P&S DRAM Bender

FPGA-based Exploration of DRAM and RowHammer

Ataberk Olgun
Prof. Onur Mutlu
ETH Zürich

Fall 2022
4 October 2022

P&S DRAM Bender: Content

- We will learn in detail how **modern DDR4 DRAM** operates

- You will learn how to characterize DRAM using an **FPGA-based DRAM characterization infrastructure (DRAM Bender)**

- You will use DRAM Bender to **develop your own DRAM experiments** and gain hand-on experience in studying DRAM characteristics

P&S DRAM Bender: Key Takeaways

- This P&S is aimed at improving your
  - Knowledge in Computer Architecture and Memory Systems
  - Technical skills in running DRAM experiments using real devices
  - Critical thinking and analysis
  - Familiarity with key research directions
  - Technical presentation of your project
  - Communication skills (by interacting with a group of researchers)

P&S DRAM Bender: Key Goal

(Learn how to) study real memory devices using an FPGA-based DRAM infrastructure to gain new insights on DRAM behavior

Prerequisites of the Course

- Digital Design and Computer Architecture (or equivalent course)
- Familiarity with FPGA programming
- Familiarity with a programming language (we will use C++/Python)
- Interest in low-level hacking and memory
- Interest in discovering why things do or do not work and solving problems

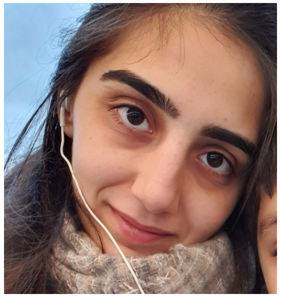
Course Info: Who Are We? (I)

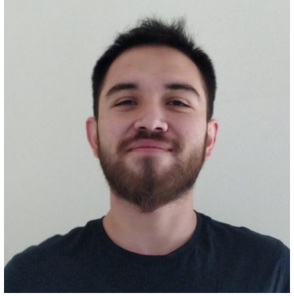
- Onur Mutlu
  - Full Professor @ ETH Zurich ITET (INFK), since September 2015
  - Strecker Professor @ Carnegie Mellon University ECE/CS, 2009-2016, 2016-…
  - PhD from UT-Austin, worked at Google, VMware, Microsoft Research, Intel, AMD
  - https://people.inf.ethz.ch/omutlu/
  - omutlu@gmail.com (Best way to reach me)
  - https://people.inf.ethz.ch/omutlu/projects.htm

- Research and Teaching in:
  - Computer architecture, computer systems, hardware security, bioinformatics
  - Memory and storage systems
  - Hardware security, safety, predictability
  - Fault tolerance
  - Hardware/software cooperation
  - Architectures for bioinformatics, health, medicine
  - …

Course Info: Who Are We? (II)

- Lead Supervisor:
  - Ataberk Olgun
- Supervisors:
  - Giray Yaglikci
  - Haocong Luo
  - Yahya Tugrul
  - Banu Cavlak
  - Ismail Yuksel
- Get to know us and our research
  - https://safari.ethz.ch/group-members





Onur Mutlu’s SAFARI Research Group

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

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

38+ Researchers

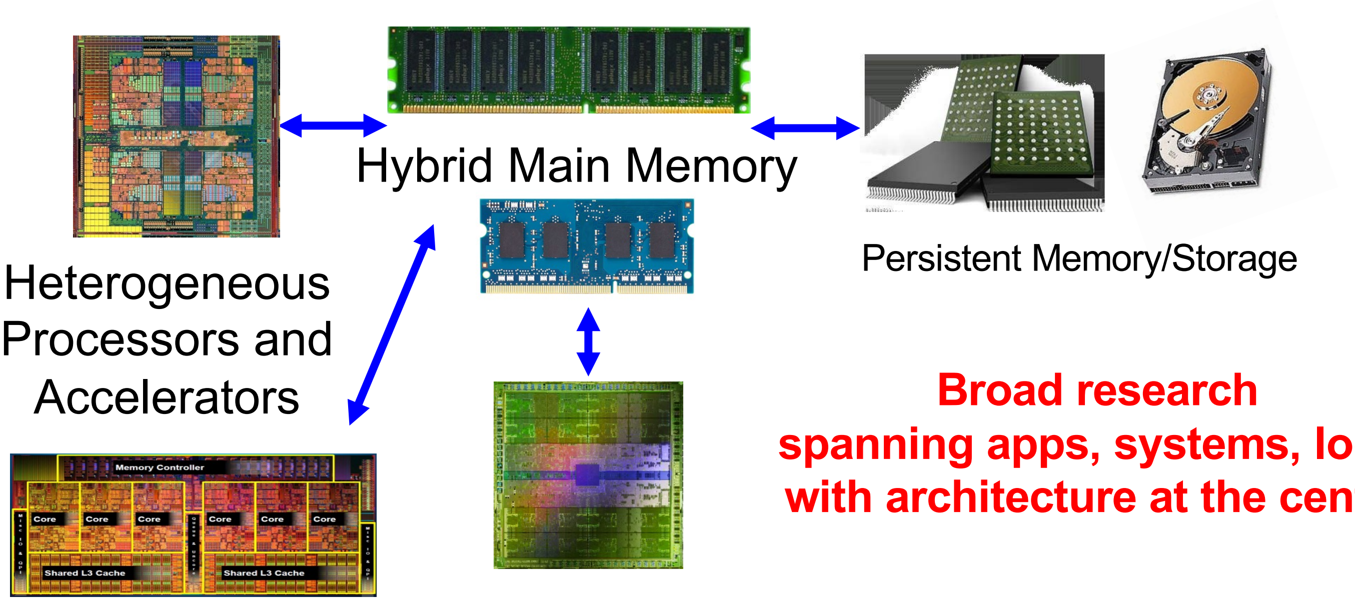
Think BIG, Aim HIGH!

https://safari.ethz.ch

Current Research Focus Areas

Research Focus: ***Computer architecture, HW/SW, bioinformatics***

- *Memory and storage (DRAM, flash, emerging), interconnects*
- *Heterogeneous & parallel systems, GPUs, systems for data analytics*
- *System/architecture interaction, new execution models, new interfaces*
- *Energy efficiency, fault tolerance, hardware security, performance*
- *Genome sequence analysis & assembly algorithms and architectures*
- *Biologically inspired systems & system design for bio/medicine*



Hybrid Main Memory

Heterogeneous Processors and Accelerators

Persistent Memory/Storage

Graphics and Vision Processing

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

Course Info: How About You?

- Let us know your background, interests

- Why did you join this P&S?

