In-Memory Acceleration of Genome Analysis

Onur Mutlu

omutlu@gmail.com

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

10 August 2023

Flash Memory Summit
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**

SAFARI

And, many, many other applications ...
Overview: Genome Analysis

- **Genome analysis**: Enables us to determine the order of the DNA sequence in an organism’s genome
  - Plays an important role in
    - Personalized medicine
    - Outbreak tracing
    - Understanding of evolution
    - ...

- Modern genome sequencing machines extract smaller randomized fragments of the original DNA sequence, known as *reads*
  - **Oxford Nanopore Technologies (ONT)**: A widely-used sequencing technology
    - Portable sequencing devices
    - High-throughput
    - Cheap

[ont sequencing device](forbes.com)
Genome Sequencers

Roche/454

AB SOLiD

Illumina MiSeq

Complete Genomics

Illumina HiSeq2000

Pacific Biosciences RS

Oxford Nanopore MinION

Illumina NovaSeq 6000

Ion Torrent PGM

Ion Torrent Proton

Oxford Nanopore GridION

... and more! All produce data with different properties.
High-Throughput Sequencers

... and more! All produce data with different properties.
Newer 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](https://doi.org/10.1093/bib/bby017)

Published: 02 April 2018   Article history ▼


[Open arxiv.org version] [Slides (pptx) (pdf)] [Talk Video at AACBB 2019]
Genome Sequencing Cost Is Reducing

*From NIH ([https://www.genome.gov/about-genomics/fact-sheets/DNA-Sequencing-Costs-Data](https://www.genome.gov/about-genomics/fact-sheets/DNA-Sequencing-Costs-Data))
Problems with (Genome) Analysis Today

Special-Purpose Machine for Data Generation

General-Purpose Machine for Data Analysis

FAST

SLOW

Slow and inefficient processing capability
Large amounts of data movement

SAFARI This picture is similar for many “data generators & analyzers” today
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

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=PL5Q2soXY2Zi8tFlQvdxOdizD_EhVAMVQV](https://www.youtube.com/watch?v=nA41964-9r8&list=PL5Q2soXY2Zi8tFlQvdxOdizD_EhVAMVQV)

- **Youtube Livestream (Spring 2022):**
  - [https://www.youtube.com/watch?v=DEL_5A_Y3TI&list=PL5Q2soXY2Zi8NrpDqOR1yRU_Cxujw-u18](https://www.youtube.com/watch?v=DEL_5A_Y3TI&list=PL5Q2soXY2Zi8NrpDqOR1yRU_Cxujw-u18)

- Project course
  - Taken by Bachelor’s/Master’s students
  - Genomics lectures
  - Hands-on research exploration
  - Many research readings
  - [https://www.youtube.com/onurmutlulectures](https://www.youtube.com/onurmutlulectures)
PIM Course (Fall 2022)

- **Fall 2022 Edition:**
  - https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=processing_in_memory

- **Spring 2022 Edition:**
  - https://safari.ethz.ch/projects_and_seminars/spring2022/doku.php?id=processing_in_memory

- **Youtube Livestream (Fall 2022):**
  - https://www.youtube.com/watch?v=QLL0wQ9I4Dw&list=PL5Q2soXY2Zi8KzG2CQYRNQOVD0GOBrnKy

- **Youtube Livestream (Spring 2022):**
  - https://www.youtube.com/watch?v=9e4Chnwdovo&list=PL5Q2soXY2Zi-841fUYUYUK9EsXKhQKRPyX

- **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
GenPIP

In-Memory Acceleration of Genome Analysis via Tight Integration of Basecalling and Read Mapping

Haiyu Mao, Mohammed Alser, Mohammad Sadrosadati, Can Firtina, Akanksha Baranwal, Damla Senol Cali, Aditya Manglik, Nour Almadhoun Alserr, Onur Mutlu
Overview: Two Limitations

Multiple steps in genome analysis

- Large data movement between multiple steps
- A lot of wasted computation done on data that is later discovered to be useless
Overview: GenPIP

- **GenPIP**: A fast and energy-efficient *in-memory* acceleration system for the Genome analysis PIPeline via *tight integration of genome analysis steps*

- **GenPIP** has two key techniques
  - **Chunk-based pipeline (CP)**
    - Provides fine-grained collaboration of genome analysis steps
  - **Early rejection (ER)**
    - Timely stops the execution on useless data by predicting which reads will not be useful

- GenPIP outperforms state-of-the-art software & hardware solutions using **CPU**, **GPU**, and optimistic **PIM** by **41.6x**, **8.4x**, and **1.4x**, respectively.
Outline

