

Tutorial on Memory-Centric Computing: Introduction

Geraldo F. Oliveira

<https://geraldofojunior.github.io>

MICRO 2024

02 November 2024

Brief Self Introduction



■ Geraldo F. Oliveira

- Researcher @ SAFARI Research Group since November 2017
- Soon, I will defend my PhD thesis, advised by Onur Mutlu
- **I am in the job market for industry positions**
- <https://geraldofojunior.github.io/>
- geraldofojunior@gmail.com (Best way to reach me)
- <https://safari.ethz.ch>

■ Research in:



















- Computer architecture, computer systems, hardware security
- Memory and storage systems
- Hardware security, safety, predictability
- Fault tolerance
- Hardware/software cooperation
- ...

Get the Materials Online



<https://events.safari.ethz.ch/micro24-memorycentric-tutorial/doku.php?id=start>

















Tutorial Materials

Time	Speaker	Title	Materials
01:00pm-01:30pm	Prof. Onur Mutlu / Geraldo F. Oliveira	Memory-Centric Computing	 (PDF)  (PPT)
01:30pm-02:00pm	Geraldo F. Oliveira	Processing-Near-Memory Systems: Developments fro Academia & Industry	 (PDF)  (PPT)
02:00pm-02:30pm	 Dr. Brian Schwedock	Architectures and Programming Models for General-Purpose Near-Data Computing	 (PDF)  (PPT)
02:30pm-03:00pm	Geraldo F. Oliveira	Processing-Using-Memory Systems for Bulk Bitwise Operations	 (PDF)  (PPT)
03:00pm-03:30pm	N/A	Coffee Break	
03:30pm-04:00pm	Ataberk Olgun	Infrastructure for Processing-Using-Memory Research	 (PDF)  (PPT)
04:00pm-04:30pm	 Dr. Christina Giannoula	System Software and Libraries for Sparse Computational Kernels in PIM Architectures	 (PDF)  (PPT)
04:30pm-05:00pm	Nika Mansouri Ghiasi	Storage-Centric Computing for Genomics and Metagenomics	 (PDF)  (PPT)
05:00pm	Geraldo F. Oliveira	Research Challenges for PIM & Closing Remarks	 (PDF)  (PPT)

Get the Materials Online

<https://events.safari.ethz.ch/micro24-memorycentric-tutorial/doku.php?id=start>

Tutorial Materials

Time	Speaker	Materials
01:00pm-01:30pm	Prof. Onur Mutlu Oliveira	 (PDF)  (PPT)
01:30pm-02:00pm	Geraldo F. Oliveira	Academia &  (PDF)  (PPT)
02:00pm-02:30pm	 Dr. Brian Scott	purpose Near-  (PDF)  (PPT)
02:30pm-03:00pm	Geraldo F. Oliveira	operations  (PDF)  (PPT)
03:00pm-03:30pm	N/A	
03:30pm-04:00pm	Ataberk Olgun	 (PDF)  (PPT)
04:00pm-04:30pm	 Dr. Christina	al Kernels in  (PDF)  (PPT)
04:30pm-05:00pm	Nika Mansouri	omics  (PDF)  (PPT)
05:00pm	Geraldo F. Oliveira	Research Challenges for PIM & Closing Remarks  (PDF)  (PPT)



Computing

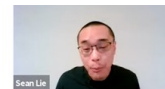
is Bottlenecked by Data

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

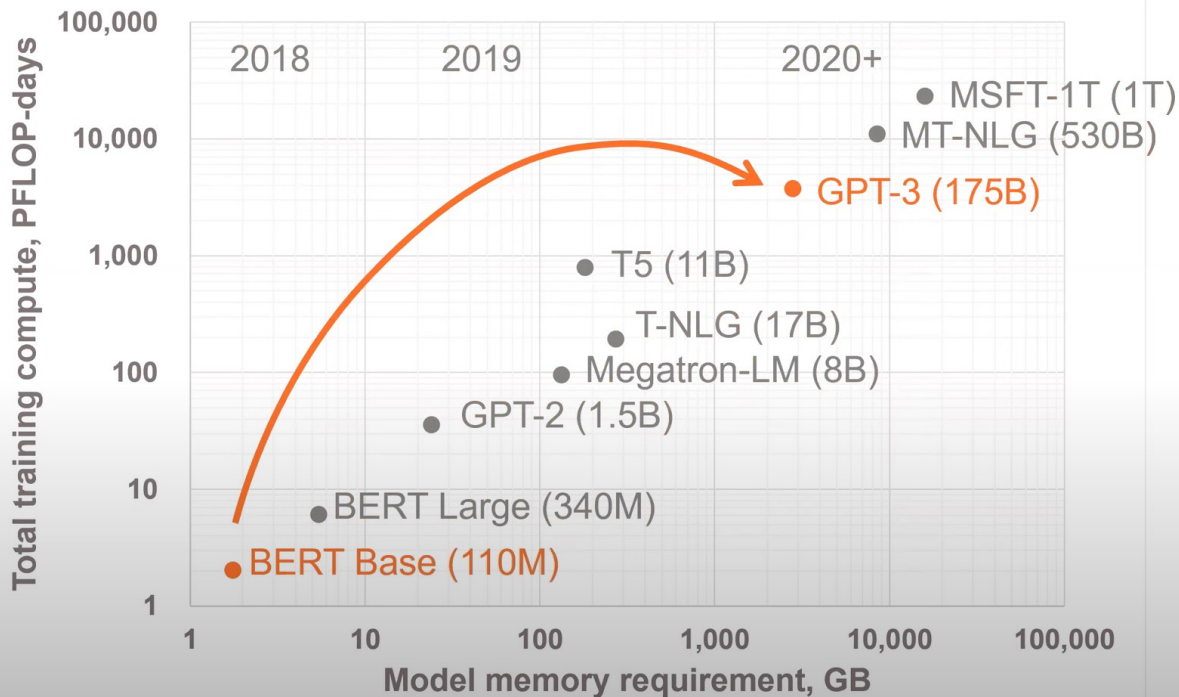
- Important workloads are all data intensive
- They require rapid and efficient processing of large amounts of data
- Data is increasing
 - We can generate more than we can process
 - We need to perform more sophisticated analyses on more data

Huge Demand for Performance & Efficiency

Exponential Growth of Neural Networks



Memory and compute requirements

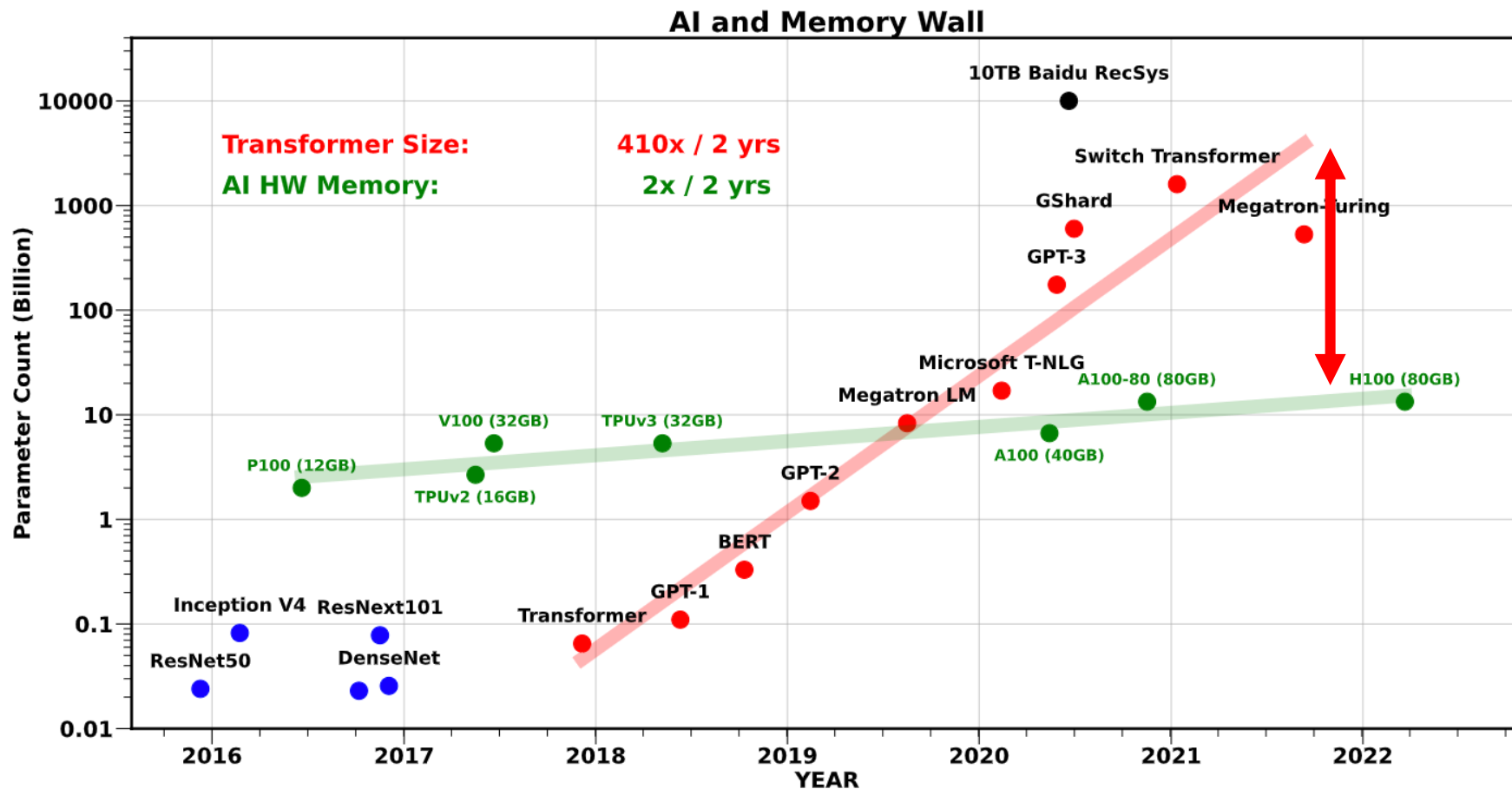


1800x more compute
In just 2 years

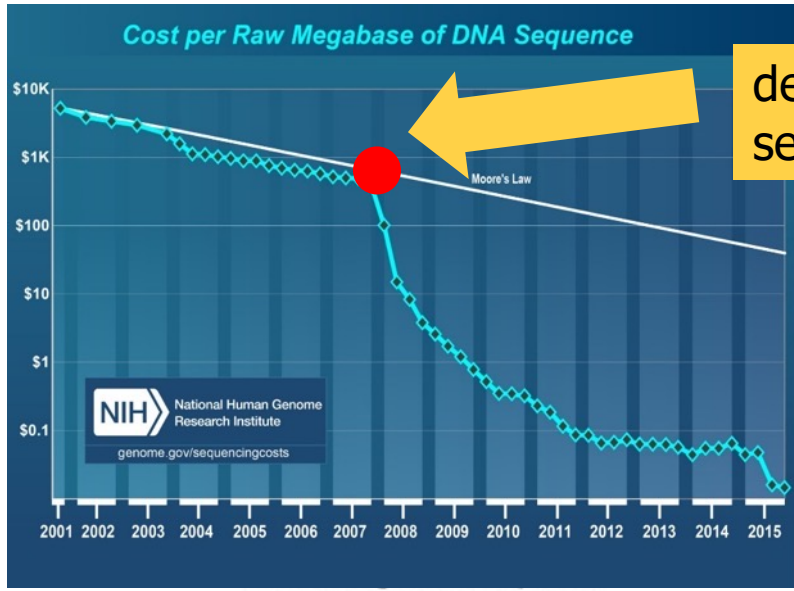
Tomorrow, **multi-trillion** parameter models

Huge Demand for Performance & Efficiency

- Gholami, Amir, et al. "[AI and Memory Wall](#)." IEEE Micro, 2024.