- Please submit HW0
  - Moodle link in the homework handout

Course Requirements and Expectations

- Attendance required for all (future) meetings
  - Meeting 2 (next week)
  - 1-1 meetings with supervisor(s)
  - Group meetings: Project updates

- Study the learning materials

- Each student will carry out a hands-on project
  - Build, implement, code, and design with close engagement from the supervisors

- Participation
  - Ask questions, contribute thoughts/ideas
  - Read relevant papers

We will help in all projects!
If your work is really good, you may get it published!

Course Website

- https://safari.ethz.ch/projects_and_seminars/doku.php?id=softmc
- Useful information about the course
- Check your email frequently for announcements

Meeting 1

- Required materials:

SoftMC Tutorial Video: https://youtu.be/909uTQu0lbA

SoftMC lecture: https://www.youtube.com/watch?v=tnSPEP3t-Ys

Paper describing SoftMC: https://people.inf.ethz.ch/omutlu/pub/softMC_hpca17.pdf

Example RowHammer study using SoftMC: https://people.inf.ethz.ch/omutlu/pub/Revisiting-RowHammer_isca20.pdf

- Recommended materials:

Example security attack study using SoftMC: https://people.inf.ethz.ch/omutlu/pub/rowhammer-TRRespass_ieee_security_privacy20.pdf

Example neural network acceleration study using SoftMC: https://people.inf.ethz.ch/omutlu/pub/EDEN-efficient-DNN-inference-with-approximate-memory_micro19.pdf

Example random number generation study using SoftMC: https://people.inf.ethz.ch/omutlu/pub/drange-dram-latency-based-true-random-number-generator_hpca19.pdf

Example physical unclonable function study using SoftMC: https://people.inf.ethz.ch/omutlu/pub/dram-latency-puf_hpca18.pdf

The original RowHammer study using SoftMC: https://people.inf.ethz.ch/omutlu/pub/dram-row-hammer_isca14.pdf

Meeting 2 (TBD)

- We will announce the projects and will give you some description about them
- You will have a week to submit your project preferences
- The supervisors would like to help you with selecting a project that matches your interests, skills, and background
- It is important that you study the learning materials before our next meeting!

Tentative Weekly Schedule

- Week 1 – Logistics & Intro to DRAM and SoftMC **[HPCA'17]**
- Week 2 – Live DRAM Bender Tutorial (Tentative) | Available Projects
- Week 3 – Deeper Look Into RowHammer **[MICRO'21]** **| 1-1 meetings start**
- Week 4 – Hidden Row Activation **[MICRO'22]**
- Week 4 – Uncovering in-DRAM TRR **[MICRO'21]**
- Week 5 – QUAC-TRNG **[ISCA'21]** **| Group meetings start**
- Week 6 – The Reach Profiler (REAPER) **[ISCA'17]**
- Week 7 – PiDRAM **[arXiv'21]**
- Week 8+ **Group meetings & 1-1 meetings**

Every week: 1-1 meeting with supervisor(s)

Every four weeks: Group meeting for project updates

Performance Assessment

We expect you to:

- **Learn** how DRAM operates and **perform** DRAM characterization using FPGAs
- **Achieve the goals** of your project
- **Deliver** your **code and results** with sufficient documentation
- Prepare a **final presentation** and present your work to SAFARI

An Introduction to DRAM and SoftMC

SoftMC

A Flexible and Practical Open-Source Infrastructure for Enabling Experimental DRAM Studies

Hasan Hassan,
Nandita Vijaykumar, Samira Khan,
Saugata Ghose, Kevin Chang,
Gennady Pekhimenko, Donghyuk Lee,
Oguz Ergin, Onur Mutlu

ETH zürich

Carnegie Mellon

TOBB
UNIVERSITY OF
ECONOMICS AND TECHNOLOGY

Executive Summary

- Two critical problems of DRAM: **Reliability** and **Performance**
  - Recently-discovered bug: *RowHammer*

- ***Characterize***, ***analyze***, and ***understand*** DRAM cell behavior

- We design and implement **SoftMC**, an FPGA-based DRAM testing infrastructure
  - Flexible and Easy to Use (C++ API)
  - Open-source (*github.com/CMU-SAFARI/SoftMC*)

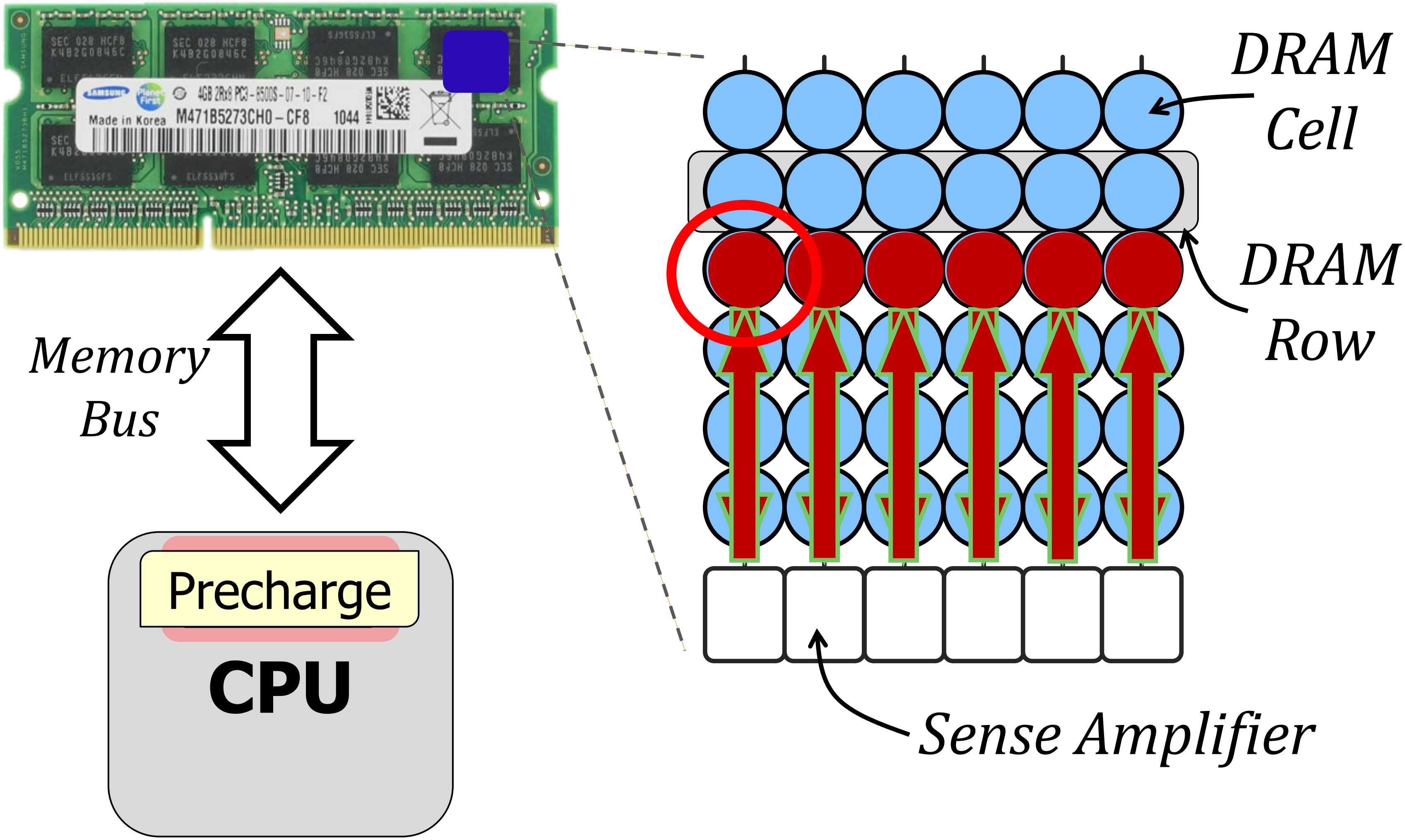
- We implement two use cases
  - A retention time distribution test
  - An experiment to validate two **latency reduction** mechanisms

