An Experimental Analysis of RowHammer in HBM2 DRAM Chips

Ataberk Olgun Majd Osseiran

A. Giray Yağlıkçı Yahya Can Tuğrul Haocong Luo Steve Rhyner

Behzad Salami Juan Gomez Luna Onur Mutlu

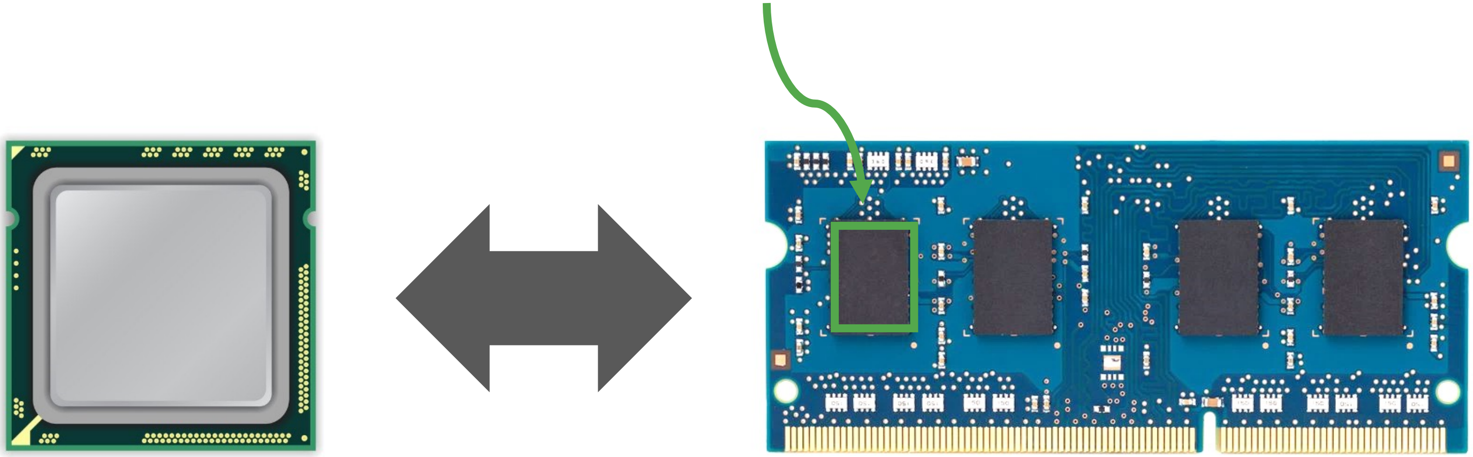
SAFARI

ETH zürich

AMERICAN UNIVERSITY OF BEIRUT

The RowHammer Vulnerability (I)

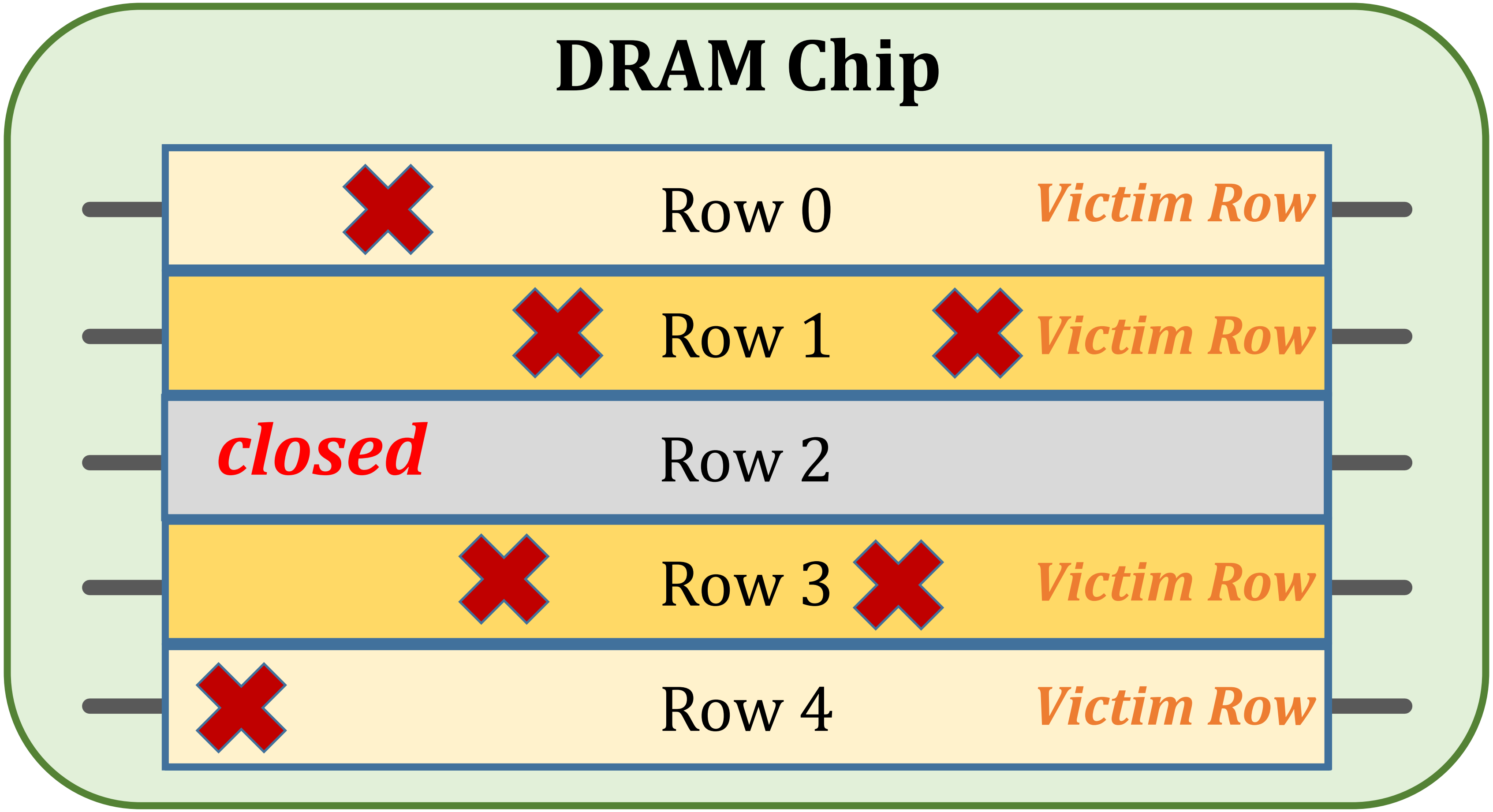
DRAM Chip



CPU

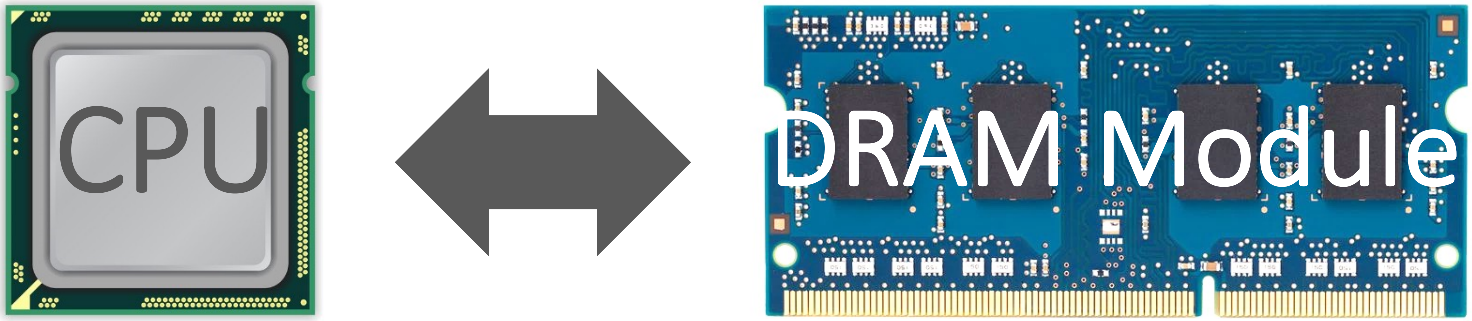
DRAM Module

The RowHammer Vulnerability (II)

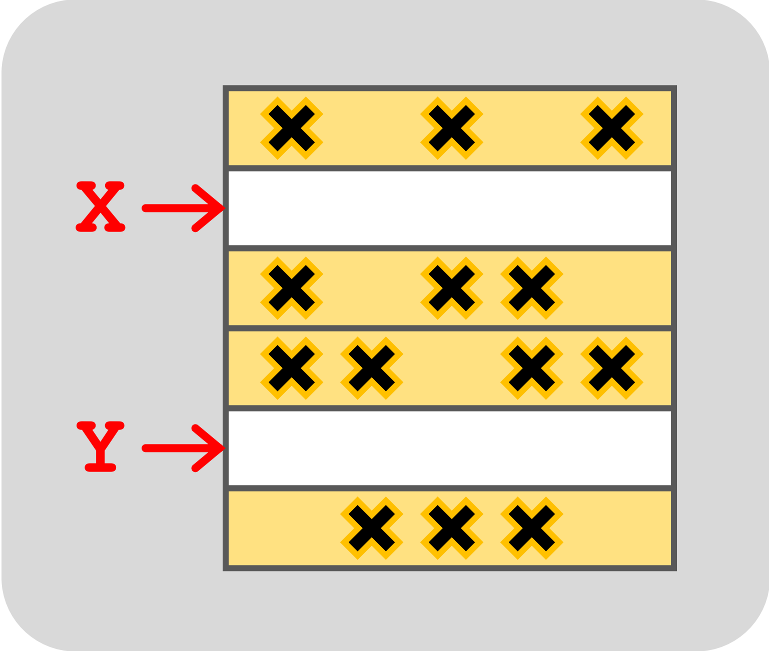


Repeatedly **opening** (activating) and **closing** (precharging) a DRAM row causes **RowHammer bit flips** in nearby rows