Huge Demand for Performance & Efficiency

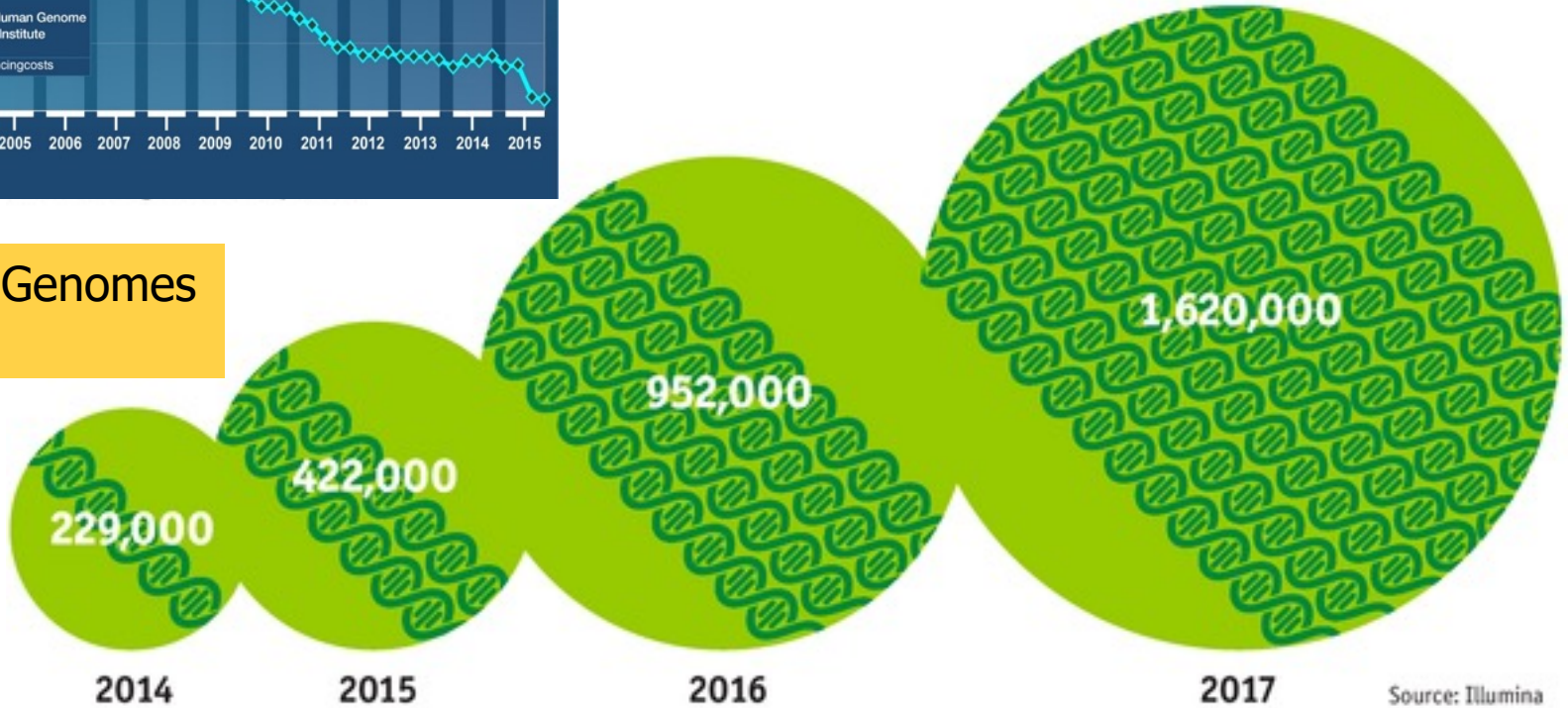


development of new sequencing technologies



Oxford Nanopore MinION

Number of Genomes Sequenced



The Economist

Source: Illumina

High Performance,

Energy Efficient,

Sustainable

(All at the Same Time)

The Problem

Data access is the major performance and energy bottleneck

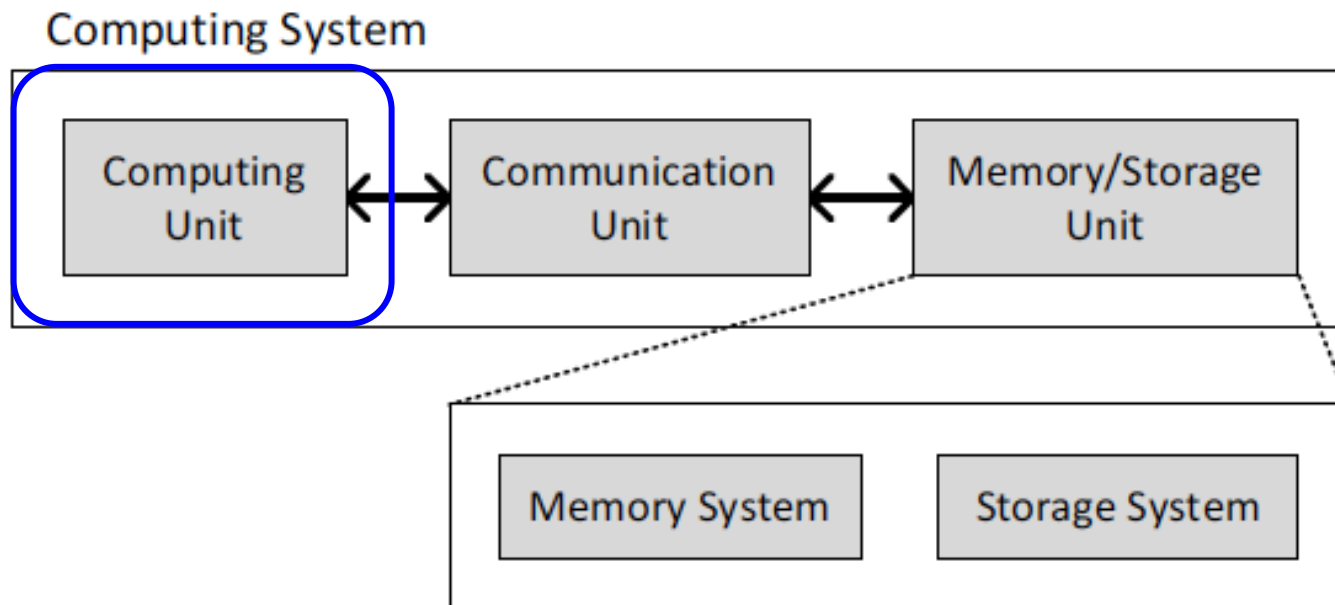
Our current
design principles
cause great energy waste
(and great performance loss)

The Problem

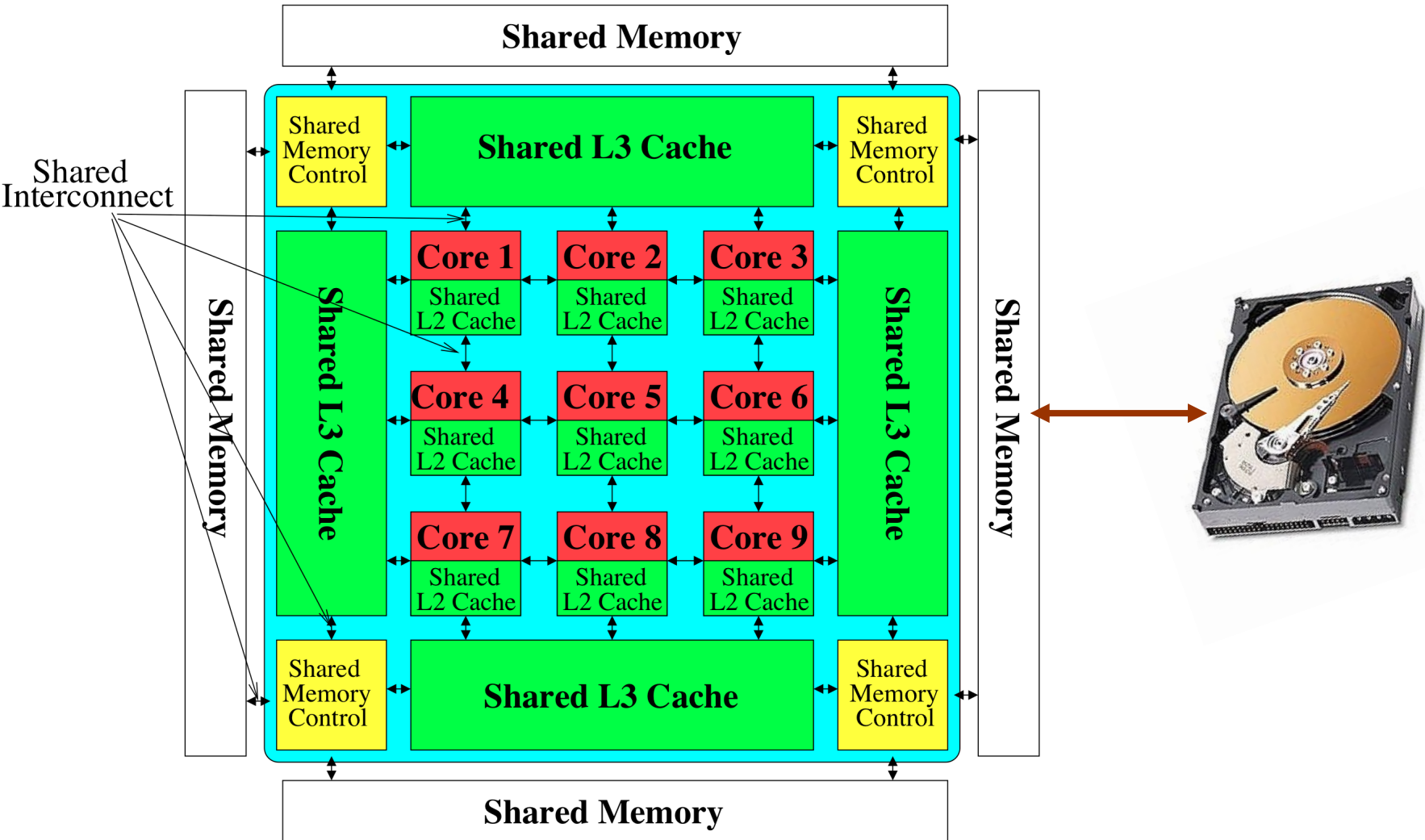
Processing of data
is performed
far away from the data

Today's Computing Systems

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



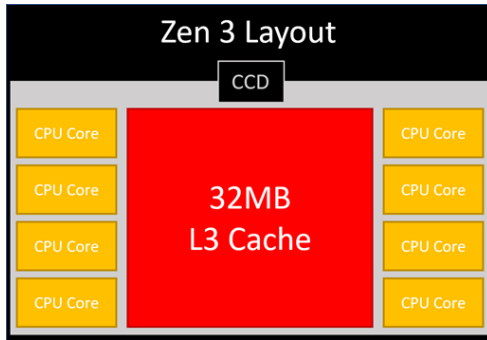
Perils of Processor-Centric Design



Most of the system is dedicated to storing and moving data

Yet, system is still bottlenecked by memory

AMD's 3D Last Level Cache (2021)