- **SoftMC enables a wide range of studies**

Outline

1. DRAM Basics & Motivation
2. SoftMC
3. Use Cases
   - Retention Time Distribution Study
   - Evaluating Recently-Proposed Ideas
4. Future Research Directions
5. Conclusion

DRAM Operations



DRAM Cell

DRAM Row

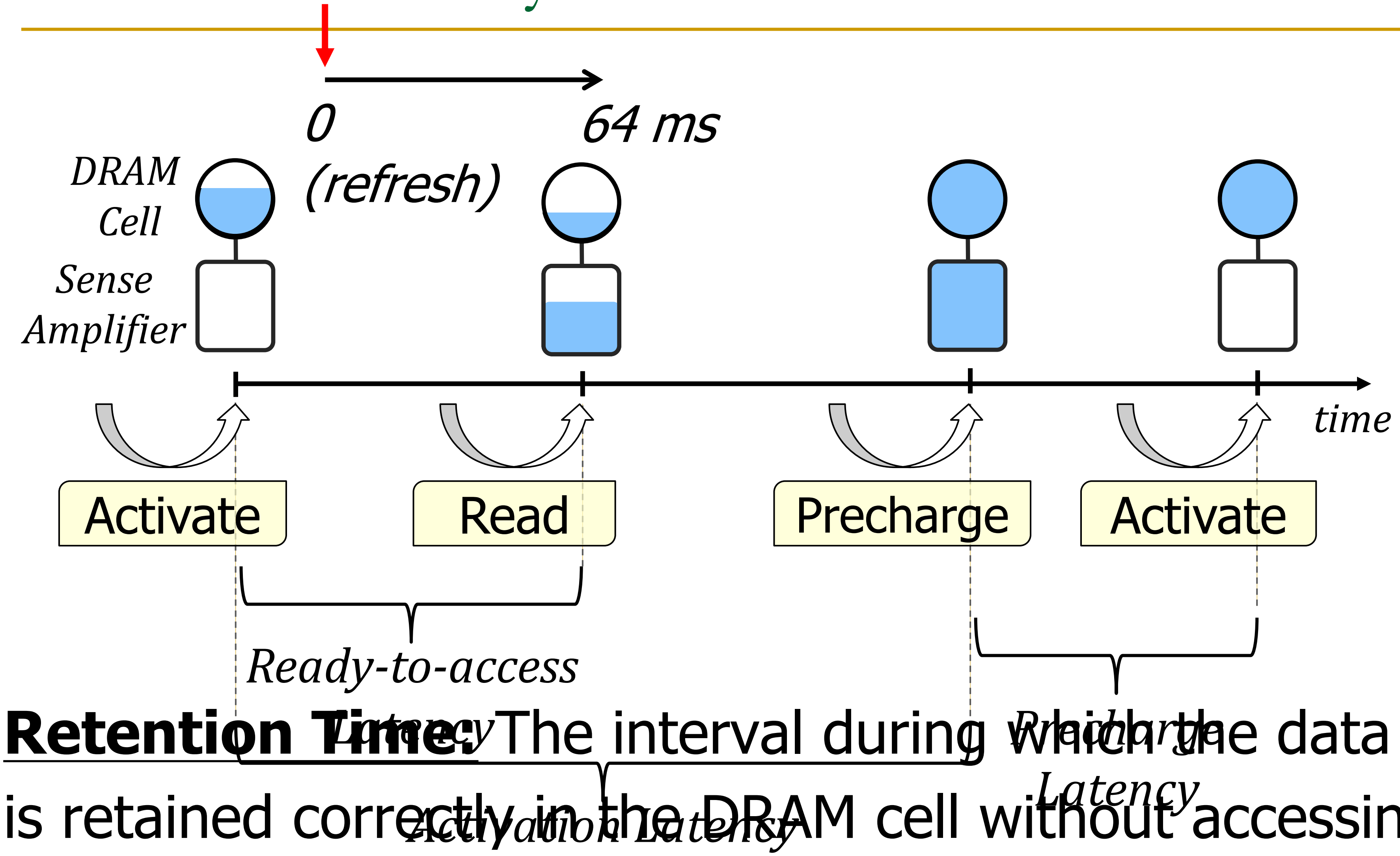
Sense Amplifier

Memory Bus

Precharge

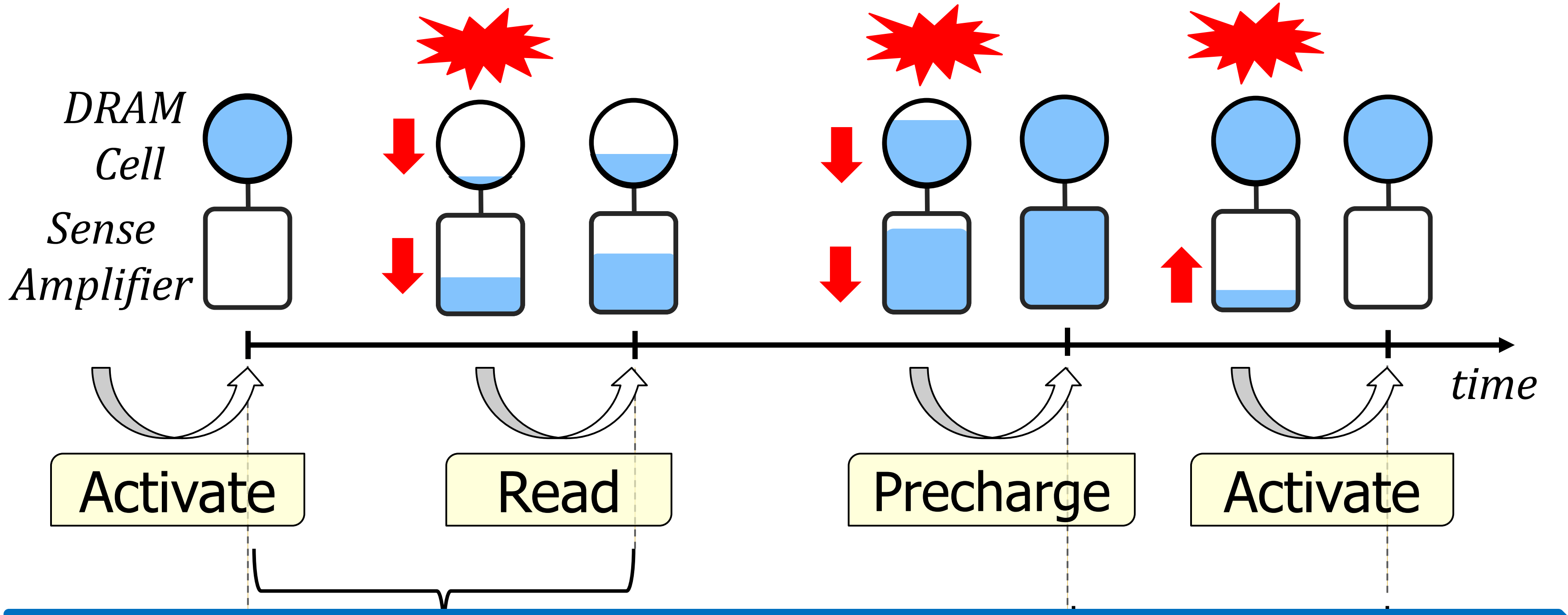
CPU

DRAM Latency



Retention Time: The interval during which the data is retained correctly in the DRAM cell without accessing it

Latency vs. Reliability



Violating latencies negatively affects DRAM reliability

Other Factors Affecting Reliability and Latency

- Temperature
- Voltage
- Inter-cell Interference
- Manufacturing Process

To develop new mechanisms improving reliability and latency, we need to better understand the effects of these factors

Characterizing DRAM

Many of the factors affecting DRAM reliability and latency cannot be properly modeled

We need to perform experimental studies of *real* DRAM chips

Outline

1. DRAM Basics & Motivation

2. SoftMC