A Simple Program Can Induce Bitflips

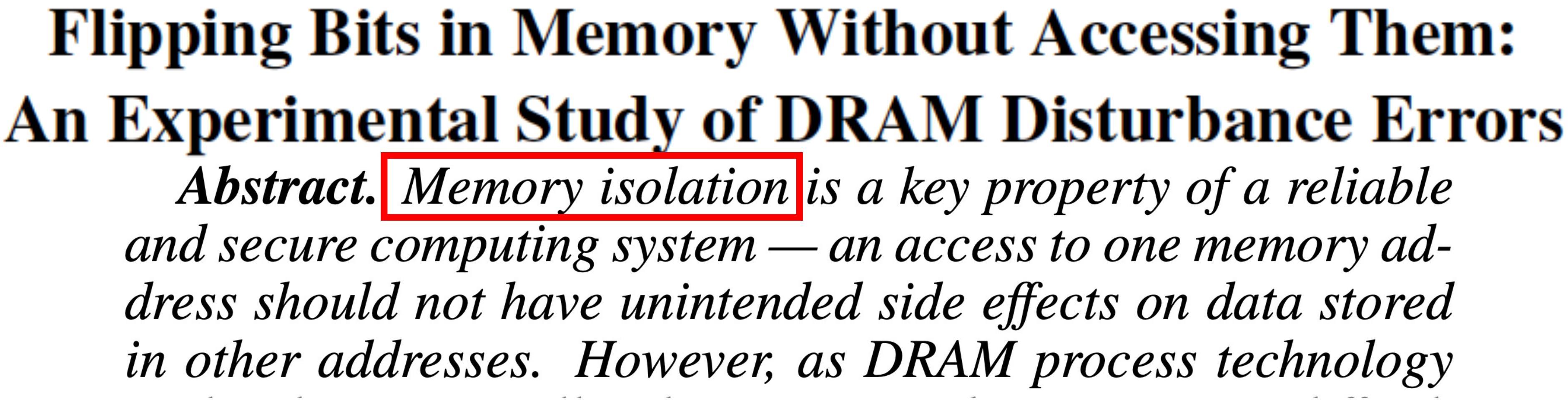


```
loop:
   mov  (X), %eax
   mov  (Y), %ebx
   clflush (X)
   clflush (Y)
   mfence
   jmp loop
```



https://github.com/CMU-SAFARI/rowhammer

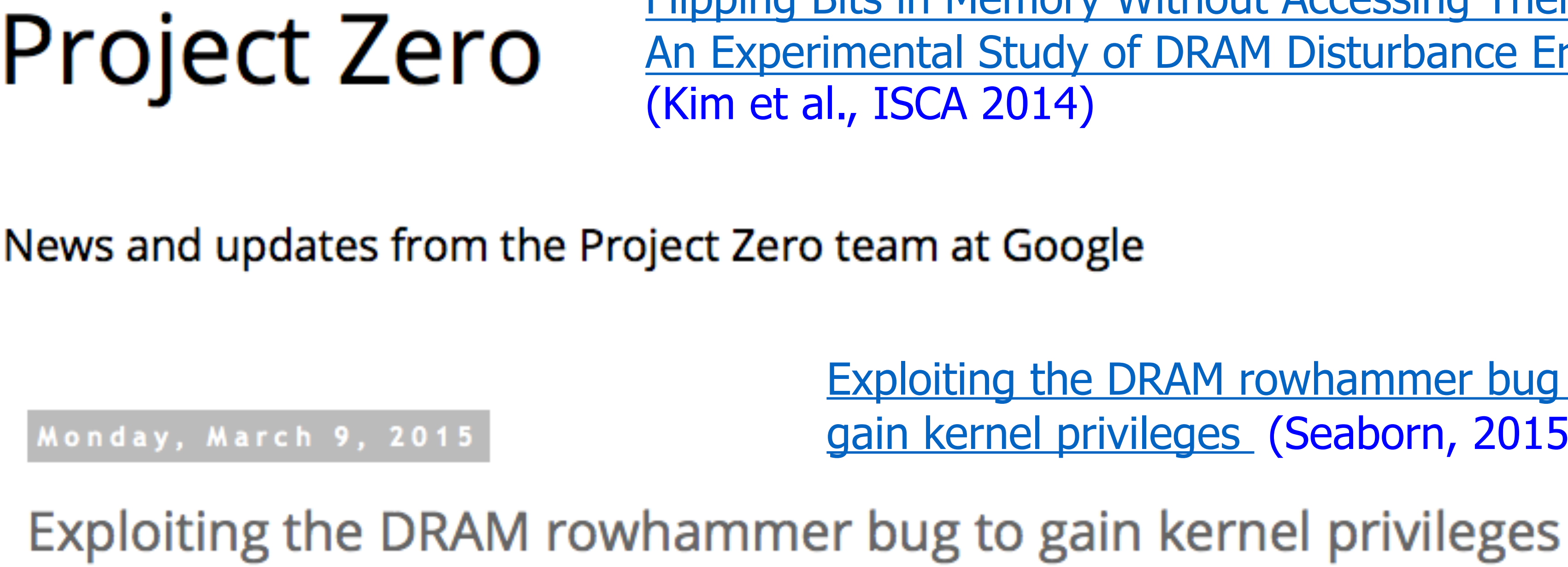
One Can Take Over a System



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

Abstract. *Memory isolation is a key property of a reliable and secure computing system — an access to one memory address should not have unintended side effects on data stored in other addresses. However, as DRAM process technology*

Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors (Kim et al., ISCA 2014)



Project Zero

News and updates from the Project Zero team at Google

Monday, March 9, 2015

Exploiting the DRAM rowhammer bug to gain kernel privileges

Exploiting the DRAM rowhammer bug to gain kernel privileges (Seaborn, 2015)

Most DRAM Modules Are Vulnerable (2020)

DRAM type-node	Number of Chips (Modules) Tested			
	Mfr. A	*Mfr. B*	*Mfr. C*	*Total*
DDR3-old	56 (10)	88 (11)	28 (7)	**172 (28)**
DDR3-new	80 (10)	52 (9)	104 (13)	**236 (32)**
DDR4-old	112 (16)	24 (3)	128 (18)	**264 (37)**
DDR4-new	264 (43)	16 (2)	108 (28)	**388 (73)**
LPDDR4-1x	12 (3)	180 (45)	N/A	**192 (48)**
LPDDR4-1y	184 (46)	N/A	144 (36)	**328 (82)**

All tested DRAM types are susceptible to RowHammer bitflips

What about High Bandwidth Memory (HBM)?

SAFARI

Kim et al., "Revisiting RowHammer: An Experimental Analysis of Modern DRAM Devices and Mitigation Techniques," in ISCA, 2020.

Executive Summary

Motivation: HBM chips have new architectural characteristics (e.g., 3D-stacked dies) that might affect the RowHammer vulnerability in various ways

Understanding RowHammer enables designing effective and efficient solutions

Problem: No prior study demonstrates the RowHammer vulnerability in HBM

Goal: Experimentally analyze how vulnerable HBM DRAM chips are to RowHammer

Experimental Study: Detailed experimental characterization of RowHammer in a modern HBM2 DRAM chip. Our study provides two main findings:

1. Spatial variation of RowHammer vulnerability

- Different channels in a 3D-stacked HBM chip exhibit different RowHammer vulnerability
- DRAM rows near the end of a DRAM bank are more RowHammer resilient

2. On-DRAM-die RowHammer mitigations

- A modern HBM chip implements undisclosed on-DRAM-die RowHammer mitigation
- The mitigation refreshes a victim row after every 17 periodic refresh operations (e.g., similar to DDR4 chips)