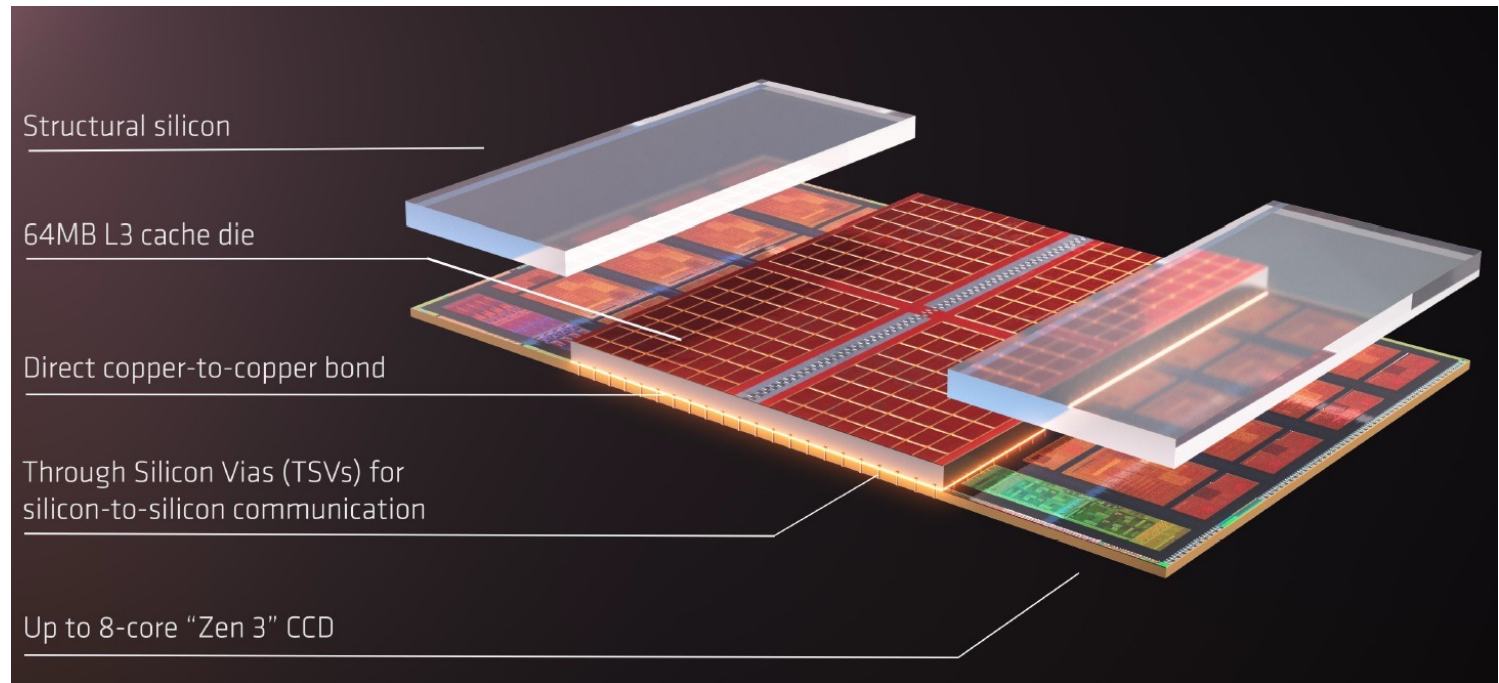


AMD increases the L3 size of their 8-core Zen 3 processors from 32 MB to 96 MB

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

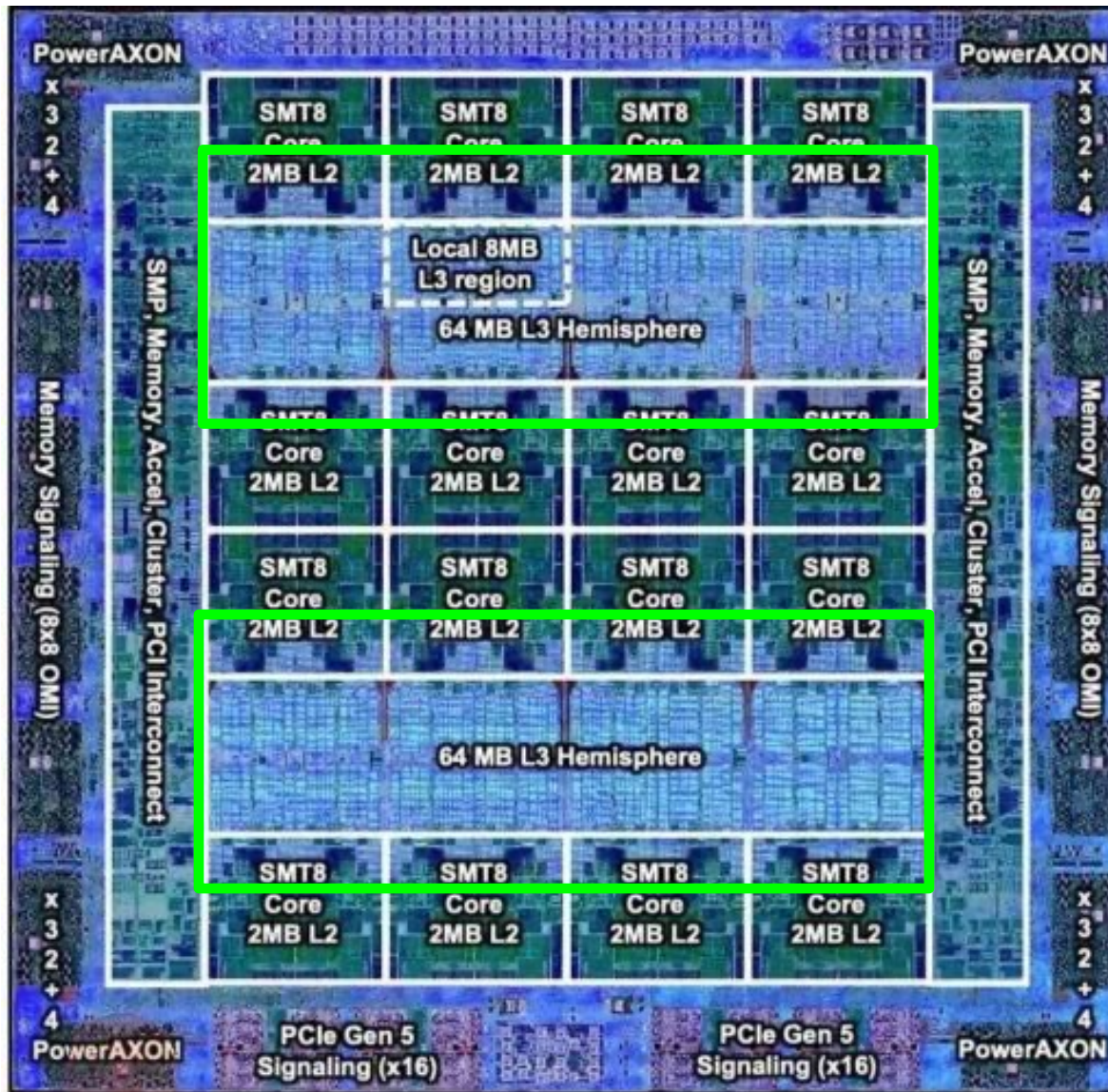
<https://community.microcenter.com/discussion/5134/comparing-zen-3-to-zen-2>



<https://youtu.be/gqAYMx34euU>

<https://www.tech-critter.com/amd-keynote-computex-2021/>

Deeper and Larger Memory Hierarchies



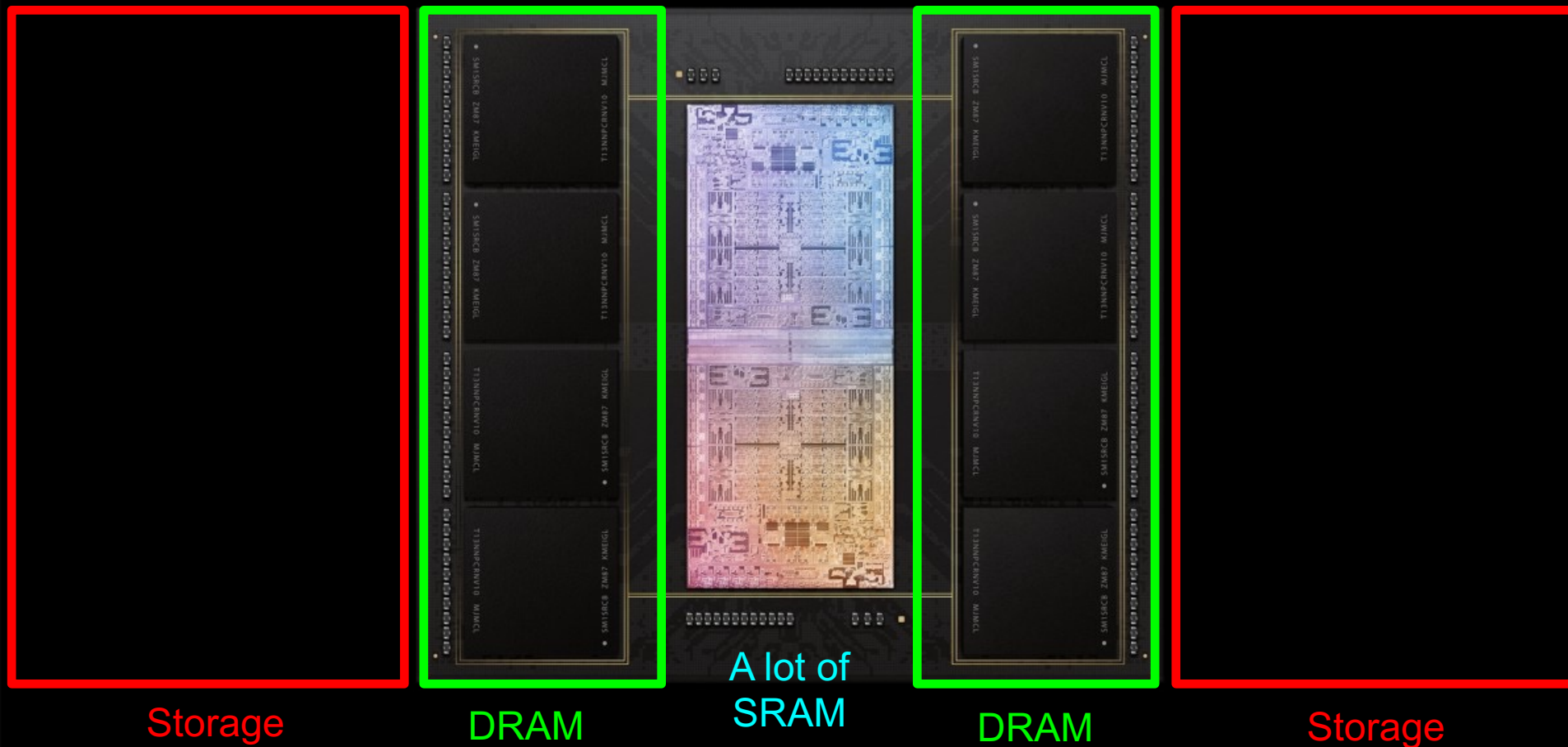
IBM POWER10,
2020

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 ...

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

It's the Memory, Stupid!

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

RICHARD SITES

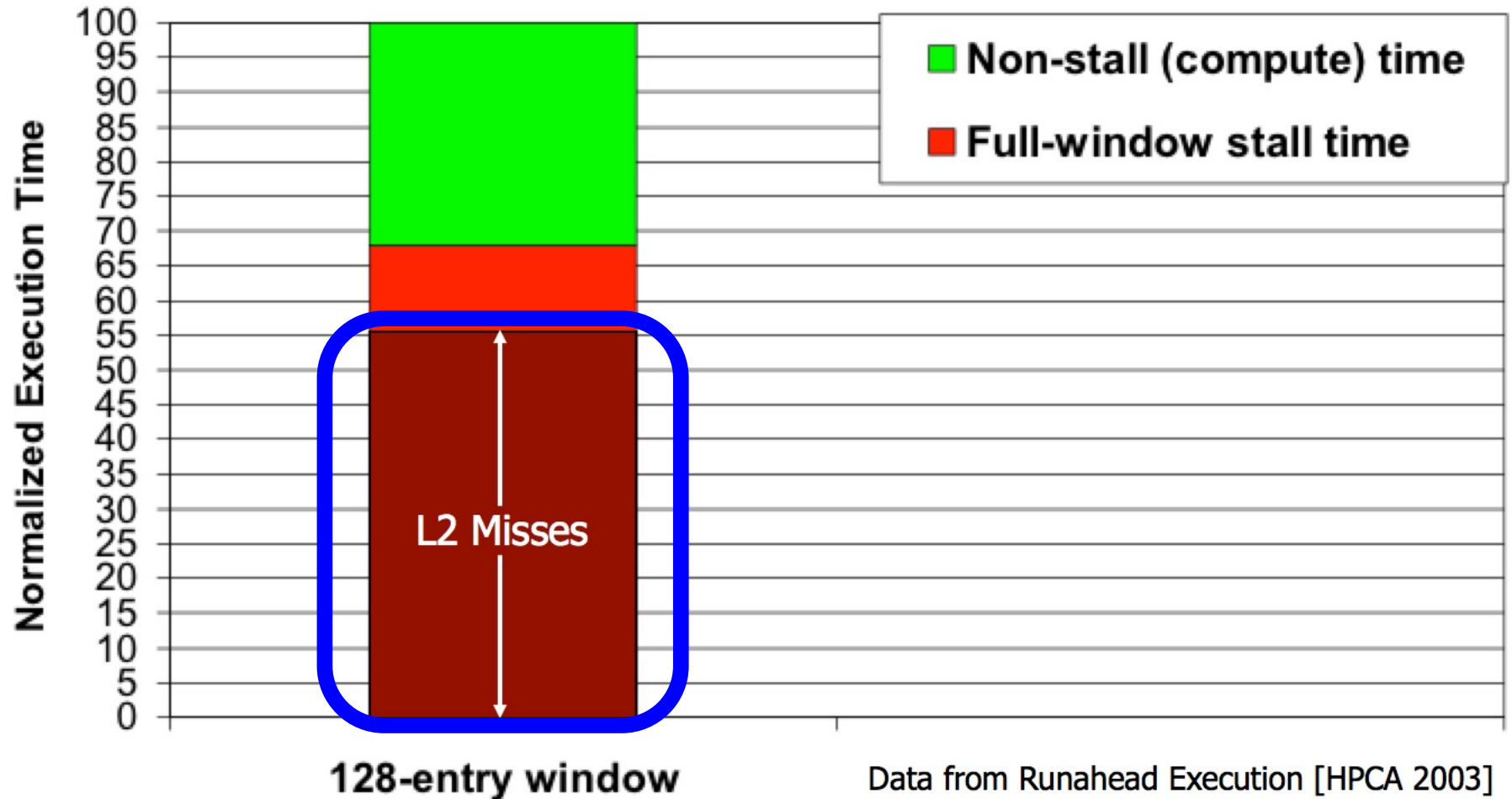
It's the Memory, Stupid!