3. Use Cases

   - Retention Time Distribution Study

   - Evaluating Recently-Proposed Ideas

4. Future Research Directions

5. Conclusion

Goals of a DRAM Testing Infrastructure

- Flexibility
  - Ability to test *any* DRAM operation
  - Ability to test *any* combination of DRAM operations and *custom* timing parameters

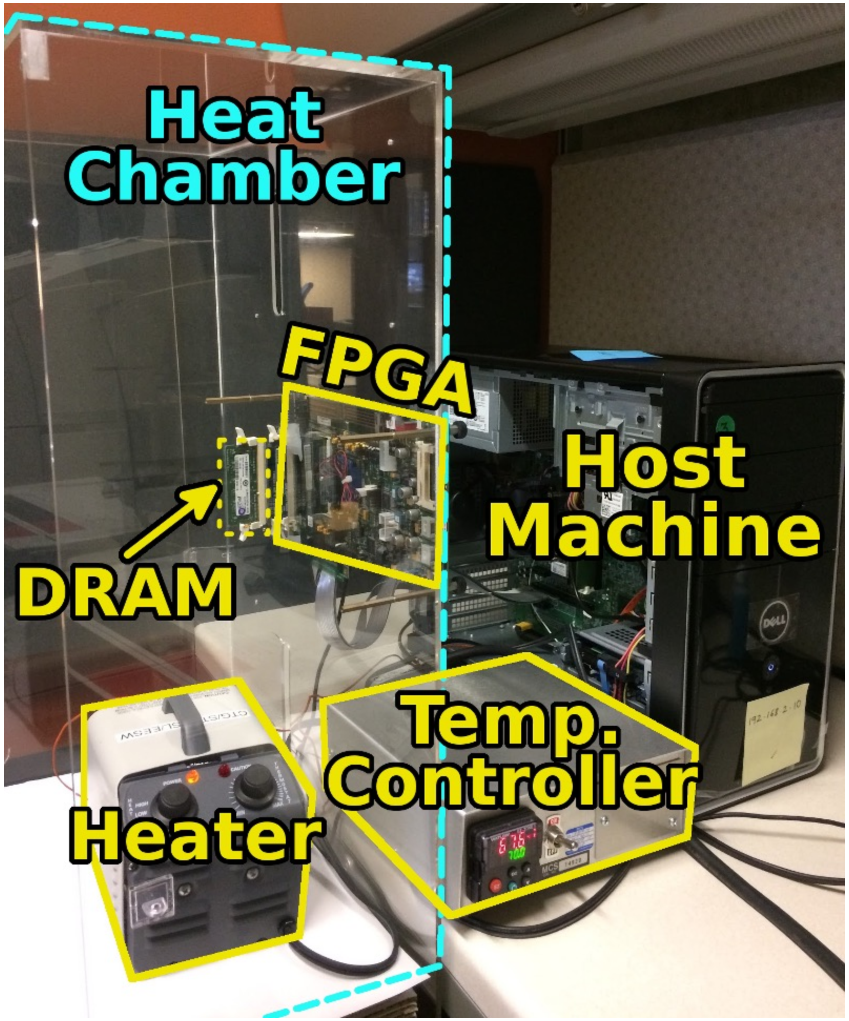
- Ease of use
  - Simple programming interface (C++)
  - Minimal programming effort and time
  - Accessible to a wide range of users
    - *who may lack experience in hardware design*

SoftMC: High-level View

FPGA-based memory characterization infrastructure

Prototype using *Xilinx ML605*

Easily programmable using the C++ API



SoftMC: Key Components

1. SoftMC API

2. PCIe Driver

3. SoftMC Hardware

SoftMC API

Writing data to DRAM:

```
InstructionSequence iseq;
iseq.insert(genACT(bank, row));
iseq.insert(genWAIT(tRCD));
iseq.insert(genWR(bank, col, data));
iseq.insert(genWAIT(tCL + tBL + tWR));
iseq.insert(genPRE(bank));
iseq.insert(genWAIT(tRP));
iseq.insert(genEND());
iseq.execute(fpga);
```