Outline

1. HBM DRAM Organization & Operation

2. DRAM Cell Leakage & RowHammer

3. HBM DRAM Testing Methodology

4. RowHammer Spatial Variation Analysis

5. On-die RowHammer Mitigation Analysis

6. Conclusion

Outline

1. HBM DRAM Organization & Operation

2. DRAM Cell Leakage & RowHammer

3. HBM DRAM Testing Methodology

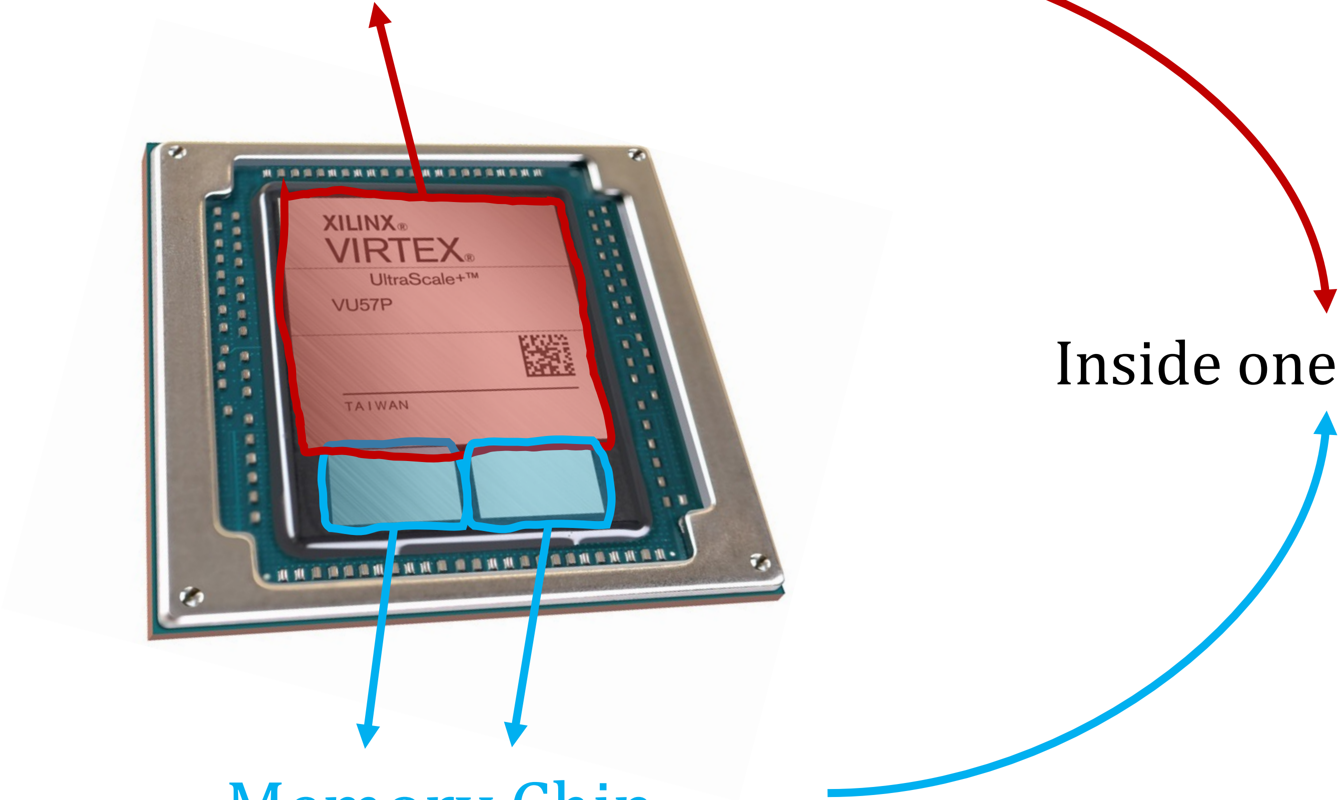
4. RowHammer Spatial Variation Analysis

5. On-die RowHammer Mitigation Analysis

6. Conclusion

System with High Bandwidth Memory

Compute Chip (e.g., FPGA)

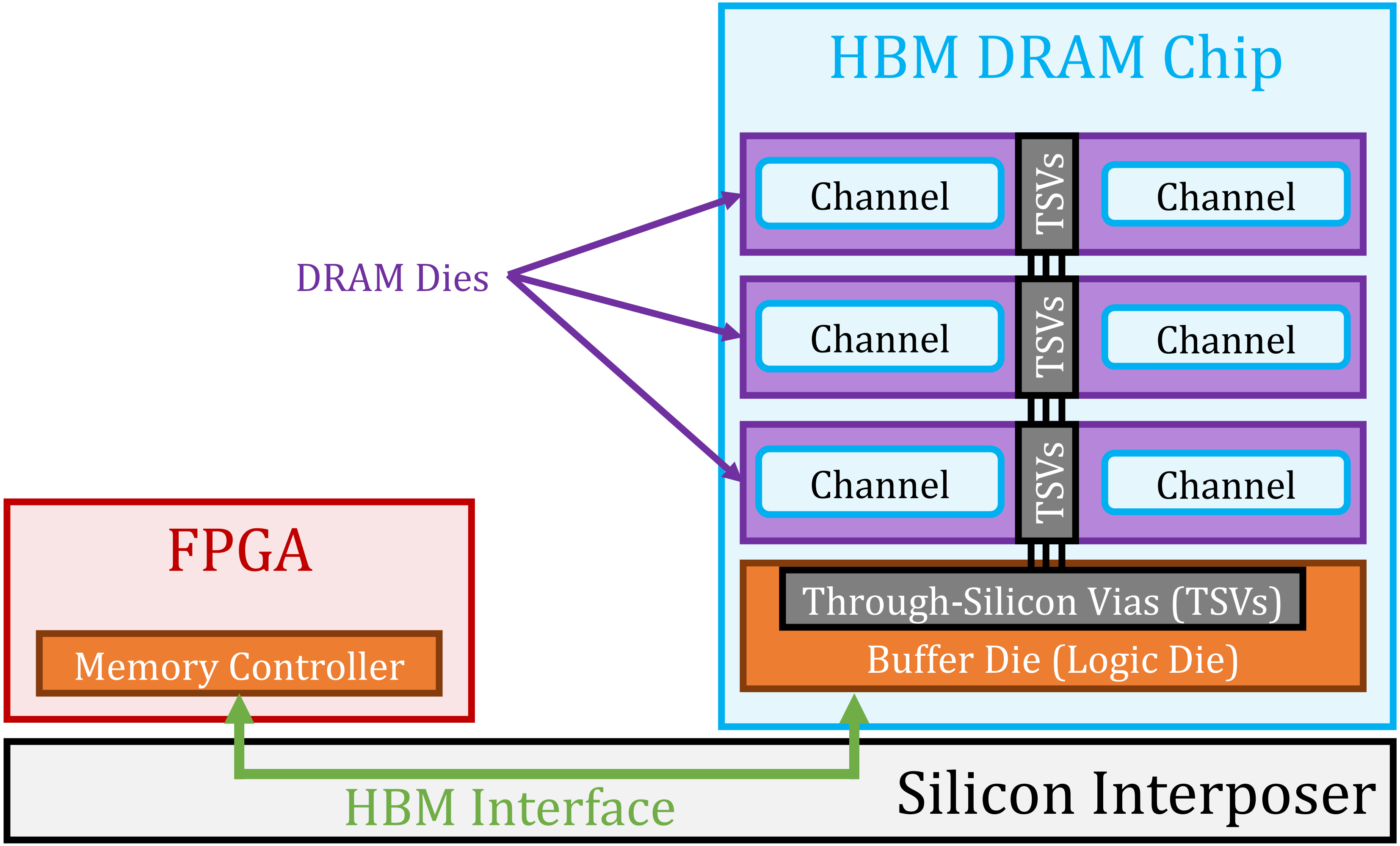


Inside one **package**

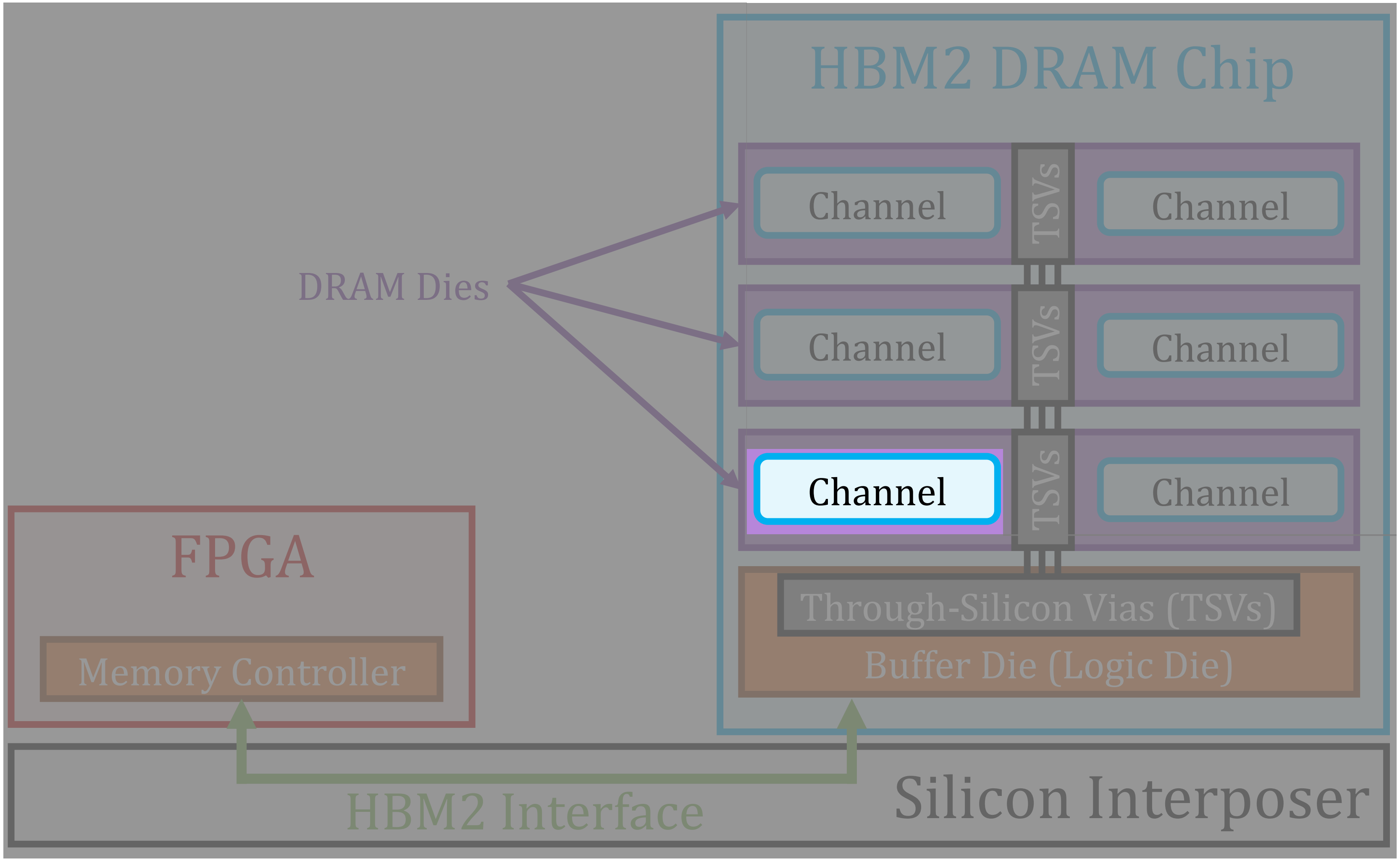
Memory Chip
(e.g., HBM DRAM)

