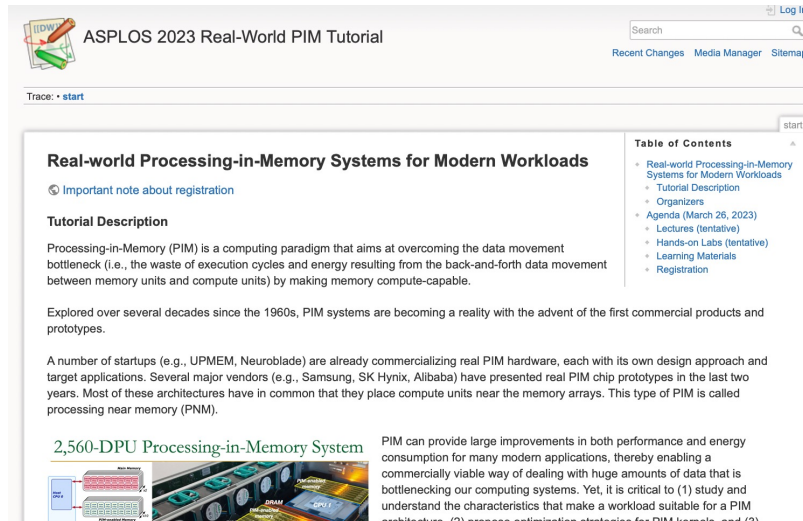
Prerequisites of the course:

- Digital Design and Computer Architecture (or equivalent course).
- Familiarity with C/C++ programming.
- Interest in future computer architectures and computing paradigms.
- Interest in discovering why things do or do not work and solving problems
- Interest in making systems efficient and usable

[https://safari.ethz.ch/projects\\_and\\_seminars/spring2023/doku.php?id=processing\\_in\\_memory](https://safari.ethz.ch/projects_and_seminars/spring2023/doku.php?id=processing_in_memory)

# Real PIM Tutorial (ASPLOS 2023)

## ■ March 26: Lectures + Hands-on labs + Invited talks



ASPLOS 2023 Real-World PIM Tutorial

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

Table of Contents

- Real-world Processing-in-Memory Systems for Modern Workloads
- Tutorial Description
- Organizers
- Agenda (March 26, 2023)
- Lectures (tentative)
- Hands-on Labs (tentative)
- Learning Materials
- Registration

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

Important note about registration

**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.

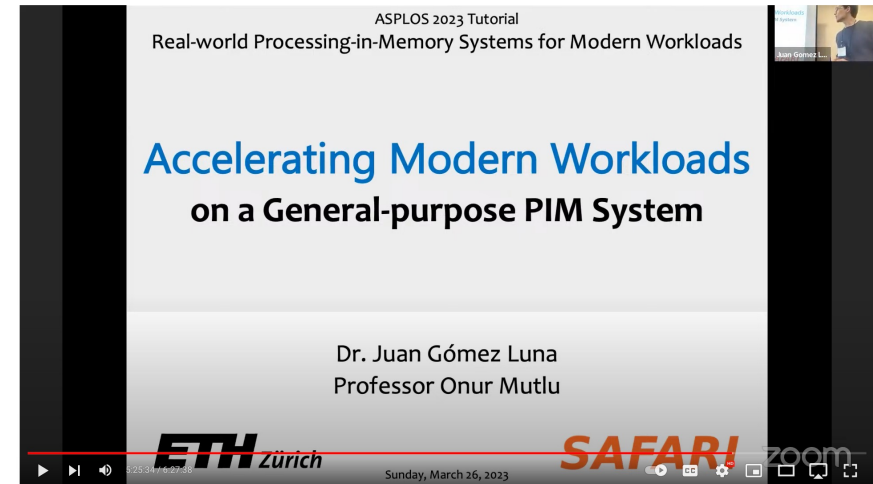
A number of startups (e.g., UPMEM, Neuroblade) are already commercializing real PIM hardware, each with its own design approach and target applications. Several major vendors (e.g., Samsung, SK Hynix, Alibaba) have presented real PIM chip prototypes in the last two years. Most of these architectures 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) assess realistic strategies for PIM kernels, and (3)

### Tutorial Materials

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



ASPLOS 2023 Tutorial

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

**Accelerating Modern Workloads on a General-purpose PIM System**

Dr. Juan Gómez Luna  
Professor Onur Mutlu

ETH Zürich SAFARI

Sunday, March 26, 2023

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

Onur Mutlu Lectures  
32.1K subscribers

Subscribed

33

Share

Clip

Save

...

views Streamed 7 days ago Livestream - Data-Centric Architectures: Fundamentally Improving Performance and Energy (Spring 2023)

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

<https://events.safari.ethz.ch/asplos-...>

<https://www.youtube.com/watch?v=oYCaLcT0Kmo>

<https://events.safari.ethz.ch/asplos-pim-tutorial/>



# Current Real PIM Tutorial (ISCA 2023)

## ■ June 18: Lectures + Hands-on labs + Invited talks



### ISCA 2023 Real-World PIM Tutorial

Search

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

Trace: • [start](#)

## 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.

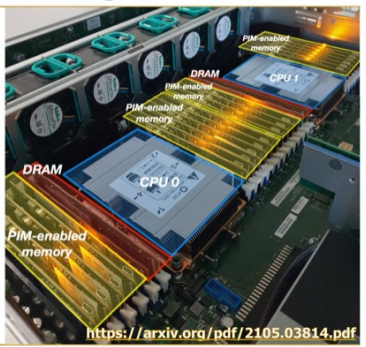
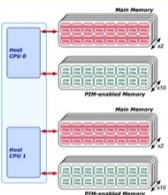
Explored over several decades since the 1960s, PIM systems are becoming a reality with the advent of the first commercial products and prototypes.

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).

### Table of Contents

- [Real-world Processing-in-Memory Systems for Modern Workloads](#)
- [Tutorial Description](#)
- [Organizers](#)
- [Agenda \(June 18, 2023\)](#)
- [Lectures \(tentative\)](#)
- [Hands-on Labs \(tentative\)](#)
- [Learning Materials](#)

### 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.

<https://events.safari.ethz.ch/isca-pim-tutorial/>

# End of Backup Slides