SoftMC: Key Components

1. SoftMC API

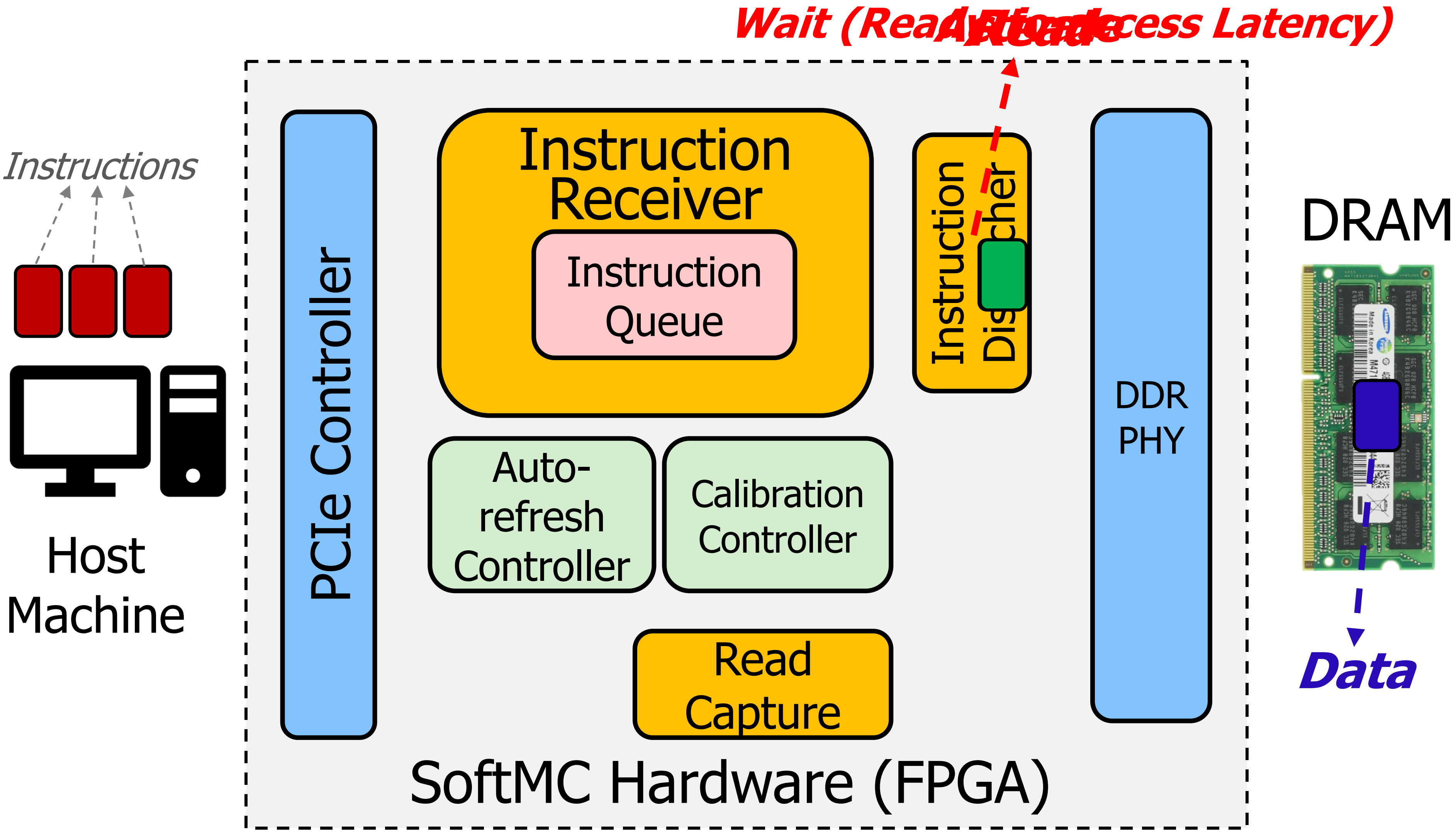
2. PCIe Driver*

Communicates raw data with the FPGA

3. SoftMC Hardware

** Jacobsen, Matthew, et al. "RIFFA 2.1: A reusable integration framework for FPGA accelerators." TRETS, 2015*

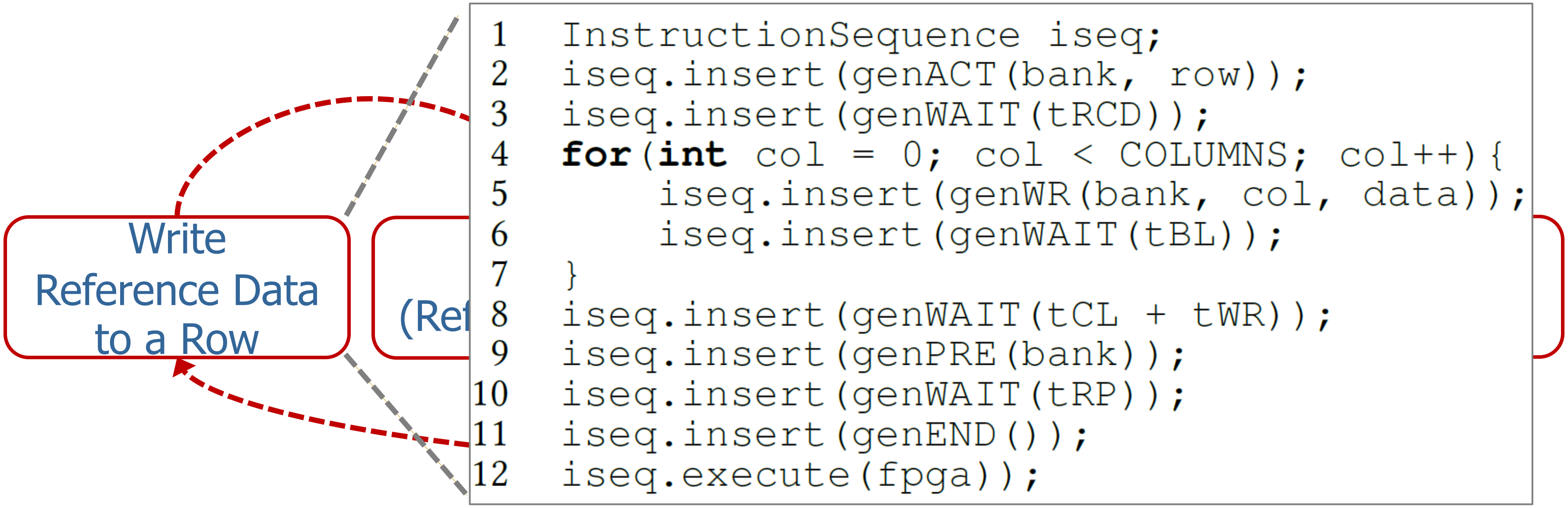
SoftMC Hardware



Outline

1. DRAM Basics & Motivation
2. SoftMC
3. Use Cases
   - Retention Time Distribution Study
   - Evaluating Recently-Proposed Ideas
4. Future Research Directions
5. Conclusion