https://www.hpcwire.com/off-the-wire/xilinx-adds-high-bandwidth-memory-capabilities-to-its-virtex-ultrascale-portfolio/

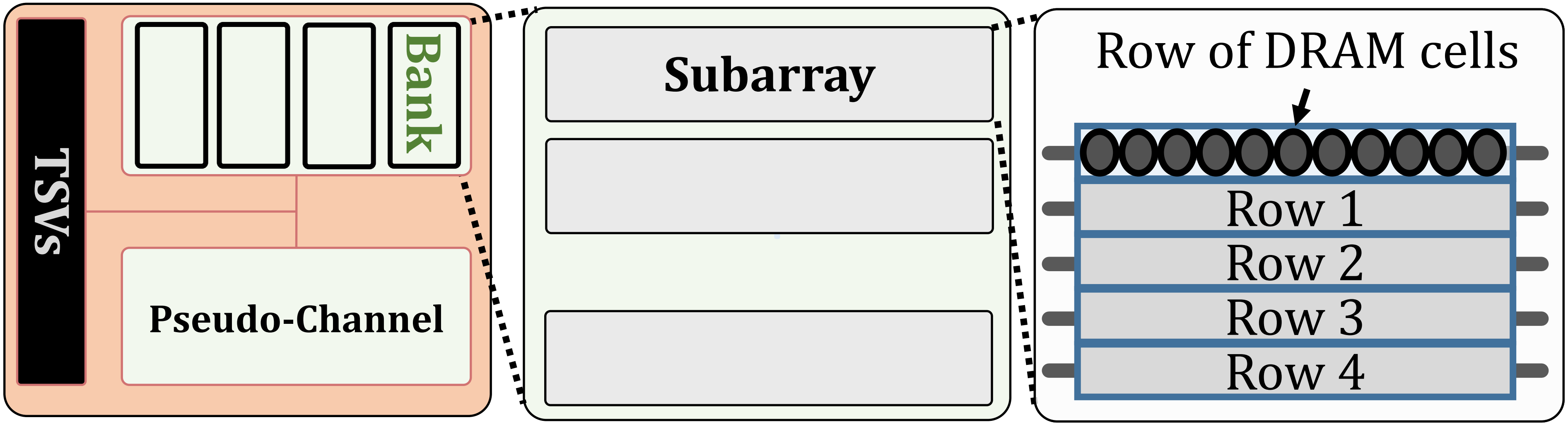
HBM DRAM Organization (I)



HBM DRAM Organization (I)



HBM DRAM Organization (II)



DRAM Channel DRAM Bank DRAM Subarray

Outline

1. HBM DRAM Organization & Operation

2. DRAM Cell Leakage & RowHammer

3. HBM DRAM Testing Methodology

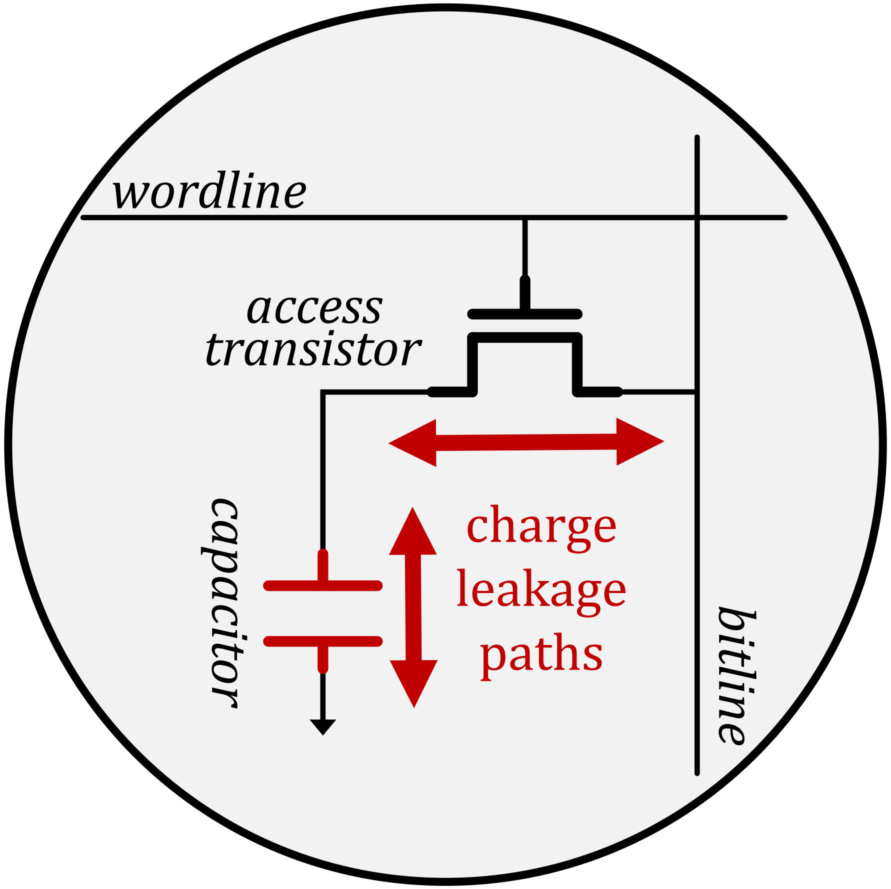
4. RowHammer Spatial Variation Analysis

5. On-die RowHammer Mitigation Analysis

6. Conclusion

DRAM Cell Leakage

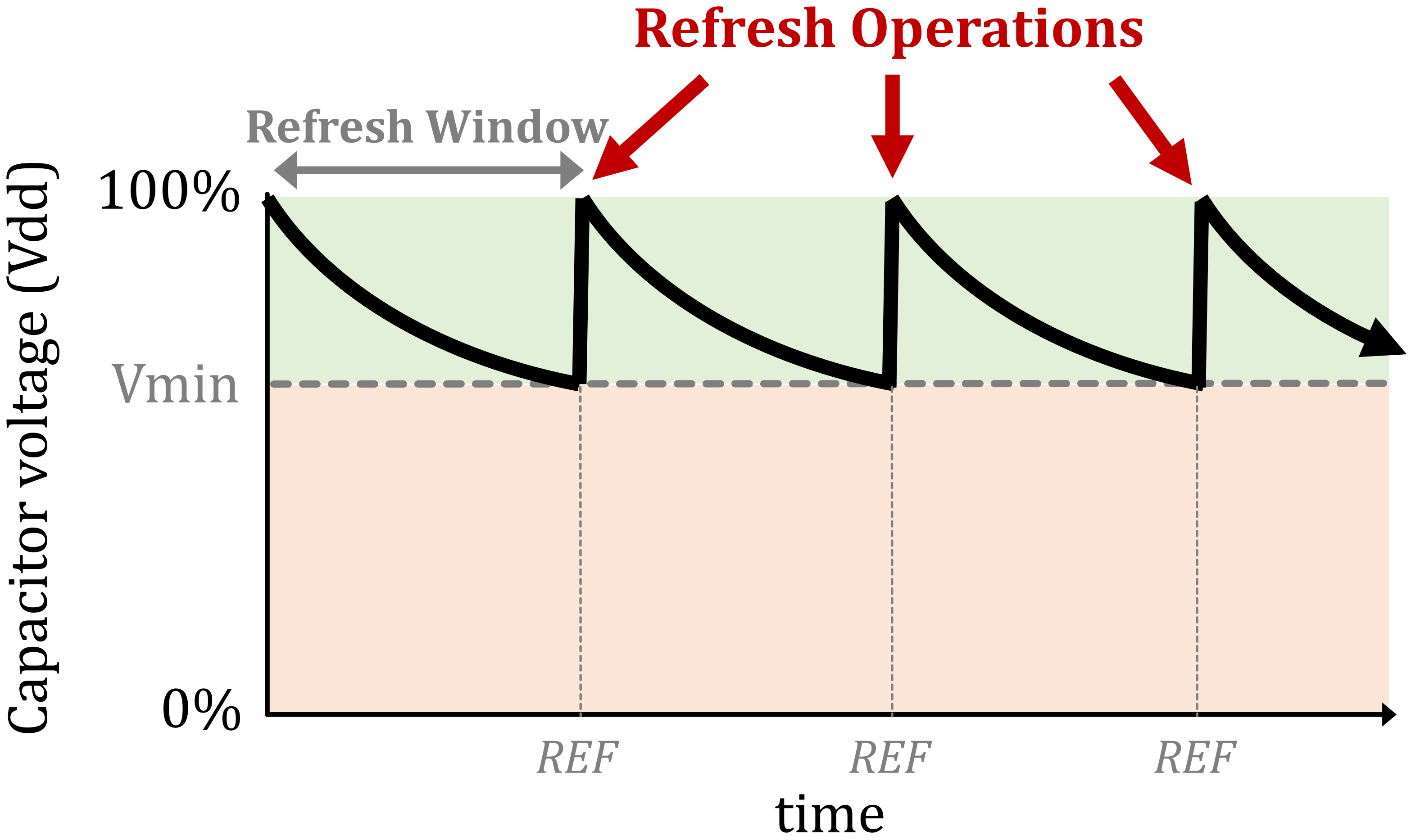
Each cell encodes information in **leaky** capacitors