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 guesstimated 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 §

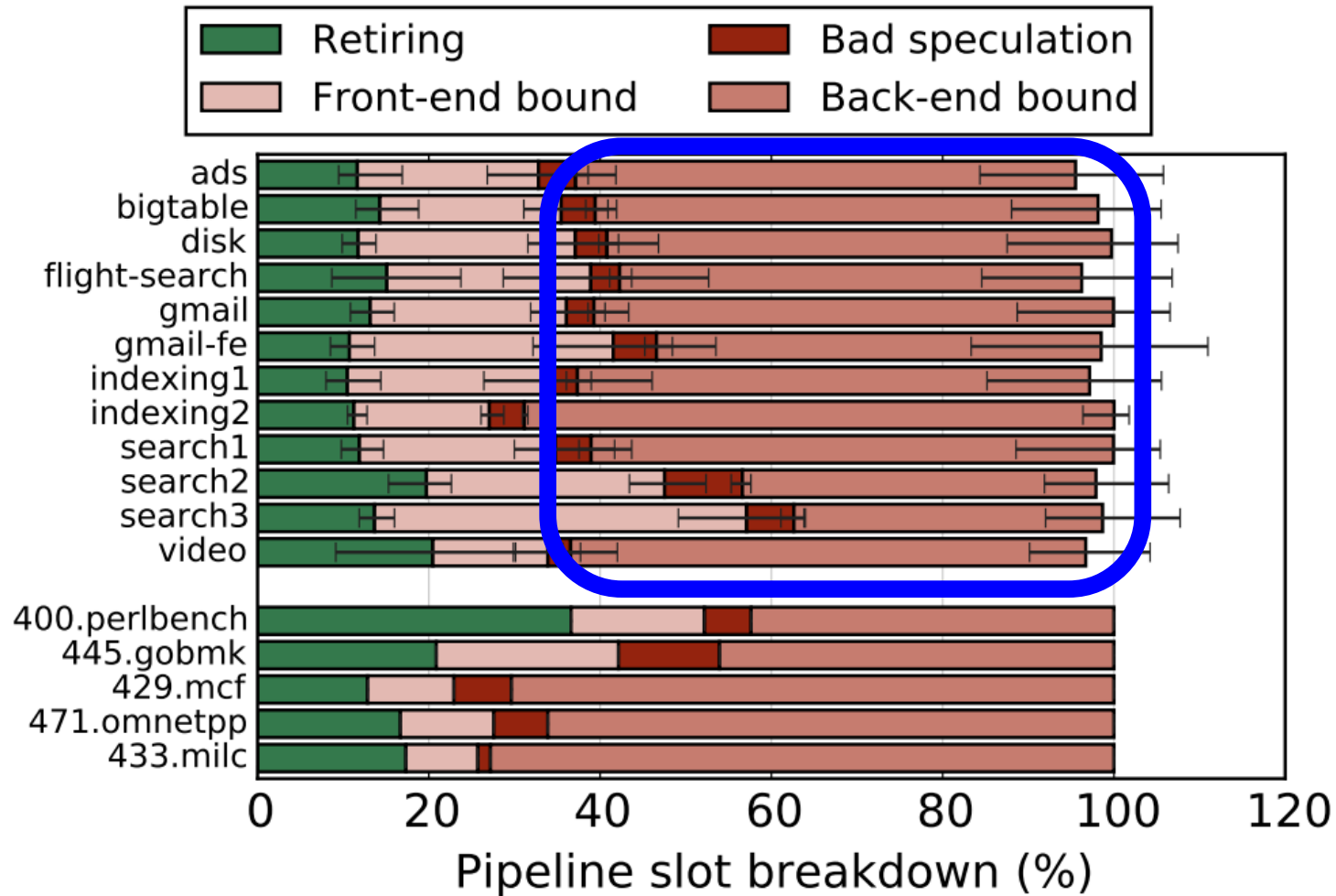
§ECE Department
The University of Texas at Austin
{onur,patt}@ece.utexas.edu

†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):



Three Key Systems Trends

1. Data access is a major bottleneck

- Applications are increasingly data hungry

2. Energy consumption is a key limiter

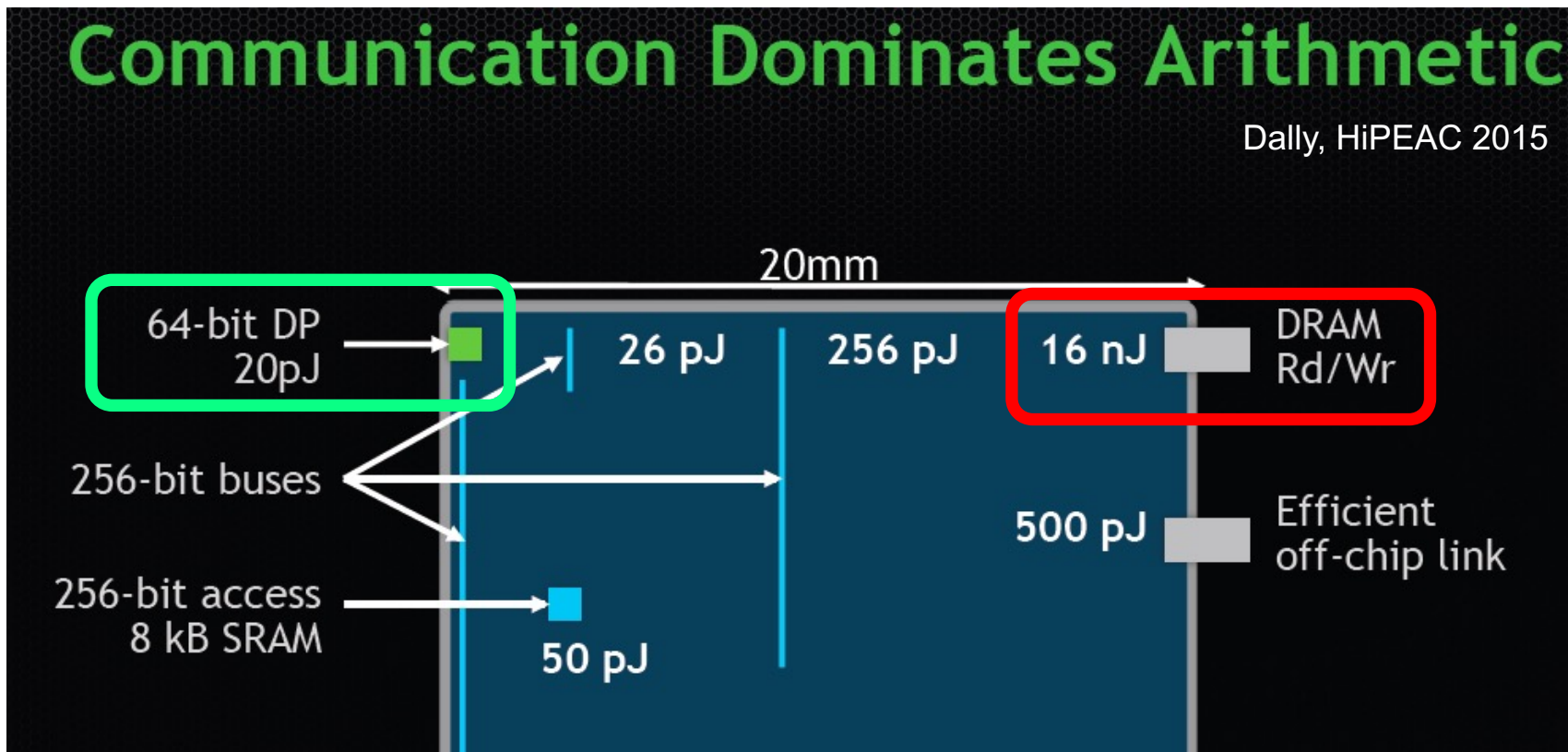
3. Data movement energy dominates compute