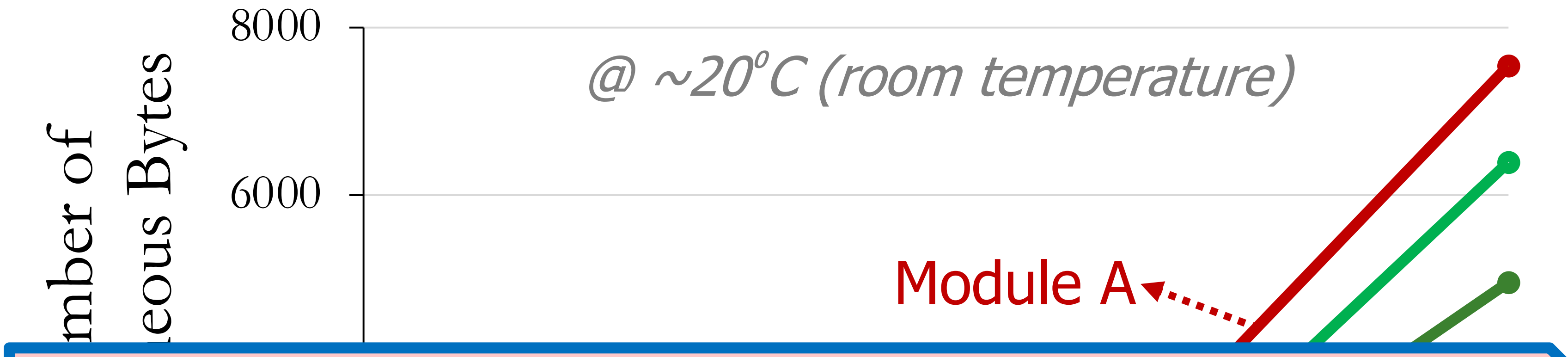
Retention Time Distribution Study



```
InstructionSequence iseq;
iseq.insert(genACT(bank, row));
iseq.insert(genWAIT(tRCD));
for(int col = 0; col < COLUMNS; col++){
    iseq.insert(genWR(bank, col, data));
    iseq.insert(genWAIT(tBL));
}
iseq.insert(genWAIT(tCL + tWR));
iseq.insert(genPRE(bank));
iseq.insert(genWAIT(tRP));
iseq.insert(genEND());
iseq.execute(fpga));
```

Can be implemented with just ~100 lines of code

Retention Time Test: Results



Validates the correctness of the SoftMC Infrastructure

Outline

1. DRAM Basics & Motivation
2. SoftMC
3. Use Cases
   - Retention Time Distribution Study
   - Evaluating Recently-Proposed Ideas
4. Future Research Directions
5. Conclusion

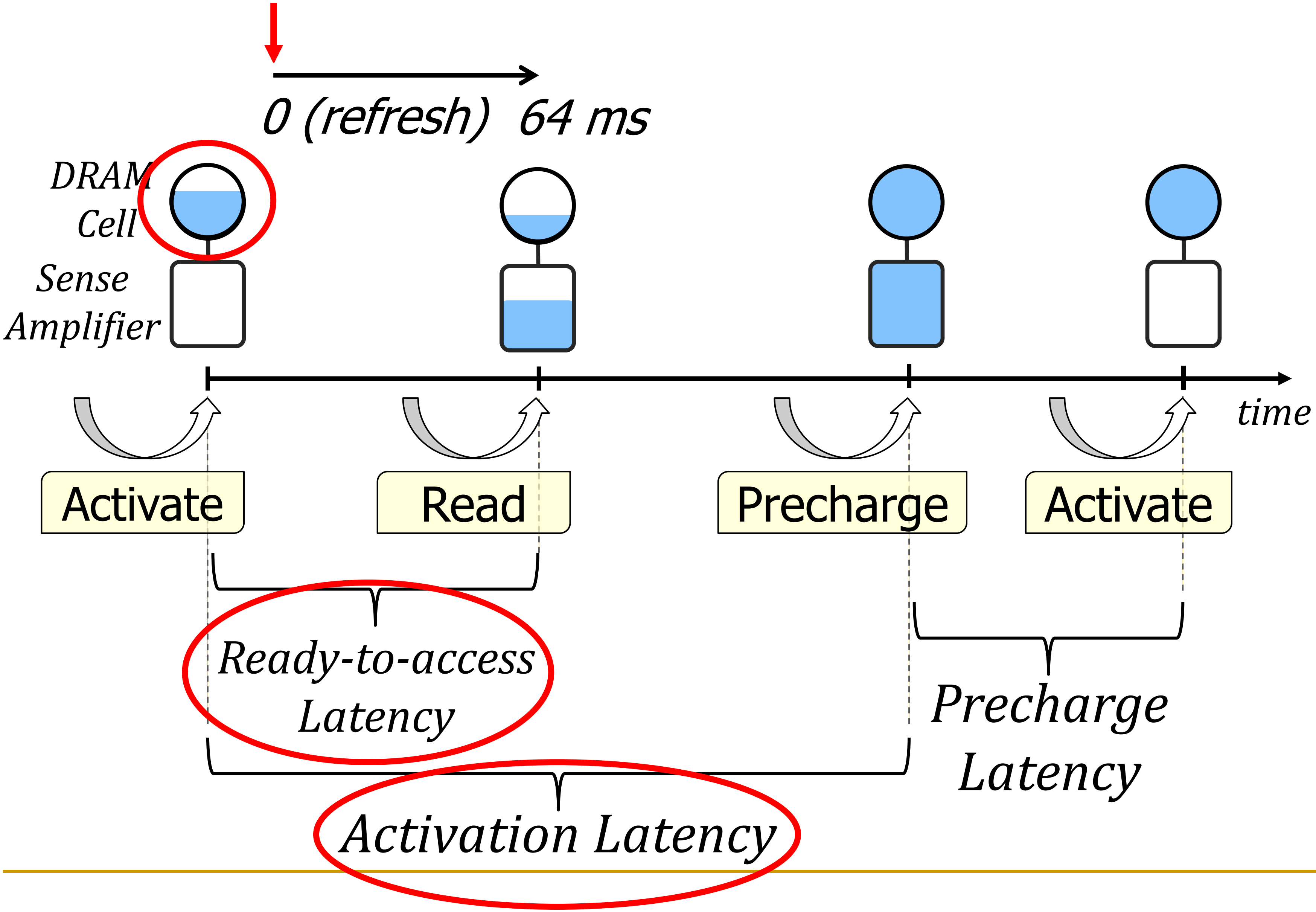
Accessing Highly-charged Cells Faster

NUAT
(Shin+, HPCA 2014)