- Background and Motivation
  - GenPIP: Tight Integration of Genome Analysis Steps
    - Chunk-based Pipeline (CP)
    - Early Rejection (ER)
  - GenPIP Implementation
- Evaluation
- Conclusion
Genome Analysis Pipeline

1. Basecalling

2. Read Quality Control

3. Read Mapping

Mapped

Unmapped

SAFARI
Limitation 1: Large Data Movement

- Using a human dataset in [NC’19] as an example:

Raw Signals → Basecalling → Reads → Read quality control → High-quality reads → Read mapping → Mapped reads

3913 GB → 546 GB → 437 GB → 382 GB

Large data movement between genome analysis steps

Limitation 2: Wasted Computation

- Using a human dataset in [NC’19] as an example:

<table>
<thead>
<tr>
<th>Raw Signals</th>
<th>Basecalling</th>
<th>Reads</th>
<th>Read quality control</th>
<th>High-quality reads</th>
<th>Read mapping</th>
<th>Mapped reads</th>
</tr>
</thead>
<tbody>
<tr>
<td>100%</td>
<td></td>
<td></td>
<td></td>
<td>79.5%</td>
<td></td>
<td>69.5%</td>
</tr>
</tbody>
</table>

- A considerable amount of computation on **useless data** due to
  - Low-quality reads
  - Unmapped reads

State-of-the-art Works

- NVM-based PIM is an efficient technique to reduce data movement by processing data using or near memory.

- Reduce the data movement in a single genome analysis step
- Exacerbate the data movement overhead between analysis steps

No prior work tackles data movement between analysis steps and reduces useless computation.
Goal: Efficiently accelerate the entire genome analysis pipeline while minimizing data movement and useless computation

We perform a study to quantify potential performance benefits

- Results are normalized to the performance of GPU

![Normalized Speedup Chart]

- NVM-based PIM accelerators for separate basecalling and read mapping: 2.7x
- No data movement between the accelerators of analysis steps: 6.1x
- No data movement and no useless reads (ideal case): 9x
Outline

- Background and Motivation

- GenPIP: Tight Integration of Genome Analysis Steps
  - Chunk-based Pipeline (CP)
  - Early Rejection (ER)

- GenPIP Implementation

- Evaluation

- Conclusion
GenPIP

- *First holistic in-memory accelerator for the genome analysis pipeline*, including basecalling, read quality control, and read mapping steps

- **GenPIP** has two key techniques
  
  - **Chunk-based Pipeline (CP)**
    - Enables fine-grained pipelining of genome analysis steps
    - Processes reads at **chunk** granularity (i.e., a subsequence; 300 bases)
  
  - **Early Rejection (ER)**
**Chunk-based Pipeline (CP)**

- **CP increases parallelism** by overlapping the execution of different steps at chunk granularity.
- **CP reduces intermediate data** by computing on data as soon as data is generated.
- **CP provides opportunities for ER** by analyzing a read at chunk granularity.

A read consists of four chunks: **C1, C2, C3, C4**.

---

**SAFARI**

23
GenPIP

- *First holistic in-memory accelerator for the genome analysis pipeline*, including basecalling, read quality control, and read mapping steps

- **GenPIP** has two key techniques

  - **Chunk-based Pipeline (CP)**
    - Enables fine-grained collaboration of genome analysis steps by processing reads at chunk granularity (i.e., a subsequence of a read, e.g., 300 bases)

  - **Early Rejection (ER)**
    - Stops the execution on useless reads as early as possible by using a small number of chunks to predict the usefulness of a read
Early Rejection (ER)

- **Predict and eliminate** low-quality and unmapped reads from the genome analysis pipeline **as early as possible**

![Diagram of Early Rejection process]

- **Early-Rejection based on chunk quality scores (ER-QSR)**
  - Predict low-quality reads using chunk quality scores

- **Early-Rejection based on chunk mapping scores (ER-CMR)**
  - Predict unmapped reads using chunk mapping scores
Implementation of CP and ER

CP and ER can be applied on different systems, e.g., CPU, GPU, and PIM

We implement CP and ER using PIM since PIM is more efficient to reduce the data movement between genome analysis steps

We also apply CP and ER on CPU and GPU baselines and observe speedup and energy savings
Outline

- Background and Motivation
- GenPIP: Tight Integration of Genome Analysis Steps
  - Chunk-based Pipeline (CP)
  - Early Rejection (ER)
- GenPIP Implementation
- Evaluation
- Conclusion
GenPIP Implementation