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

Data Movement vs. Computation Energy

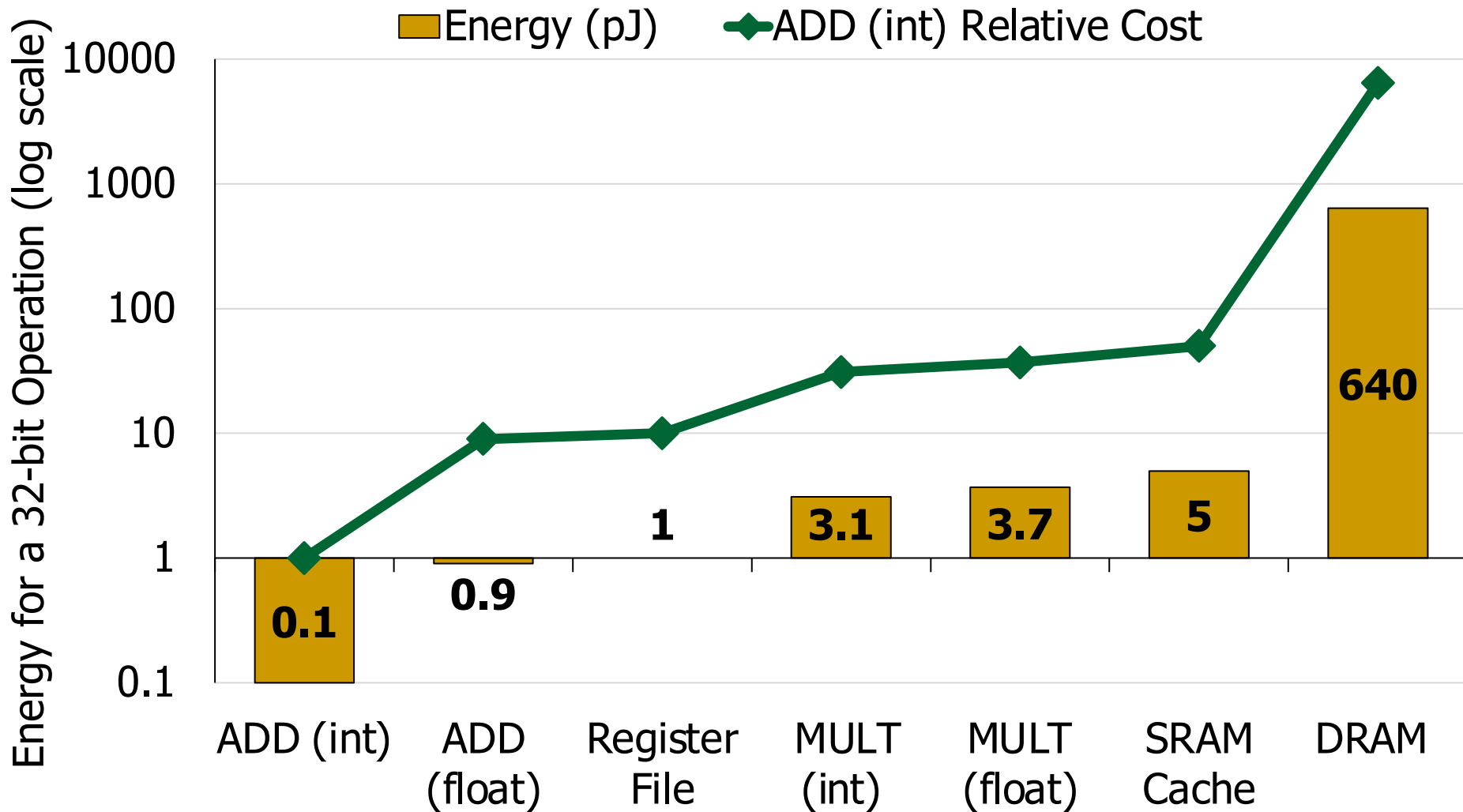
Communication Dominates Arithmetic

Dally, HiPEAC 2015

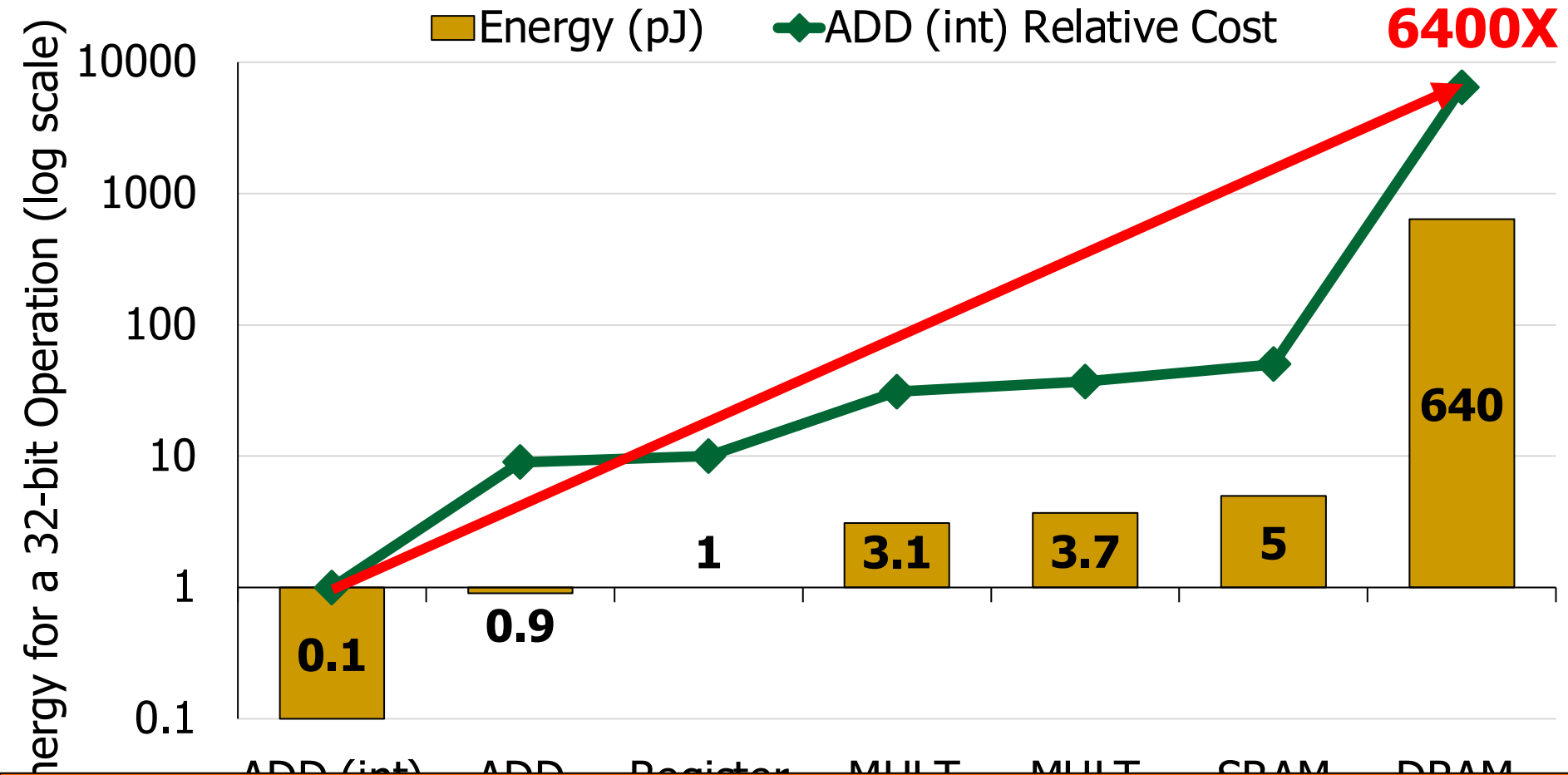


A memory access consumes $\sim 100-1000X$ the energy of a complex addition

Data Movement vs. Computation Energy



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 Boroumand¹

Saugata Ghose¹

Youngsok Kim²

Rachata Ausavarungnirun¹

Eric Shiu³

Rahul Thakur³

Daehyun Kim^{4,3}

Aki Kuusela³

Allan Knies³

Parthasarathy Ranganathan³

Onur Mutlu^{5,1}

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^{†◇}
Geraldo F. Oliveira^{*}

Saugata Ghose[‡]
Xiaoyu Ma[§]

Berkin Akin[§]
Eric Shiu[§]

Ravi Narayanaswami[§]
Onur Mutlu^{*†}

[†]Carnegie Mellon Univ.

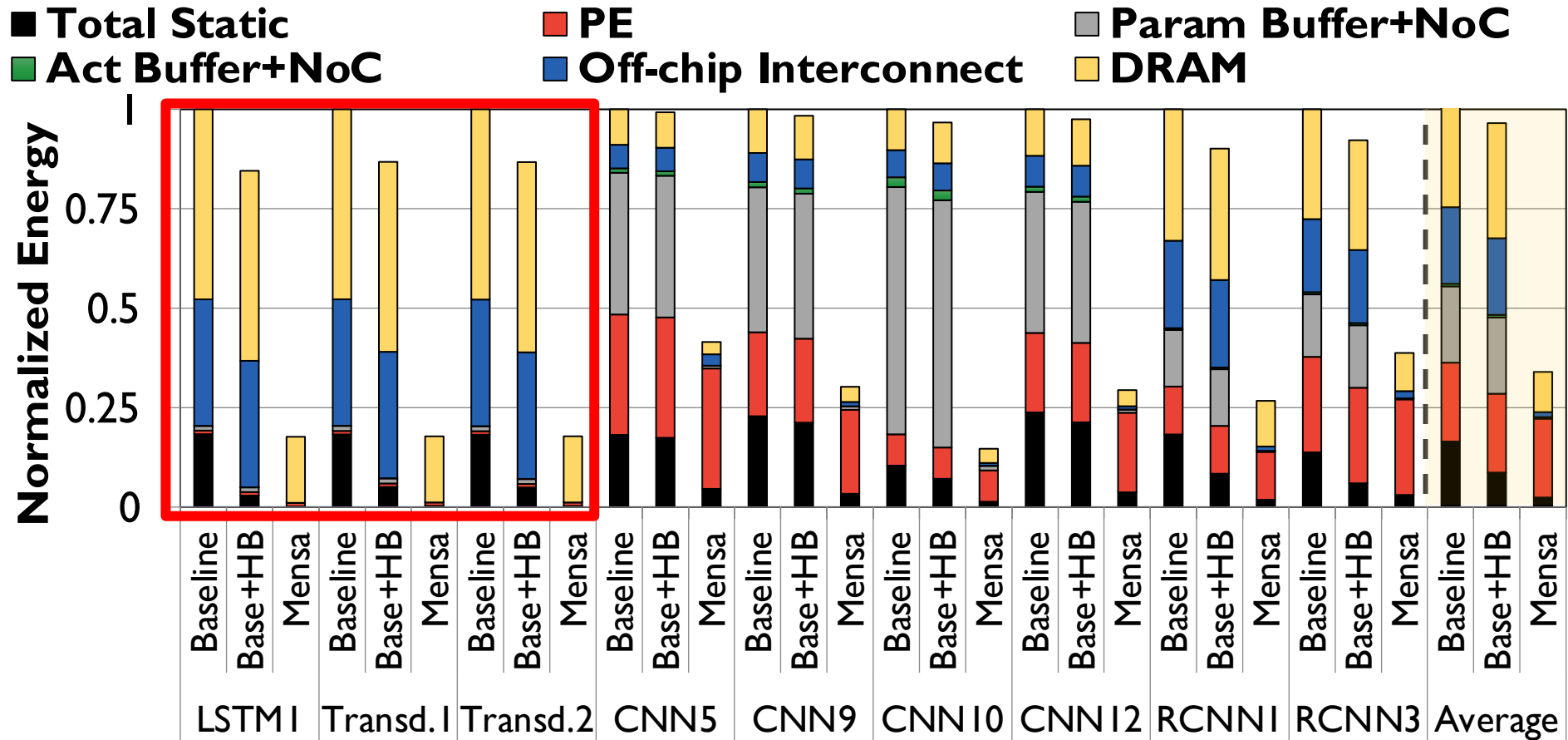
[◇]Stanford Univ.

[‡]Univ. of Illinois Urbana-Champaign

[§]Google

^{*}ETH Zürich

Example Energy Breakdowns



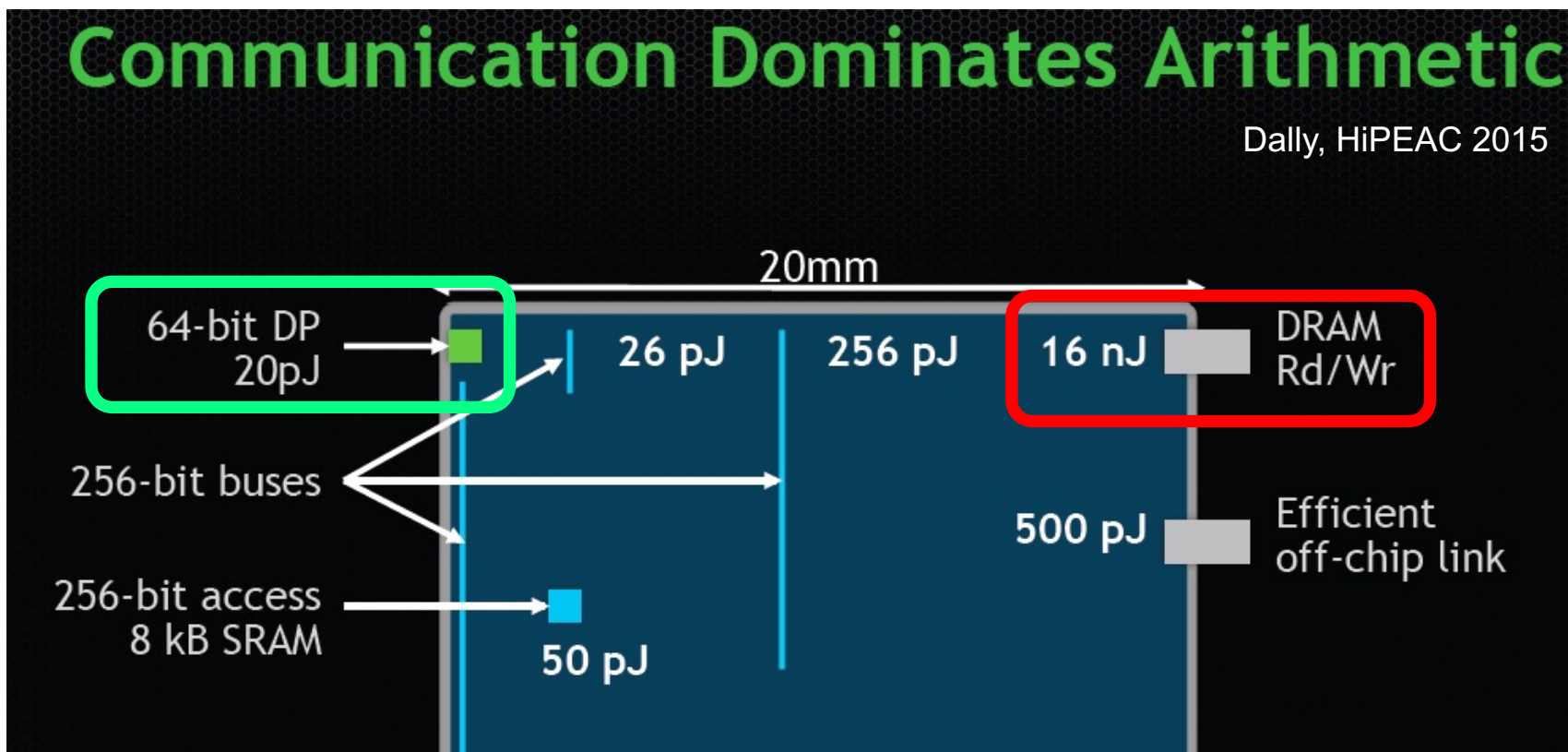
**In LSTMs and Transducers used by Google,
>90% energy spent on off-chip interconnect and DRAM**

<https://arxiv.org/pdf/2109.14320>

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

An Intelligent Architecture Handles Data Well

How to Handle Data Well

- **Ensure data does not overwhelm** the components
 - via intelligent algorithms
 - via intelligent architectures
 - via whole 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**

Architectures for Intelligent Machines

Data-centric

Data-driven

Data-aware

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

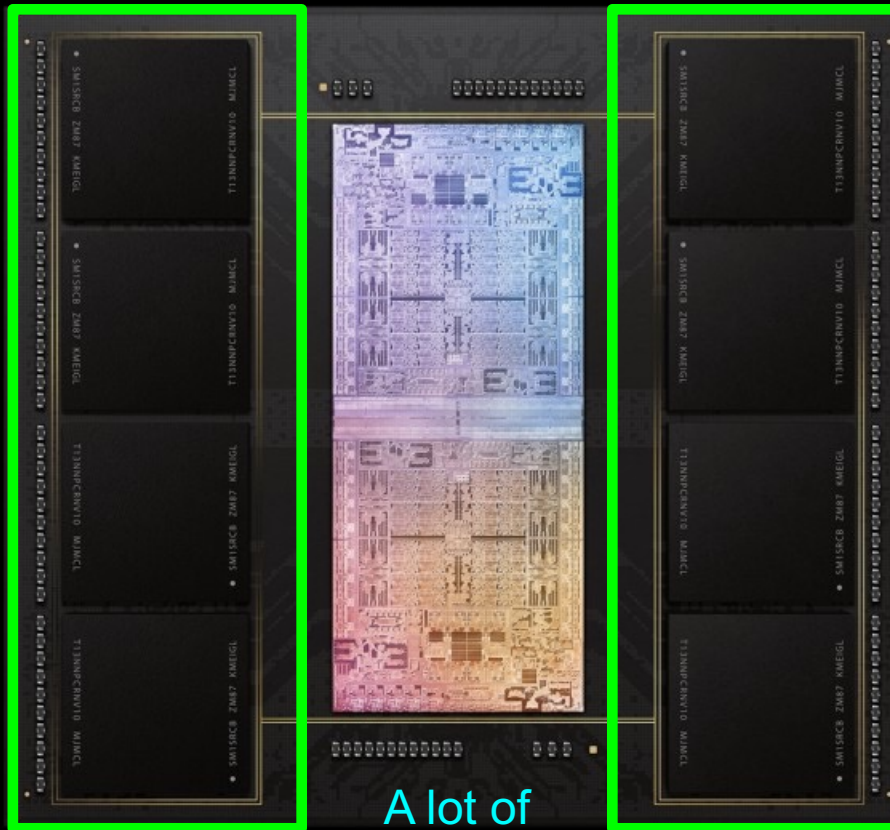
Data-Centric (Memory-Centric) Architectures