Stored data is **corrupted** if too much charge leaks (i.e., the capacitor voltage degrades too much)

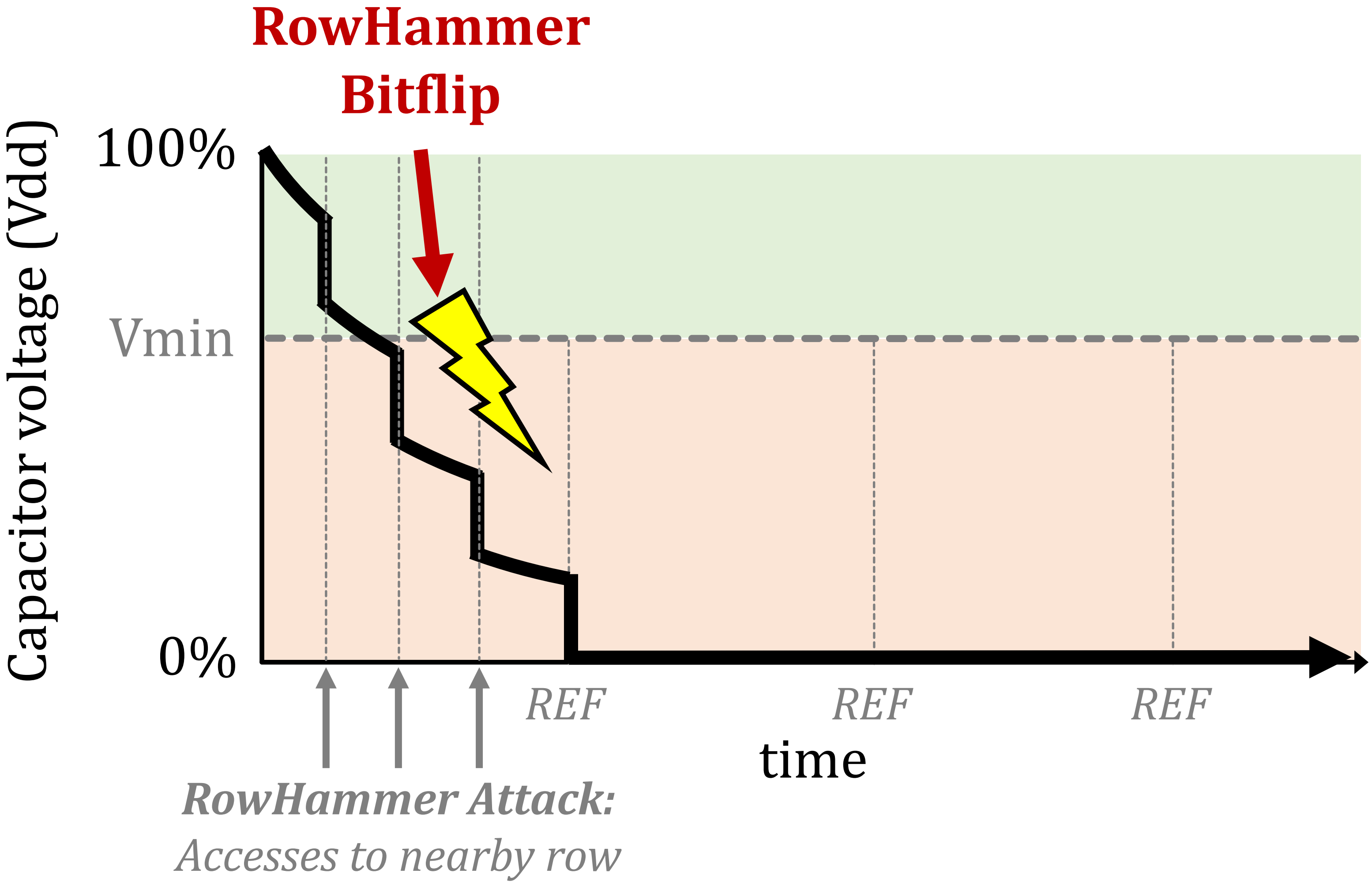
[Patel+, ISCA'17]

DRAM Refresh



Periodic **refresh operations** preserve stored data

RowHammer Bitflips



Problem & Goal

Problem

No prior study demonstrates the RowHammer vulnerability in high bandwidth memory

Our Goal

Experimentally analyze how vulnerable real high bandwidth memory chips are to RowHammer