Raw signals from the sequencing machine

- In-memory Basecaller [Helix, PACT’20]
- Basecalling Module
  - Base quality score
  - PIM-CQS
    - PIM chunk quality score calculation
- eDRAM
  - Signal chunk
  - Basecalled chunk
  - Chunk quality score
  - Chunk mapping score
- Average Calculator
  - Quality score
- ER Controller
  - GenPIP Controller
  - ER
- Read Mapping Controller
  - Read Mapping Module
  - Read mapping result
  - To storage

+ Our design
- In-memory Read Mapping [PARC, ASPDAC’20]

GenPIP Implementation

Tightly integrating the genome analysis steps
- Reduces data movement
- Eliminates useless computation
Outline

- Background and Motivation
- GenPIP: Tight Integration of Genome Analysis Steps
  - Chunk-based Pipeline (CP)
  - Early Rejection (ER)
- GenPIP Implementation
- Evaluation
- Conclusion
Evaluation Methodology

- **Performance, Area and Power Analysis:**
  - Simulation via Verilog HDL, NVSim [TCAD’12], and CACTI 6.5 [MICRO’07]
  - See methodology in the paper for more

- **Baselines:**
  - **CPU** (Intel Xeon Gold 5118 CPU)
  - **GPU** (NVIDIA GeForce RTX 2080 Ti GPU)
  - **Optimistic integration of two PIM accelerators** (Helix [PACT’20] and PARC [ASP-DAC’20])
    - Assumes no data movement between steps
    - Assumes intermediate data causes no overhead

- **Datasets:**
  - **E. coli** ([http://lab.loman.net/2016/07/30/nano_pore-r9-data-release/](http://lab.loman.net/2016/07/30/nano_pore-r9-data-release/))
  - **Human** ([https://www.ebi.ac.uk/ena/browser/view/PRJEB30620](https://www.ebi.ac.uk/ena/browser/view/PRJEB30620))
GenPIP provides $41.6x$, $8.4x$, and $1.4x$ speedup over CPU, GPU, and optimistic PIM.

Both CP and ER are critical to speedup.
Key Results – Energy Efficiency

GenPIP provides 32.8x, 20.8x, and 1.37x energy savings over CPU, GPU, and optimistic PIM.

ER is especially critical to energy efficiency.
More in the Paper

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

Haiyu Mao¹  Mohammed Alser¹  Mohammad Sadrosadati¹  Can Firtina¹  Akanksha Baranwal¹
Damla Senol Cali²  Aditya Manglik¹  Nour Almadhoun Alserr¹  Onur Mutlu¹

¹ETH Zürich  ²Bionano Genomics

- Timely early rejection implementation
- In-memory seeding accelerator
- More comparison points
- Sensitivity analysis for ER
- Area and power analysis
More in the Paper

- Details of **CP and ER**

- Detailed **GenPIP** implementation
  - GenPIP controller
  - Early rejection implementation
  - In-memory seeding accelerator

- Results of applying **CP and ER in CPU and GPU**

- **Sensitivity analysis** on the number of sampled chunks used for ER

- **Area and power** analysis
Outline

- Background and Motivation
- GenPIP: Tight Integration of Genome Analysis Steps
  - Chunk-based Pipeline (CP)
  - Early Rejection (ER)
- GenPIP Implementation
- Evaluation
- Conclusion
Summary

- **Problem:** The genome analysis pipeline has large data movement between genome analysis steps and a significant amount of wasted computation on useless data.

- **Goal:** Tightly integrate genome analysis steps to reduce the data movement between steps and eliminate computation on useless data.

- **GenPIP:** The first in-memory genome analysis accelerator that tightly integrates genome analysis steps.
  - GenPIP has two key techniques:
    - A chunk-based pipeline
    - A new early-rejection technique

- GenPIP outperforms state-of-the-art software & hardware solutions using CPU, GPU, and optimistic PIM by 41.6x, 8.4x, and 1.4x, respectively.
GenPIP

In-Memory Acceleration of Genome Analysis via Tight Integration of Basecalling and Read Mapping

Haiyu Mao, Mohammed Alser, Mohammad Sadrosadati, Can Firtina, Akanksha Baranwal, Damla Senol Cali, Aditya Manglik, Nour Almadhoun Alserr, Onur Mutlu

SAFARI

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

Haiyu Mao¹ Mohammed Alser¹ Mohammad Sadrosadati¹ Can Firtina¹ Akanksha Baranwal¹ Damla Senol Cali² Aditya Manglik¹ Nour Almadhoun Alser² Onur Mutlu¹

¹ETH Zürich ²Bionano Genomics

- Timely early rejection implementation
- In-memory seeding accelerator
- More comparison points for ER
- Sensitivity analysis
- Area and power analysis