Data-Centric Architectures: Properties

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

Process Data Where It Makes Sense

Sensors



A lot of
SRAM

Storage

DRAM

DRAM

Storage

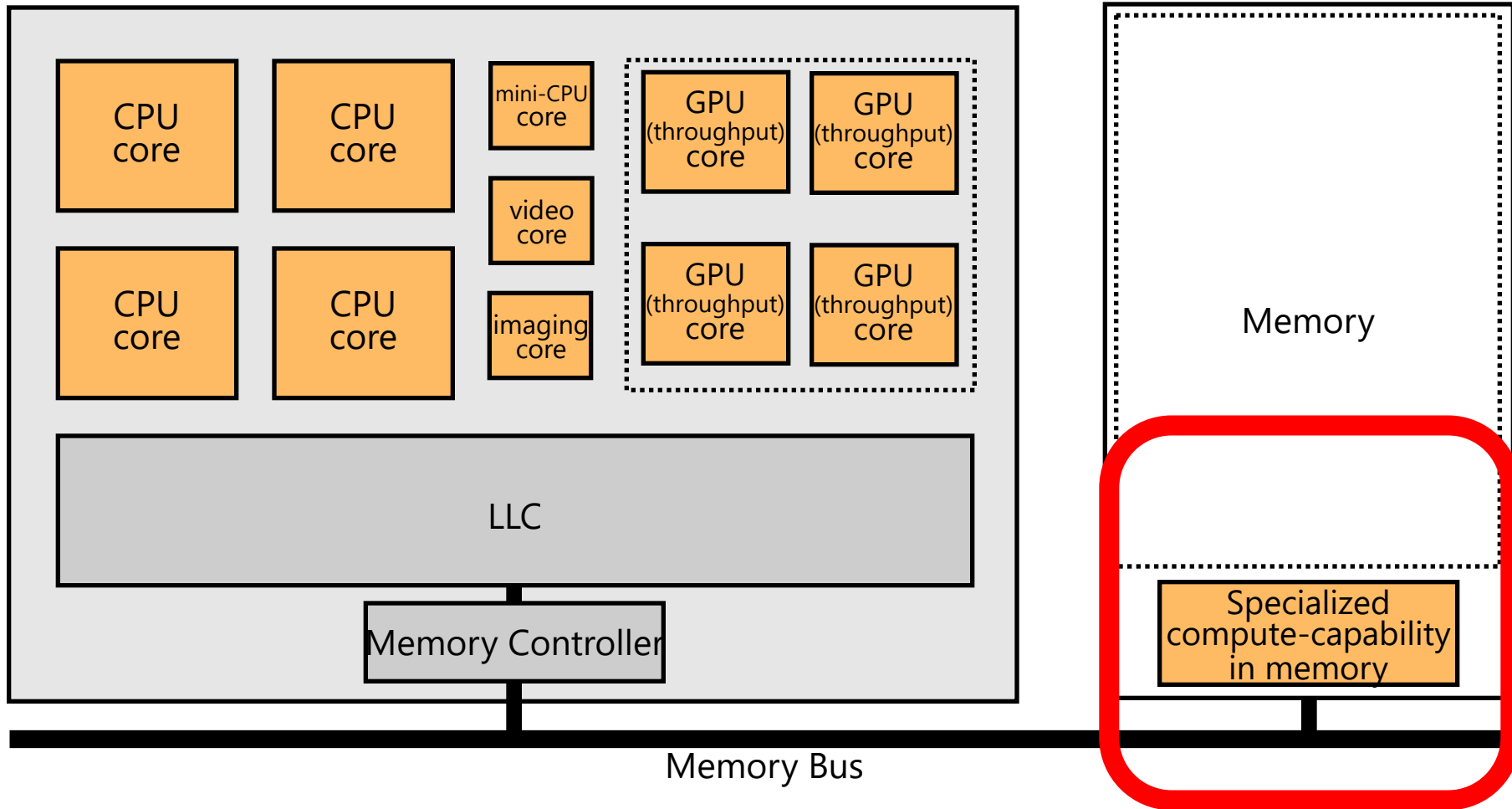
Apple M1 Ultra System (2022)

We Need to Think Differently
from the Past Approaches

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

Mindset: Memory as an Accelerator



Memory similar to a "conventional" accelerator

Processing in/near Memory: An Old Idea (I)

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

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

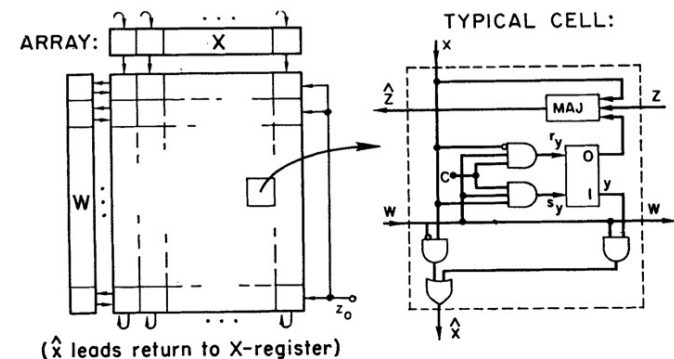
Cellular Logic-in-Memory Arrays

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

Fig. 1. Cellular sorting array I.

Processing in/near Memory: An Old Idea (II)

- 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.

Processing in/near Memory: An Old Idea (III)

- Patterson et al., “A Case for Intelligent RAM,” IEEE Micro 1997.

A CASE FOR INTELLIGENT RAM

David Patterson

Thomas Anderson

Neal Cardwell

Richard Fromm

Kimberly Keeton

Christoforos Kozyrakis

Randi Thomas

Katherine Yelick

*University of California,
Berkeley*

Two trends call into question the current practice of fabricating microprocessors and DRAMs as different chips on different fabrication lines. The gap between processor and DRAM speed is growing at 50% per year; and the size and organization of memory on a single DRAM chip is becoming awkward to use, yet size is growing at 60% per year.

Intelligent RAM, or IRAM, merges processing and memory into a single chip to lower memory latency, increase memory bandwidth, and improve energy efficiency. It also allows more flexible selection of memory size and organization, and promises savings in board area. This article reviews the state of microprocessors and DRAMs today, explores some of the opportunities and challenges for IRAMs, and finally esti-

puter designers can scale the number of memory chips independently of the number of processors. Most desktop systems have one processor and 4 to 32 DRAM chips, but most server systems have 2 to 16 processors and 32 to 256 DRAMs. Memory systems have standardized on single in-line memory module (SIMM) or dual in-line memory module (DIMM) packaging, which allow the end user to scale the amount of memory in a system.

Quantitative evidence of the industry's success is its size: In 1995, DRAMs were a \$37-billion industry, and microprocessors were a \$20-billion industry. In addition to financial success, the technologies of these industries have improved at unparalleled rates. DRAM capacity has quadrupled on average every three years since 1976, while microprocessor speed has done the same

Why In-Memory Computation Today?

- **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
- **Huge problems with Memory Technology**
 - Memory technology scaling is not going well (e.g., RowHammer)
 - Many scaling issues demand intelligence in memory
- **Designs are squeezed in the middle**

Why In-Memory Computation Today?

- **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
- **Huge problems with Memory Technology**
 - Memory technology scaling is not going well (e.g., RowHammer)
 - Many scaling issues demand intelligence in memory
- **Designs are squeezed in the middle**

Why In-Memory Computation Today?

- **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
- **Huge problems with Memory Technology**
 - Memory technology scaling is not going well (e.g., RowHammer)
 - Many scaling issues demand intelligence in memory
- **Designs are squeezed in the middle**

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/>

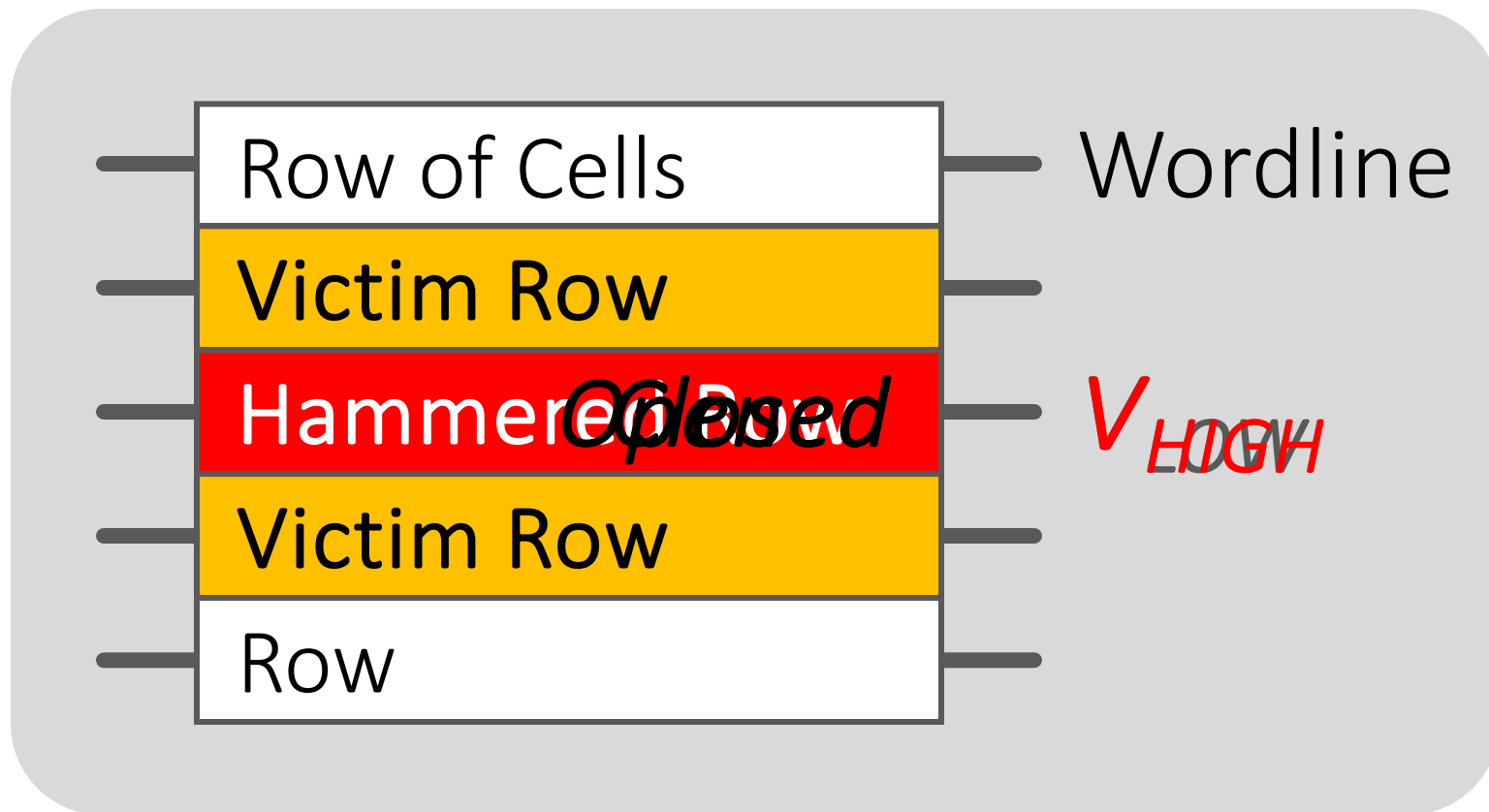
A Curious Phenomenon [Kim et al., ISCA 2014]

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.