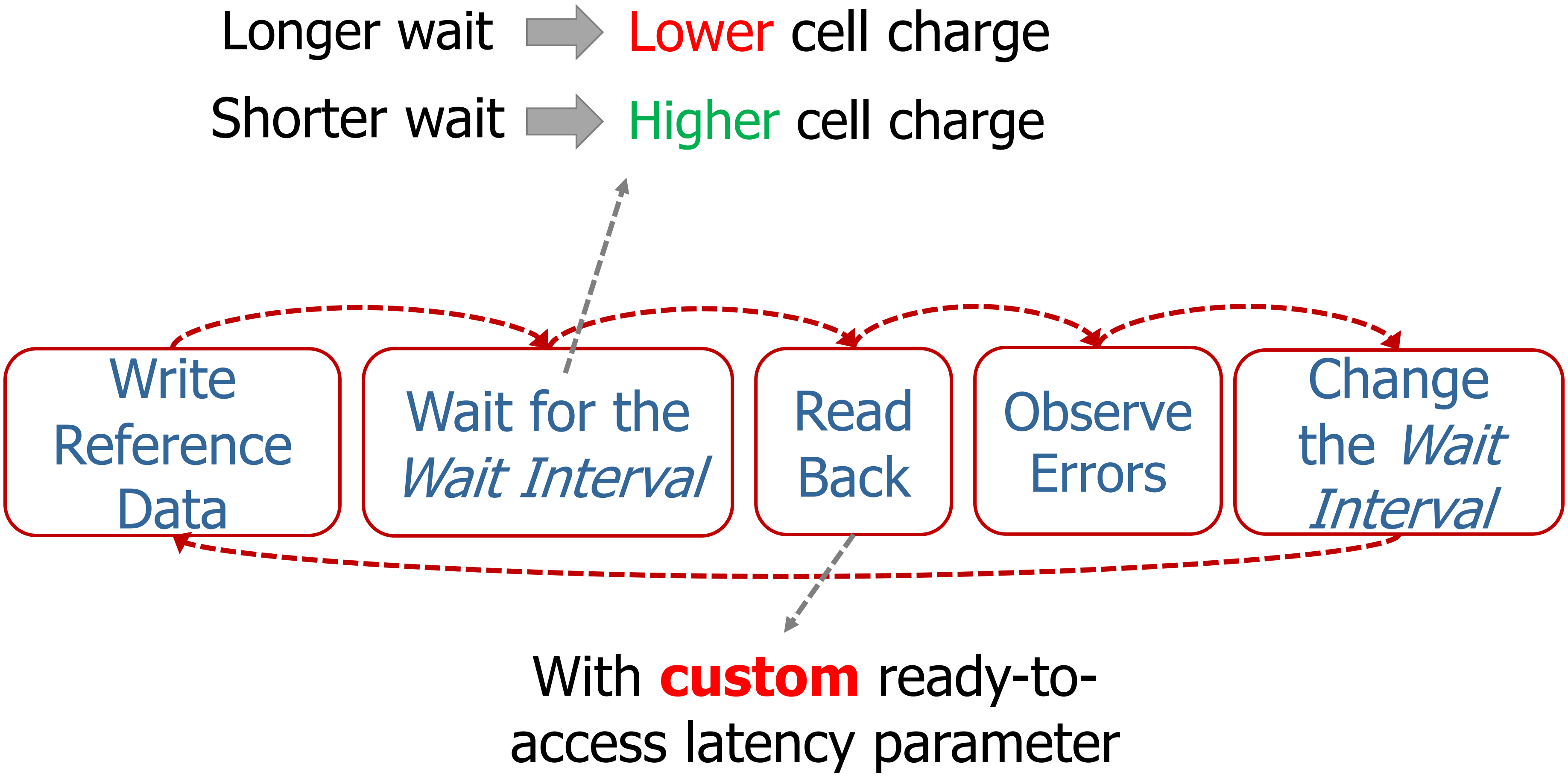
ChargeCache
(Hassan+, HPCA 2016)

A highly-charged cell can be accessed with low latency

How a Highly-Charged Cell Is Accessed Faster?