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 Mao¹ Mohammed Alser¹ Mohammad Sadrosadati¹ Can Firtina¹ Akanksha Baranwal¹
Damla Senol Cali² Aditya Manglik¹ Nour Almadhoun Alserr¹ Onur Mutlu¹
¹ETH Zürich ²Bionano Genomics

Comp Arch (Fall 2021)

- **Fall 2021 Edition:**
  - [https://safari.ethz.ch/architecture/fall2021/doku.php?id=schedule](https://safari.ethz.ch/architecture/fall2021/doku.php?id=schedule)

- **Fall 2020 Edition:**

- **Youtube Livestream (2021):**
  - [https://www.youtube.com/watch?v=4yfkM_5Efqo&list=PL5Q2soXY2Zi-Mnk1PxjEIG32HAGILkTOF](https://www.youtube.com/watch?v=4yfkM_5Efqo&list=PL5Q2soXY2Zi-Mnk1PxjEIG32HAGILkTOF)

- **Youtube Livestream (2020):**
  - [https://www.youtube.com/watch?v=c3mPdZAFmc&list=PL5Q2soXY2Zi9xydIgBxUz7xRPS-wisBN](https://www.youtube.com/watch?v=c3mPdZAFmc&list=PL5Q2soXY2Zi9xydIgBxUz7xRPS-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](https://www.youtube.com/onurmutlulectures)
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=PL5Q2soXY2Zi8tFIQvdxOdzD_EhVAMVQV](https://www.youtube.com/watch?v=nA41964-9r8&list=PL5Q2soXY2Zi8tFIQvdxOdzD_EhVAMVQV)

- **Youtube Livestream (Spring 2022):**
  - [https://www.youtube.com/watch?v=DEL_5A_Y3TI&list=PL5Q2soXY2Zi8NnPqOR1yRU_Cxjw-u18](https://www.youtube.com/watch?v=DEL_5A_Y3TI&list=PL5Q2soXY2Zi8NnPqOR1yRU_Cxjw-u18)

- **Project course**
  - Taken by Bachelor’s/Master’s students
  - Genomics lectures
  - Hands-on research exploration
  - Many research readings

[https://www.youtube.com/onurmutlulectures](https://www.youtube.com/onurmutlulectures)
**PIM Course (Fall 2022)**

- **Fall 2022 Edition:**
  - [https://safari.ethz.ch/projects_and_seminars/fall2022/doku.php?id=processing_in_memory](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=PL5Q2soXY2Zi8KzG2CQYRNQVD0GOBrnK](https://www.youtube.com/watch?v=QLL0wQ9I4Dw&list=PL5Q2soXY2Zi8KzG2CQYRNQVD0GOBrnK)

- **Youtube Livestream (Spring 2022):**
  - [https://www.youtube.com/watch?v=9e4Chnwdovo&list=PL5Q2soXY2Zi-841fUYYUK9EsXKhQKRPyX](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](https://www.youtube.com/onurmutlulectures)
Processing-in-Memory Course (Spring 2023)

- Short weekly lectures
- Hands-on projects

https://www.youtube.com/playlist?list=PL5Q2soXY2Zi_EObuoaZVSQ_o6UySWQHzV

https://safari.ethz.ch/projects_and_seminars/spring2023/doku.php?id=processing_in_memory
Real PIM Tutorials [ISCA’23, ASPLOS’23, HPCA’23]

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

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

2,560-DPU Processing-in-Memory System

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

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

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/

### Tutorial Materials

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

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

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

[https://www.youtube.com/live/GIb5EgSrWko?feature=share](https://www.youtube.com/live/GIb5EgSrWko?feature=share)

[https://events.safari.ethz.ch/isca-pim-tutorial/](https://events.safari.ethz.ch/isca-pim-tutorial/)
Real PIM Tutorial [ASPLOS 2023]

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

---

**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., UPNEM, 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 applicability of design techniques for new domains, and (3) design optimized PIM platforms.

**Tutorial Materials**

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

---

[https://www.youtube.com/watch?v=oYCaLcT0Kmo](https://www.youtube.com/watch?v=oYCaLcT0Kmo)

[https://events.safari.ethz.ch/asplos-pim-tutorial/](https://events.safari.ethz.ch/asplos-pim-tutorial/)
Real PIM Tutorial [HPCA 2023]

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

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?

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

https://events.safari.ethz.ch/real-pim-tutorial/
In-Memory Acceleration of Genome Analysis

Onur Mutlu
omutlu@gmail.com
https://people.inf.ethz.ch/omutlu
10 August 2023
Flash Memory Summit