Outline

1. HBM DRAM Organization & Operation

2. DRAM Cell Leakage & RowHammer

3. HBM DRAM Testing Methodology

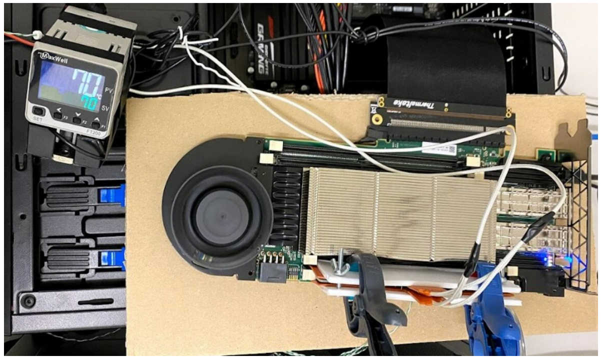
4. RowHammer Spatial Variation Analysis

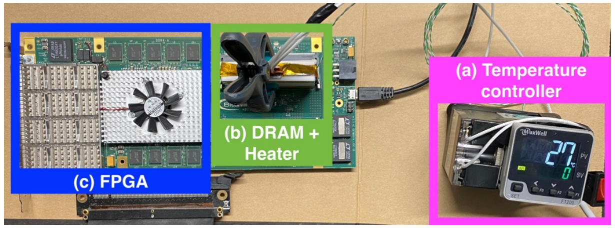
5. On-die RowHammer Mitigation Analysis

6. Conclusion

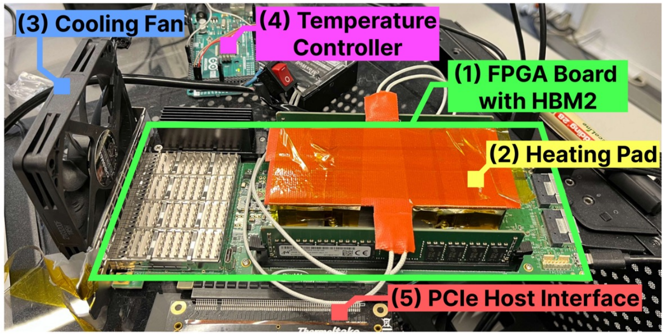
DRAM Testing Infrastructure

DRAM Bender DDR3/4 Testing Infrastructure

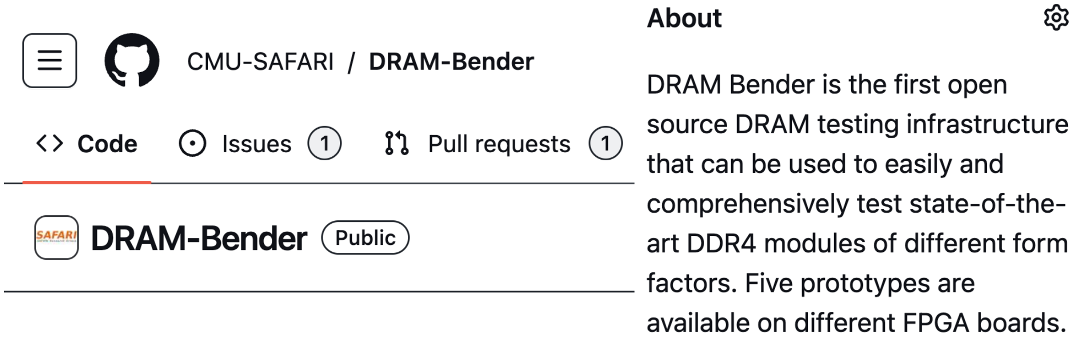




Adapt to work with HBM2 chips



https://github.com/CMU-SAFARI/DRAM-Bender



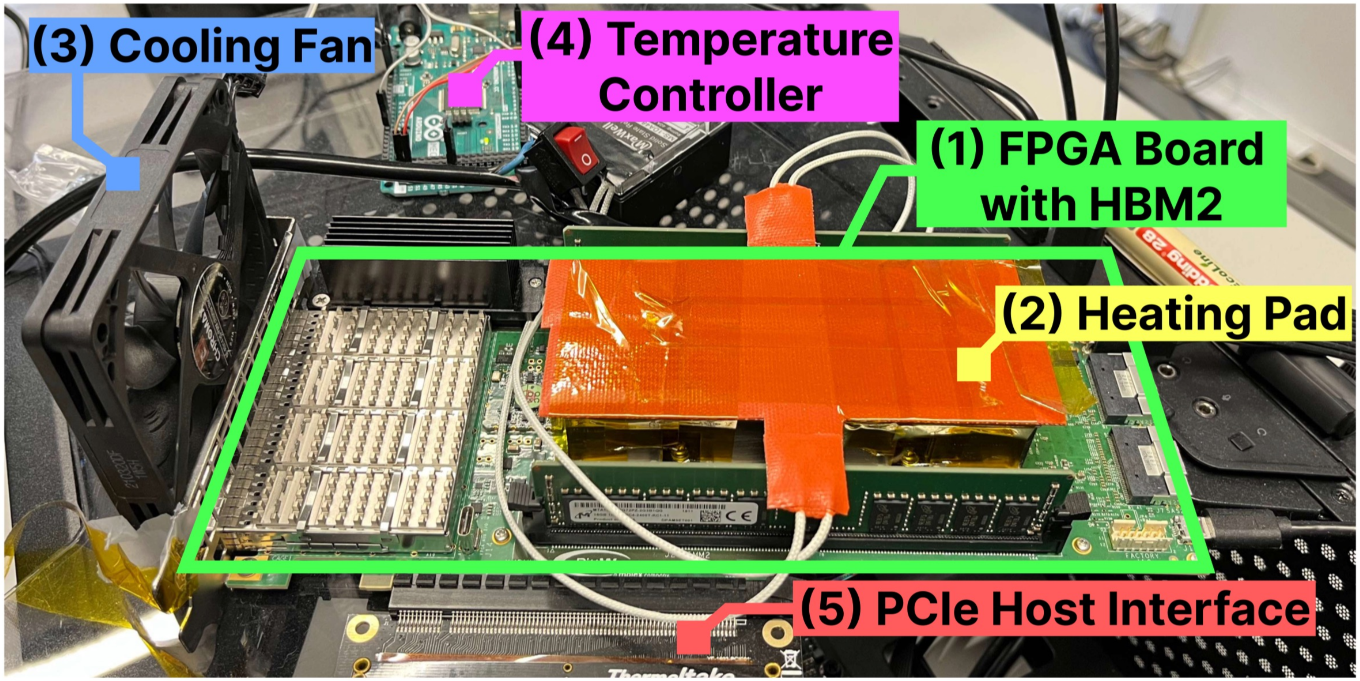
About

DRAM Bender is the first open source DRAM testing infrastructure that can be used to easily and comprehensively test state-of-the-art DDR4 modules of different form factors. Five prototypes are available on different FPGA boards.

Olgun et al., "[DRAM Bender: An Extensible and Versatile FPGA-based Infrastructure to Easily Test State-of-the-art DRAM Chips](#)," in TCAD, 2023.

DRAM Testing Infrastructure

FPGA-based HBM2 Testing Setup (Bittware XUPVVH)



Fine-grained control over **DRAM commands**,
timing parameters (±1.66ns)

Olgun et al., "DRAM Bender: An Extensible and Versatile FPGA-based Infrastructure to Easily Test State-of-the-art DRAM Chips," in TCAD, 2023.

RowHammer Testing Methodology (I)

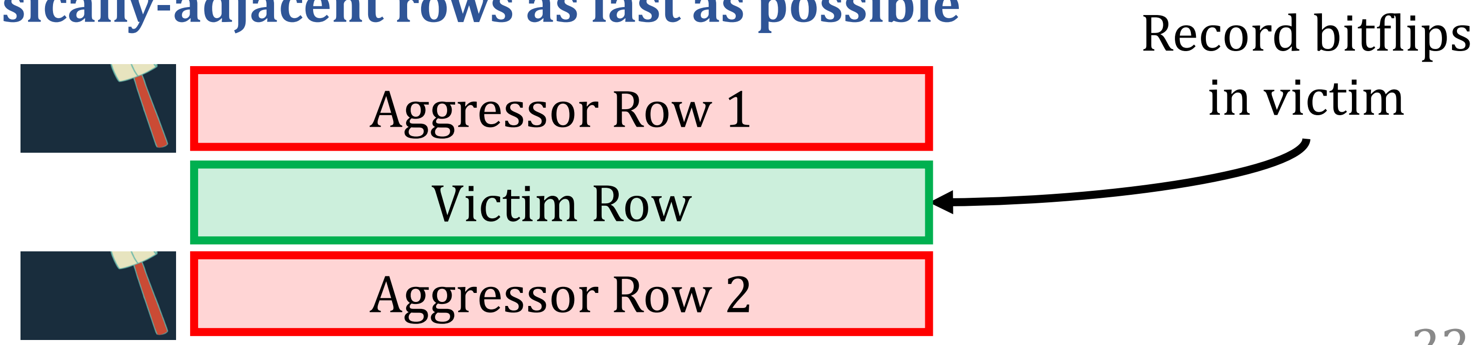
To characterize our DRAM chips at **worst-case** conditions:

1. Prevent sources of interference during core test loop

- **No DRAM refresh**: to avoid refreshing victim row
- **No RowHammer mitigation mechanisms**: to observe circuit-level effects
- Test for **less than a refresh window (32ms)** to avoid retention failures
- **Repeat tests** for five times

2. Worst-case RowHammer access sequence

- We use **worst-case** RowHammer access sequence based on prior works' observations
- Double-sided RowHammer: **repeatedly access the two physically-adjacent rows as fast as possible**



Record bitflips in victim

Aggressor Row 1

Victim Row

Aggressor Row 2

RowHammer Testing Methodology (II)

- Tested HBM2 chip's organization:
  - 8 channels
  - 2 pseudo-channels
  - 16 banks
  - 16384 rows (1 KiB each)

Xilinx FPGA
with HBM2 DRAM chips

- Test all channels, pseudo-channels, banks

- Test first, middle, and last 3K rows in a bank
  - 9K out of 16K (more than half)

- Keep HBM2 chip temperature at 85°C

Metrics

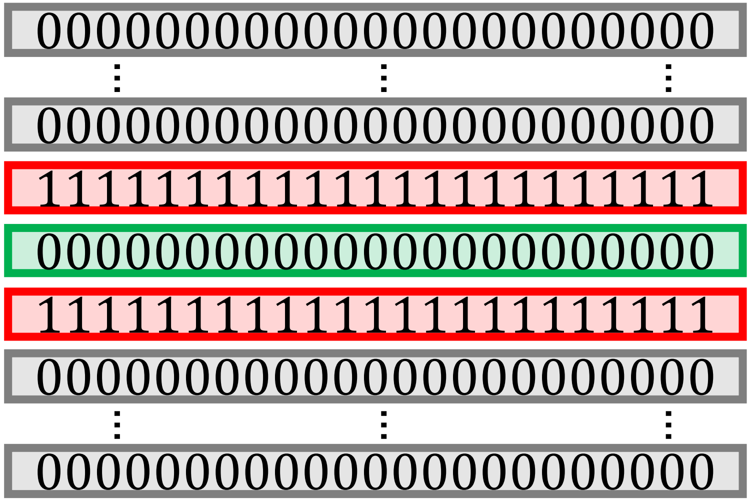
1. **Bit error rate (BER):**
   The fraction of DRAM cells in a row
   that experience a bitflip after 512K activations

   Higher is worse

2. **Hammer Count for the First Bitflip (HC_{first}):**
   Aggressor row activation count
   to cause the first bitflip in the victim row