Ready-to-access Latency Test



Longer wait → Lower cell charge

Shorter wait → Higher cell charge

Write Reference Data

Wait for the *Wait Interval*

Read Back

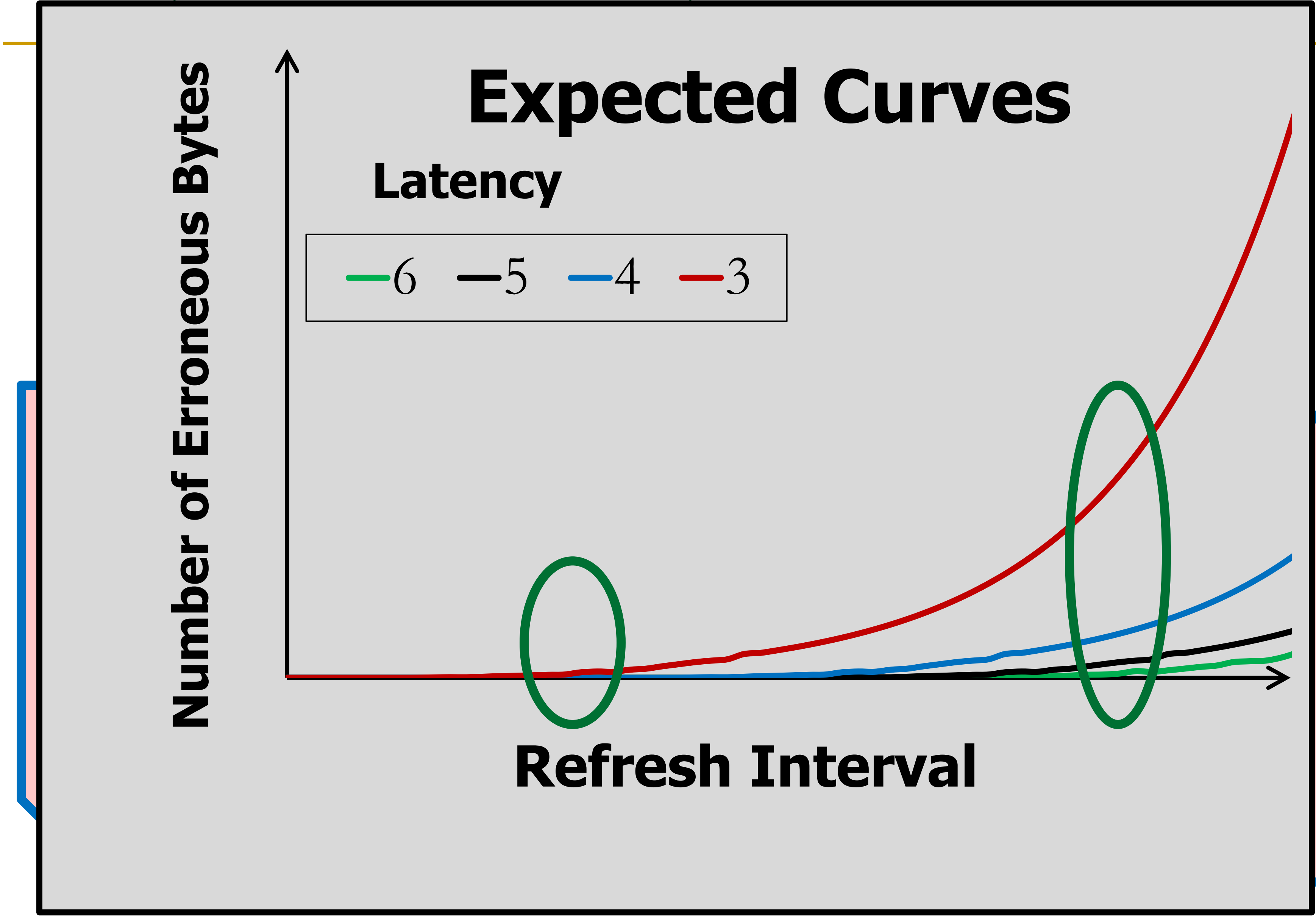
Observe Errors

Change the *Wait Interval*

With **custom** ready-to-access latency parameter

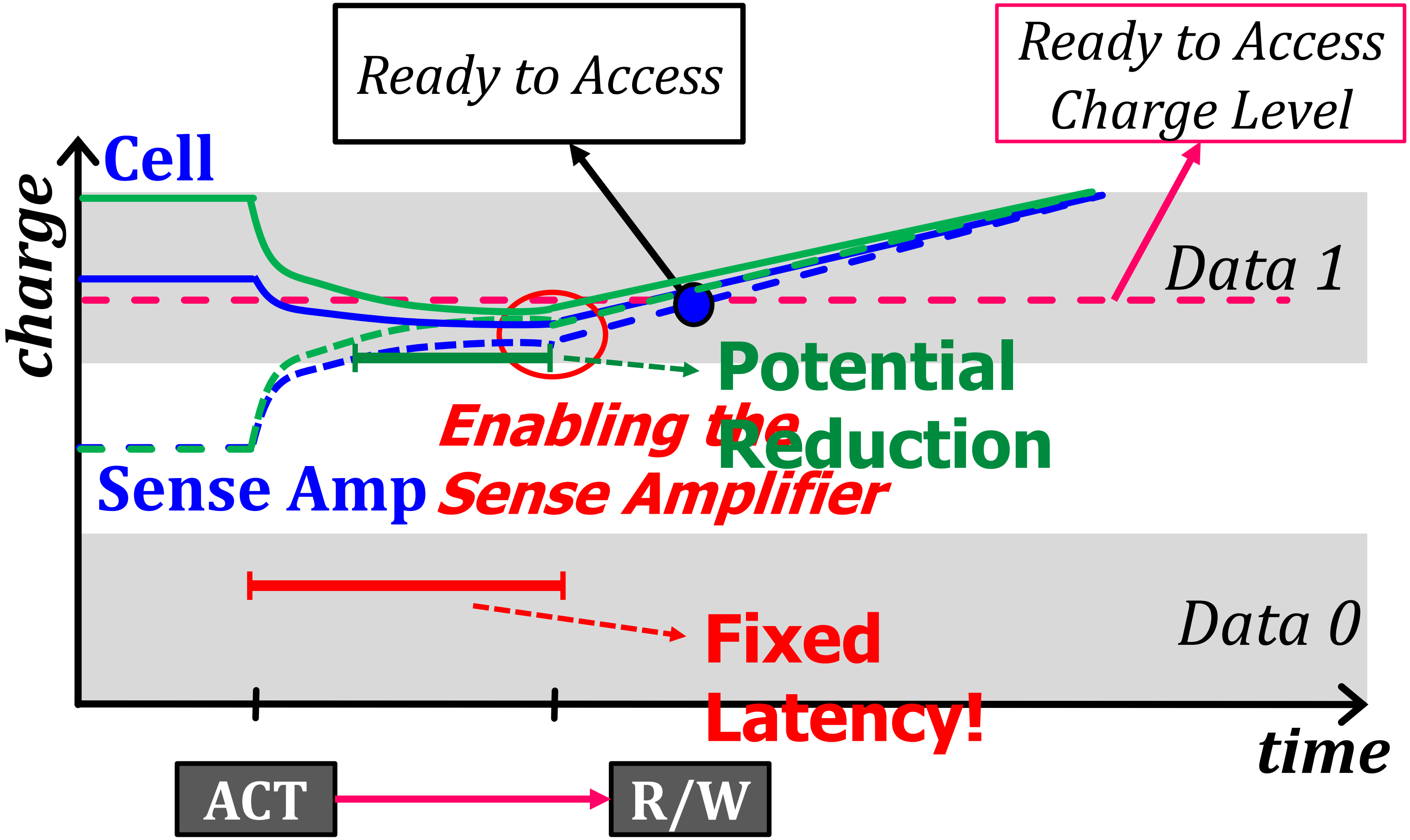
<u>Can be implemented with just ~150 lines of code</u>

Ready-to-access Latency: Results



Why Don't We See the Latency Reduction Effect?

- The memory controller cannot externally control when a sense amplifier gets enabled in existing DRAM chips



Outline

1. DRAM Basics & Motivation
2. SoftMC
3. Use Cases
   - Retention Time Distribution Study
   - Evaluating Recently-Proposed Ideas
4. Future Research Directions
5. Conclusion

Future Research Directions

- More Characterization of DRAM
  - How are the cell characteristics changing with different generations of technology nodes?
  - What types of usage accelerate aging?
- Characterization of Non-volatile Memory
- Extensions
  - Memory Scheduling
  - Workload Analysis
  - Testbed for in-memory Computation

Outline

1. DRAM Basics & Motivation
2. SoftMC
3. Use Cases
   - Retention Time Distribution Study
   - Evaluating Recently-Proposed Ideas
4. Future Research Directions
5. Conclusion

Conclusion

- **SoftMC**: First publicly-available FPGA-based DRAM testing infrastructure
- Flexible and Easy to Use
- Implemented two use cases
  - Retention Time Distribution Study
  - Evaluation of two recently-proposed latency reduction mechanisms
- SoftMC can enable many other studies, ideas, and methodologies in the design of future memory systems
- **Download** our first prototype

github.com/CMU-SAFARI/SoftMC

SoftMC

A Flexible and Practical Open-Source Infrastructure for Enabling Experimental DRAM Studies

Hasan Hassan,
Nandita Vijaykumar, Samira Khan,
Saugata Ghose, Kevin Chang,
Gennady Pekhimenko, Donghyuk Lee,
Oguz Ergin, Onur Mutlu

ETH zürich

Carnegie Mellon

TOBB
UNIVERSITY OF
ECONOMICS AND TECHNOLOGY

P&S DRAM Bender

FPGA-based Exploration of DRAM and RowHammer

Ataberk Olgun
Prof. Onur Mutlu
ETH Zürich

Fall 2022
4 October 2022