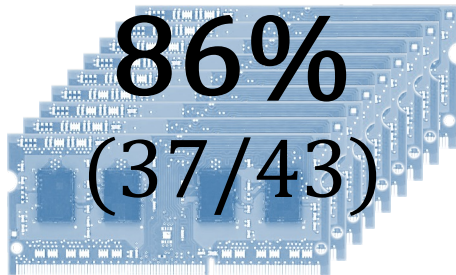
Modern Memory is Prone to Disturbance Errors



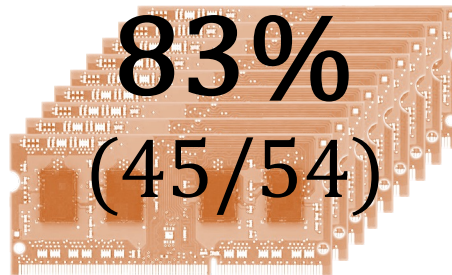
Repeatedly reading a row enough times (before memory gets refreshed) induces **disturbance errors** in adjacent rows in **most real DRAM chips you can buy today**

Most DRAM Modules Are Vulnerable

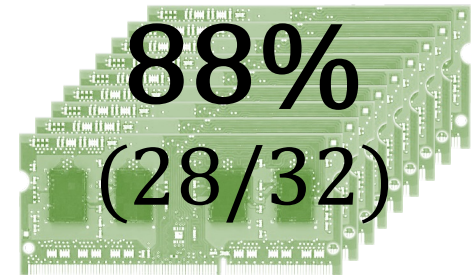
A company



B company



C company



Up to
 1.0×10^7
errors

Up to
 2.7×10^6
errors

Up to
 3.3×10^5
errors

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

¹Carnegie Mellon University ²Intel 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^{§‡}
§ETH Zürich

Jeremie S. Kim^{‡§}
‡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.olgün@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)]



Security Aspects of DRAM
The Story of RowHammer

Onur Mutlu
omutlu@gmail.com
<https://people.inf.ethz.ch/omutlu>
15 May 2022
IMW Tutorial

SAFARI ETH zürich Carnegie Mellon

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

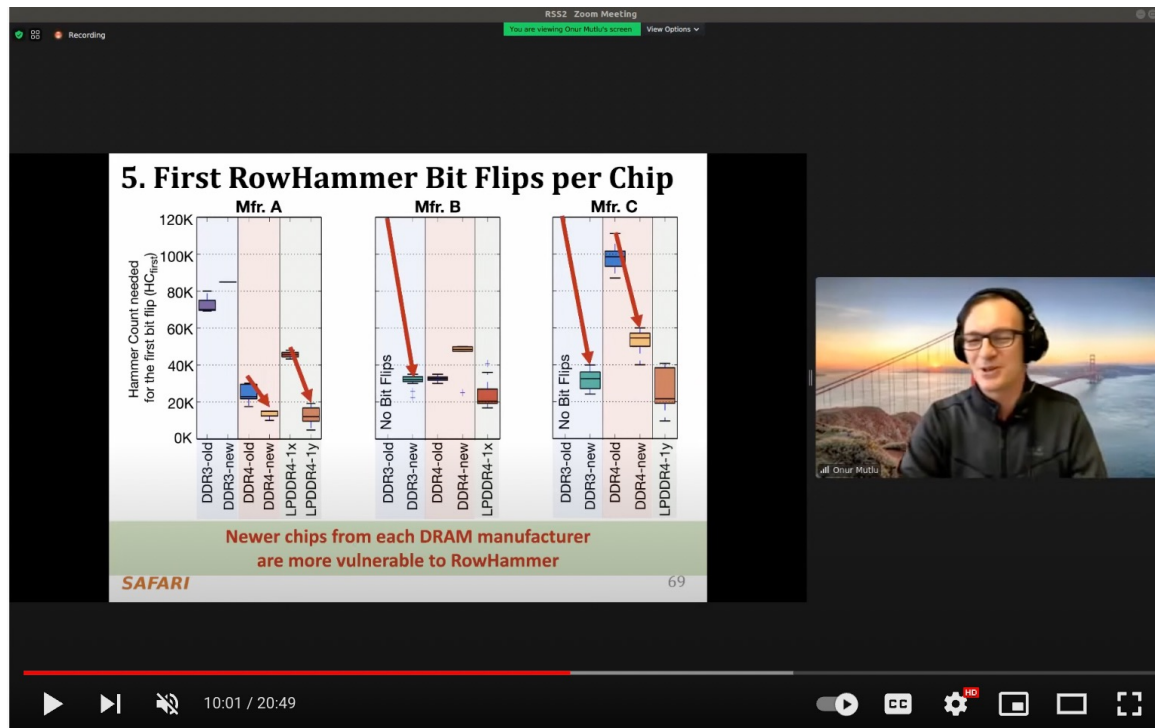
<https://www.youtube.com/watch?v=37hWqIkQRGO>

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



Latest RowHammer Lecture

Collapse of the “Galloping Gertie”



2:18 / 1:24:50 • Secure, Reliable and Safe >
SAFARI <http://www.wsdot.wa.gov/tnbhistory/connections/connections3.htm>

Securing the Memory System: The Story of RowHammer - Talk at NYU 23 June 2023 (Prof. Onur Mutlu)



Onur Mutlu Lectures
35.2K subscribers



Subscribed



14



Share



Download



Clip



454 views 1 month ago

Title: Securing the Memory System: The Story of RowHammer

Main Memory Needs Intelligent Controllers

An Example Intelligent Controller

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

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

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

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

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

[[Intel Hardware Security Academic Awards Short Talk Slides \(pptx\)](#) ([pdf](#))]

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

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

[[Intel Hardware Security Academic Awards Short Talk Video](#) (2 minutes)]

[[BlockHammer Source Code](#)]

Intel Hardware Security Academic Award Finalist (one of 4 finalists out of 34 nominations)

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

A. Giray Yağlıkcı¹ Minesh Patel¹ Jeremie S. Kim¹ Roknoddin Azizi¹ Ataberk Olgun¹ Lois Orosa¹
Hasan Hassan¹ Jisung Park¹ Konstantinos Kanellopoulos¹ Taha Shahroodi¹ Saugata Ghose² Onur Mutlu¹

¹*ETH Zürich*

²*University of Illinois at Urbana-Champaign*

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, 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

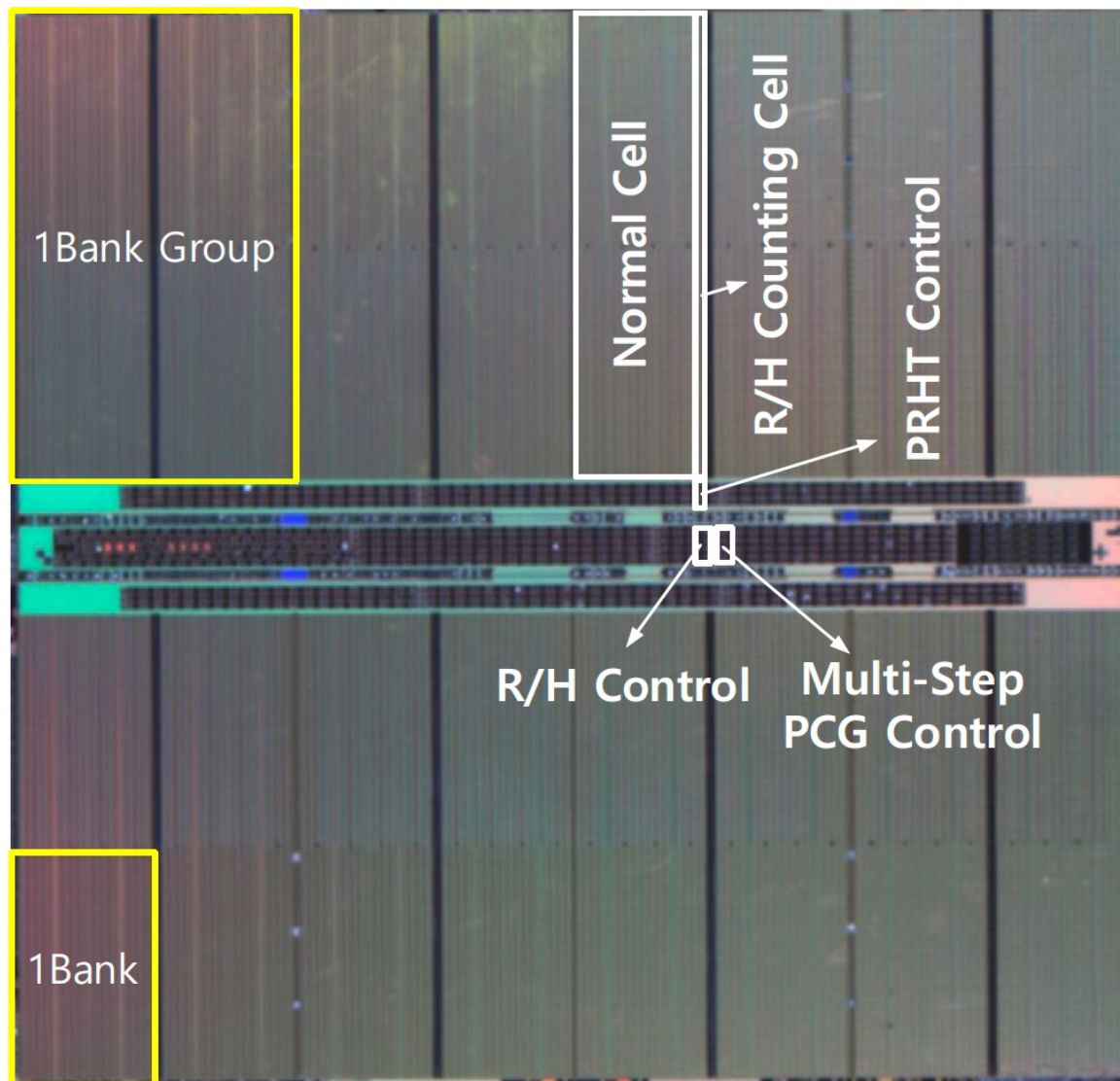


Industry's Intelligent DRAM Controllers (II)

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, Dhyong 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?



- 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](#))]

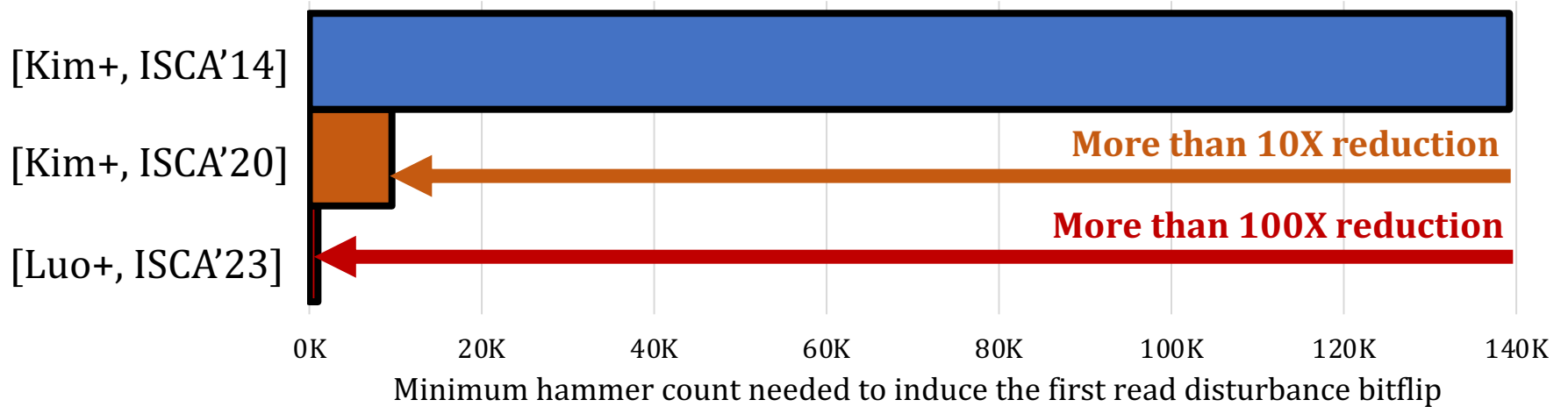
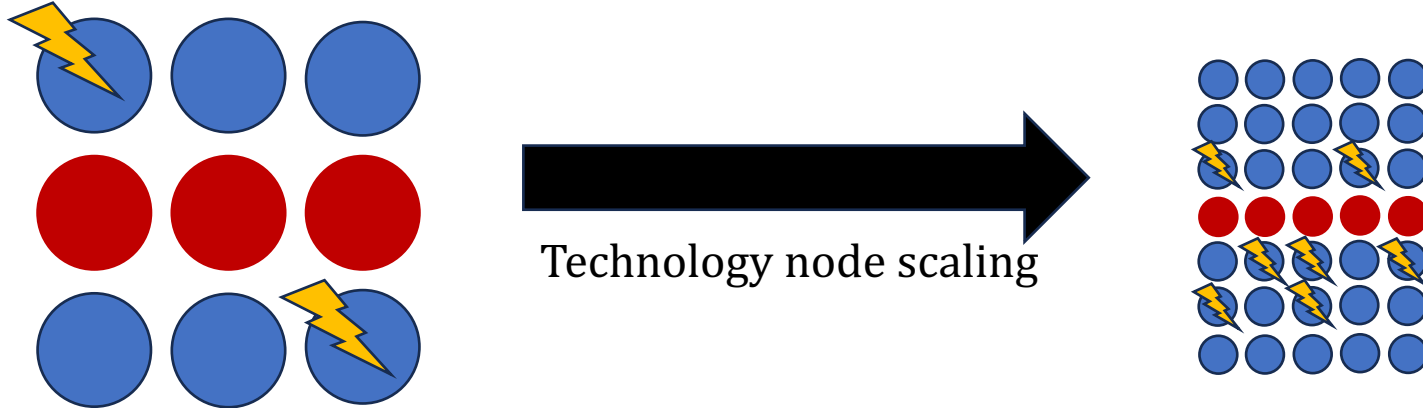
[[Lightning Talk Slides \(pptx\)](#) ([pdf](#))]

***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

DRAM Read Disturbance: A Critical Challenge



DRAM cells become increasingly more vulnerable to read disturbance

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[†] Engin Ipek[†] Onur Mutlu[‡] Doug Burger[†]