   Lower is worse

Tested Data Patterns



00000000000000000000000

⋮ ⋮ ⋮

00000000000000000000000

11111111111111111111111

00000000000000000000000

11111111111111111111111

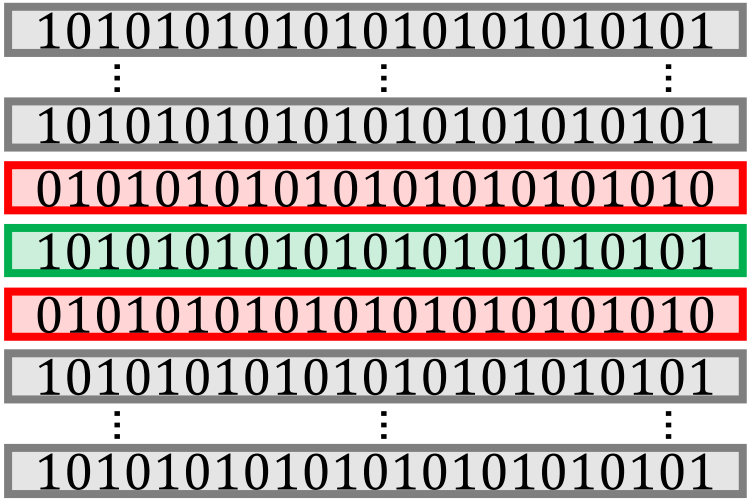
00000000000000000000000

⋮ ⋮ ⋮

00000000000000000000000

Row Addresses	***Rowstripe0***	***Rowstripe1***	***Checkered0***	***Checkered1***
Victim (V)	0x00	0xFF	0x55	0xAA
Aggressors (V ± 1)	0xFF	0x00	0xAA	0x55
V ± [2:8]	0x00	0xFF	0x55	0xAA

Tested Data Patterns



10101010101010101010101

⋮ ⋮ ⋮

10101010101010101010101

01010101010101010101010

10101010101010101010101

01010101010101010101010

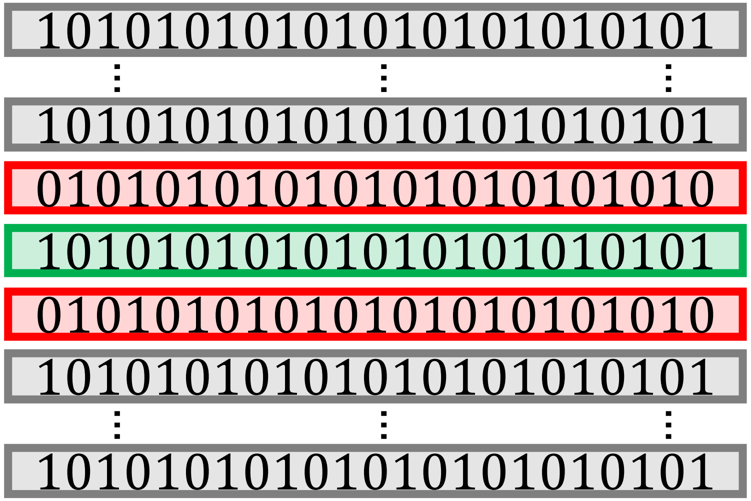
10101010101010101010101

⋮ ⋮ ⋮

10101010101010101010101

Row Addresses	***Rowstripe0***	***Rowstripe1***	***Checkered0***	***Checkered1***
Victim (V)	0x00	0xFF	0x55	0xAA
Aggressors (V ± 1)	0xFF	0x00	0xAA	0x55
V ± [2:8]	0x00	0xFF	0x55	0xAA

Tested Data Patterns



10101010101010101010101
⋮
10101010101010101010101
01010101010101010101010
10101010101010101010101
01010101010101010101010
10101010101010101010101
⋮
10101010101010101010101

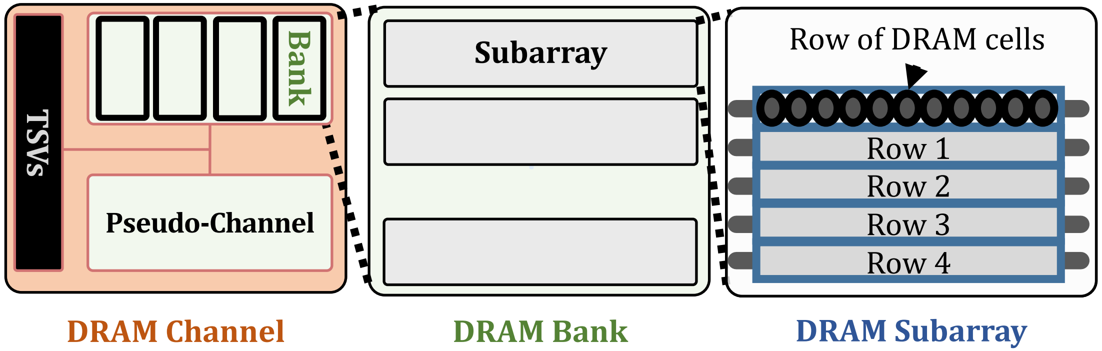
Row Addresses	***Rowstripe0***	***Rowstripe1***	***Checkered0***	***Checkered1***
Victim (V)	0x00	0xFF	0x55	0xAA
Aggressors (V ± 1)	0xFF	0x00	0xAA	0x55
V ± [2:8]	0x00	0xFF	0x55	0xAA

Worst-case data pattern (WCDP) of a row: Causes smallest HC_{first} for a row

Two Main Analyses

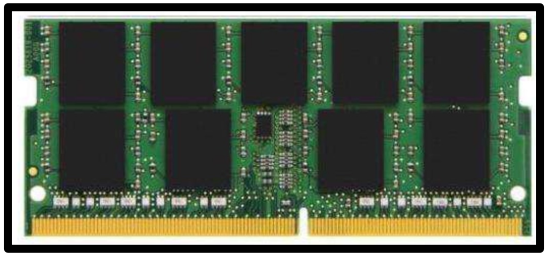
1. Spatial variation of RowHammer vulnerability

How does the RowHammer vulnerability change across channels, pseudo-channels, banks, rows in HBM?



2. On-DRAM-die RowHammer mitigations

Do real HBM chips implement undisclosed RowHammer mitigations resembling those that exist in DDR4?



Outline

1. HBM DRAM Organization & Operation

2. DRAM Cell Leakage & RowHammer

3. HBM DRAM Testing Methodology

4. RowHammer Spatial Variation Analysis

5. On-die RowHammer Mitigation Analysis

6. Conclusion

Key Takeaways from Spatial Variation Analysis

Takeaway 1

Different 3D-stacked HBM2 channels exhibit different RowHammer vulnerability

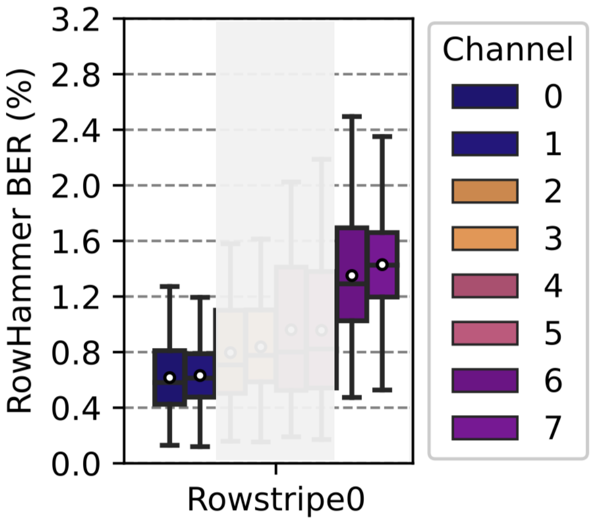
Takeaway 2

DRAM rows near the end of a DRAM bank experience smaller bit error rate (BER) than others

Takeaway 3

Activation count needed to induce the first RowHammer bitflip (HC_{first}) changes with the data pattern and the physical location of the DRAM row

Spatial Distribution of BER (I)