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

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

Industry Is Writing Papers About It, Too

DRAM Process Scaling Challenges

❖ Refresh

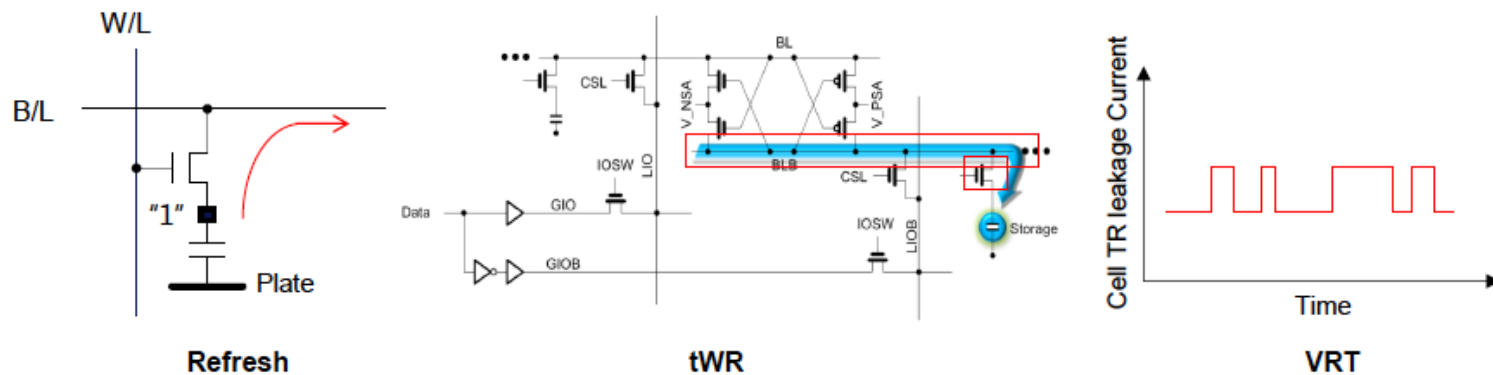
- Difficult to build high-aspect ratio cell capacitors decreasing cell capacitance
- Leakage current of cell access transistors increasing

❖ tWR

- Contact resistance between the cell capacitor and access transistor increasing
- On-current of the cell access transistor decreasing
- Bit-line resistance increasing

❖ VRT

- Occurring more frequently with cell capacitance decreasing



Call for Intelligent Memory Controllers

DRAM Process Scaling Challenges

❖ Refresh

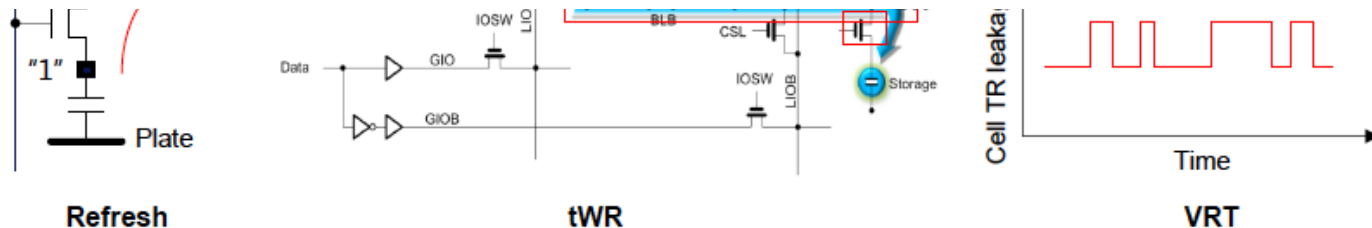
- Difficult to build high-aspect ratio cell capacitors decreasing cell capacitance

THE MEMORY FORUM 2014

Co-Architecting Controllers and DRAM to Enhance DRAM Process Scaling

Uksong Kang, Hak-soo Yu, Churoo Park, *Hongzhong Zheng,
**John Halbert, **Kuljit Bains, SeongJin Jang, and Joo Sun Choi

*Samsung Electronics, Hwasung, Korea / *Samsung Electronics, San Jose / **Intel*



**Intelligent
Memory Controllers
Can Avoid Many Failures
& Enable Better Scaling**

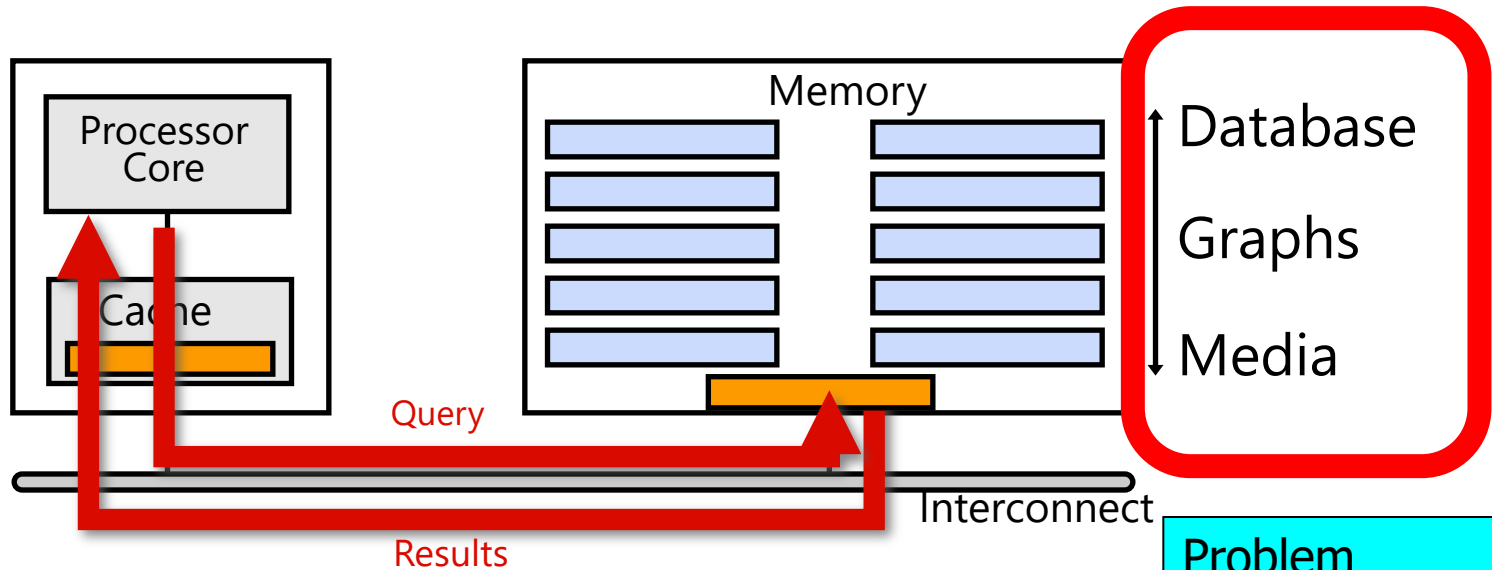
Why In-Memory Computation Today?

- **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

- **Huge problems with Memory Technology**
 - Memory technology scaling is not going well (e.g., RowHammer)
 - Many scaling issues demand intelligence in memory

- **Designs are squeezed in the middle**

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

SAFARI Research Group

^a*ETH Zürich*

^b*Carnegie Mellon University*

^c*University of Illinois at Urbana-Champaign*

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

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

"A Modern Primer on Processing in Memory"

*Invited Book Chapter in **Emerging Computing: From Devices to Systems - Looking Beyond Moore and Von Neumann**, Springer, 2022.*

Two PIM Approaches

5.2. Two Approaches: Processing Using Memory (PUM) vs. Processing Near Memory (PNM)

Many recent works take advantage of the memory technology innovations that we discuss in Section 5.1 to enable and implement PIM. We find that these works generally take one of two approaches, which are categorized in Table 1: (1) *processing using memory* or (2) *processing near memory*. We briefly describe each approach here. Sections 6 and 7 will provide example approaches and more detail for both.

Table 1: Summary of enabling technologies for the two approaches to PIM used by recent works. Adapted from [341] and extended.

Approach	Example Enabling Technologies
Processing Using Memory	SRAM DRAM Phase-change memory (PCM) Magnetic RAM (MRAM) Resistive RAM (RRAM)/memristors
Processing Near Memory	Logic layers in 3D-stacked memory Silicon interposers Logic in memory controllers Logic in memory chips (e.g., near bank) Logic in memory modules Logic near caches Logic near/in storage devices

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. [[Tutorial Video on "Memory-Centric Computing Systems"](#) (1 hour 51 minutes)]

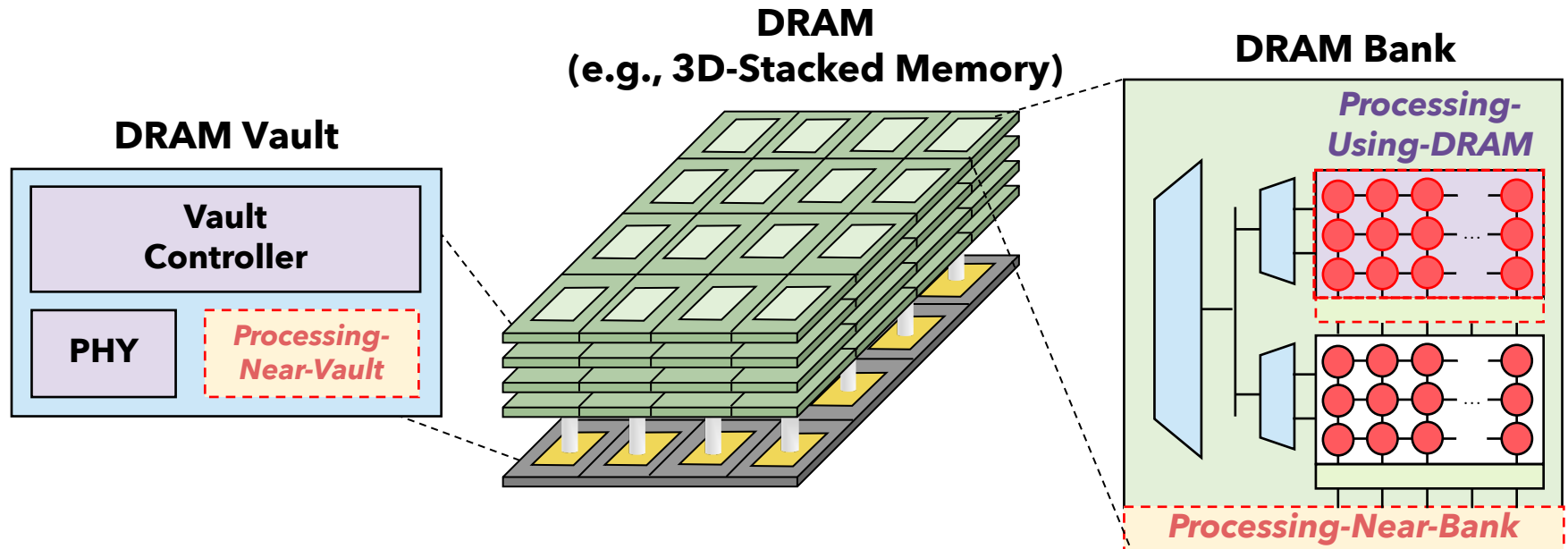
Processing in Memory: Two Approaches

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

Processing-in-Memory: Nature of Computation

Two main approaches for Processing-in-Memory:

- 1 Processing-Near-Memory:** Design compute logic and memory separately (as today) and integrate logic closer to memory
- 2 Processing-Using-Memory:** Use analog operational principles of memory circuitry to perform computation (no compute logic)



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

Agenda

- Processing-Near-Memory Systems: Developments from Academia & Industry
- **Invited Talk by Dr. Brian Schwedock:**
Architectures and Programming Models for General-Purpose Near-Data Computing
- Processing-Using-Memory Systems for Bulk Bitwise Operations
- Coffee Break
- Ataberk Olgun:
Infrastructure for Processing-Using-Memory Research
- **Invited Talk by Dr. Christina Giannoula:**
System Software and Libraries for Sparse Computational Kernels in PIM Architectures
- Nika Mansouri Ghiasi:
Storage-Centric Computing for Genomics and Metagenomics
- Research Challenges for PIM & Closing Remarks

Tutorial on Memory-Centric Computing: Introduction

Geraldo F. Oliveira

<https://geraldofojunior.github.io>

MICRO 2024

02 November 2024

Agenda

- **Processing-Near-Memory Systems: Developments from Academia & Industry**
- **Invited Talk by Dr. Brian Schwedock:**
Architectures and Programming Models for General-Purpose Near-Data Computing
- Processing-Using-Memory Systems for Bulk Bitwise Operations
- Coffee Break
- Ataberk Olgun:
Infrastructure for Processing-Using-Memory Research
- **Invited Talk by Dr. Christina Giannoula:**
System Software and Libraries for Sparse Computational Kernels in PIM Architectures
- Nika Mansouri Ghiasi:
Storage-Centric Computing for Genomics and Metagenomics
- Research Challenges for PIM & Closing Remarks