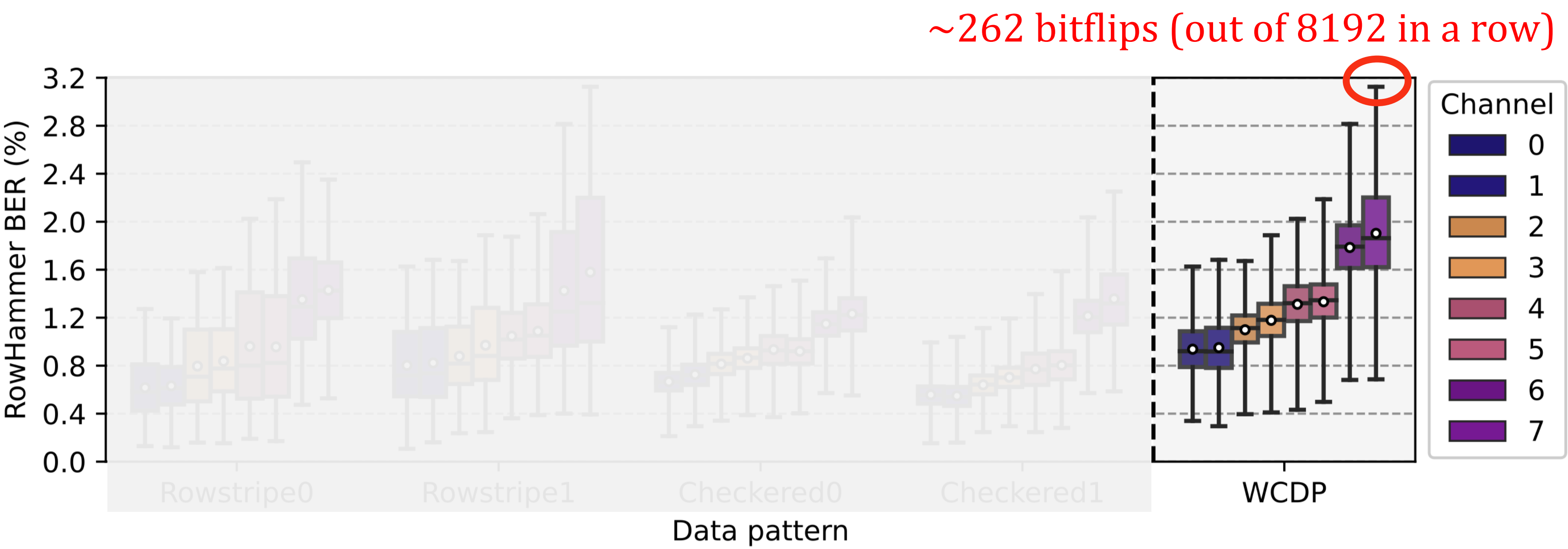


There are bitflips in every tested DRAM row across all tested HBM2 channels

BER varies across channels: groups of two channels have different BERs

SAFARI

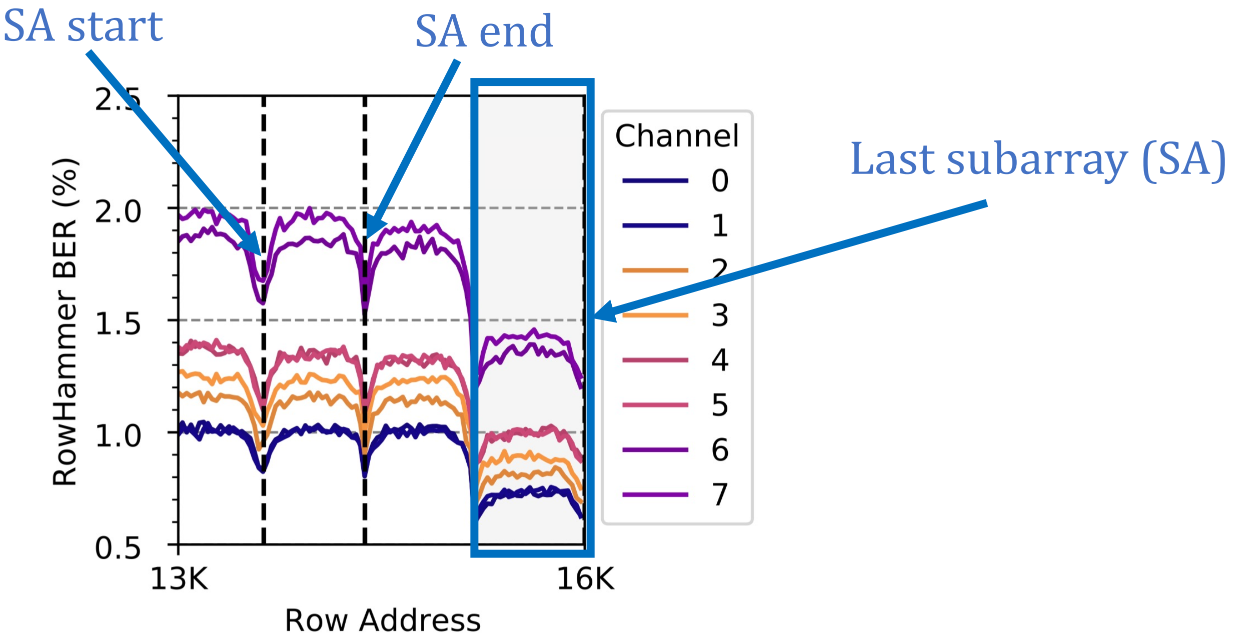
Spatial Distribution of BER (I)



The data pattern affects the BER distribution

Up to ~262 bitflips in a row of 8K bits
with 512K aggressor row activations

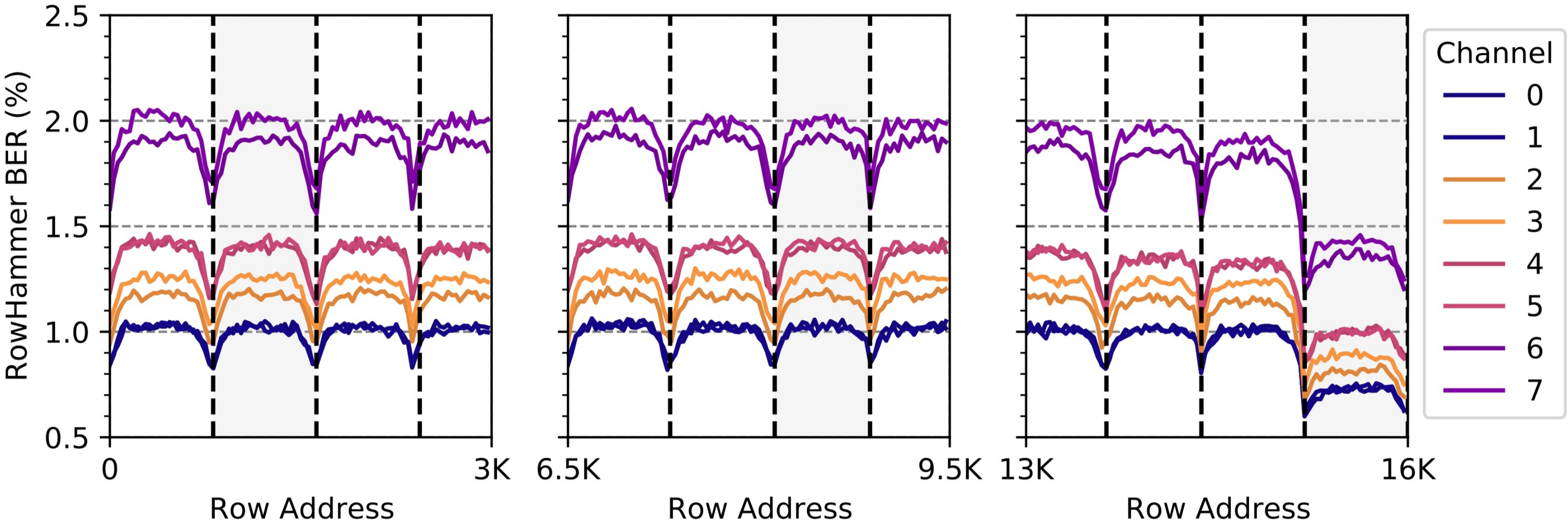
Spatial Distribution of BER (II)



BER is substantially smaller in the last subarray (i.e., last 832 rows)

BER periodically increases and decreases across rows:
BER is higher in the middle of a subarray

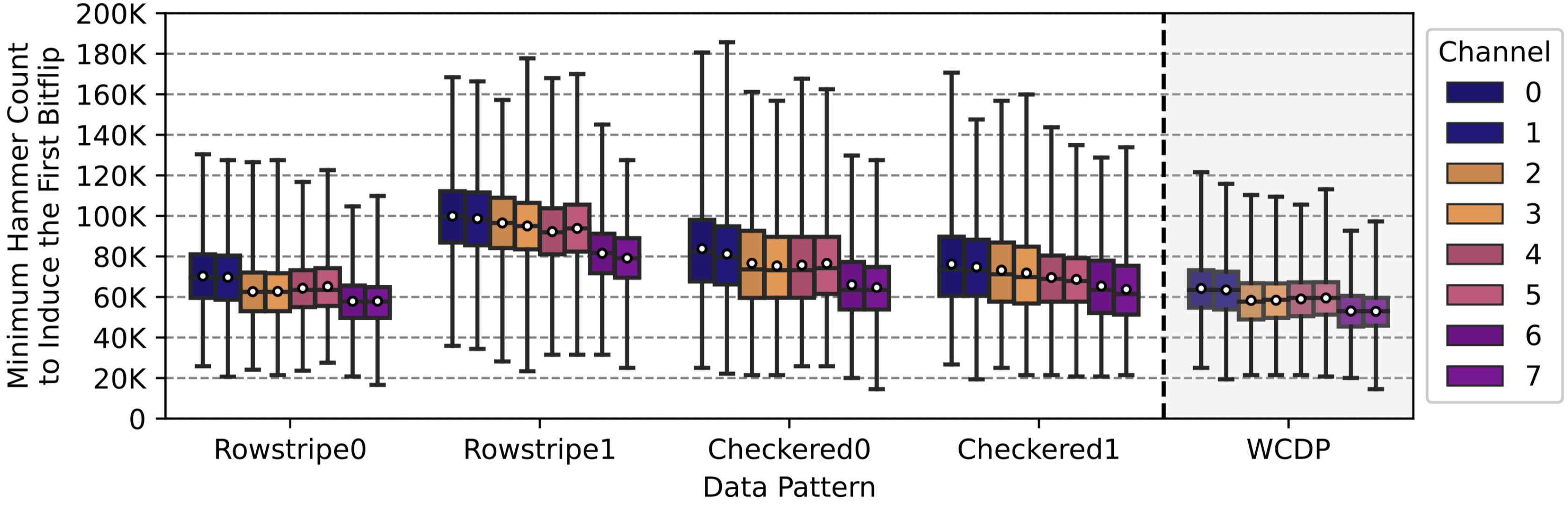
Spatial Distribution of BER (II)



BER is substantially smaller in the last subarray (i.e., last 832 rows)

BER periodically increases and decreases across rows:
BER is higher in the middle of a subarray

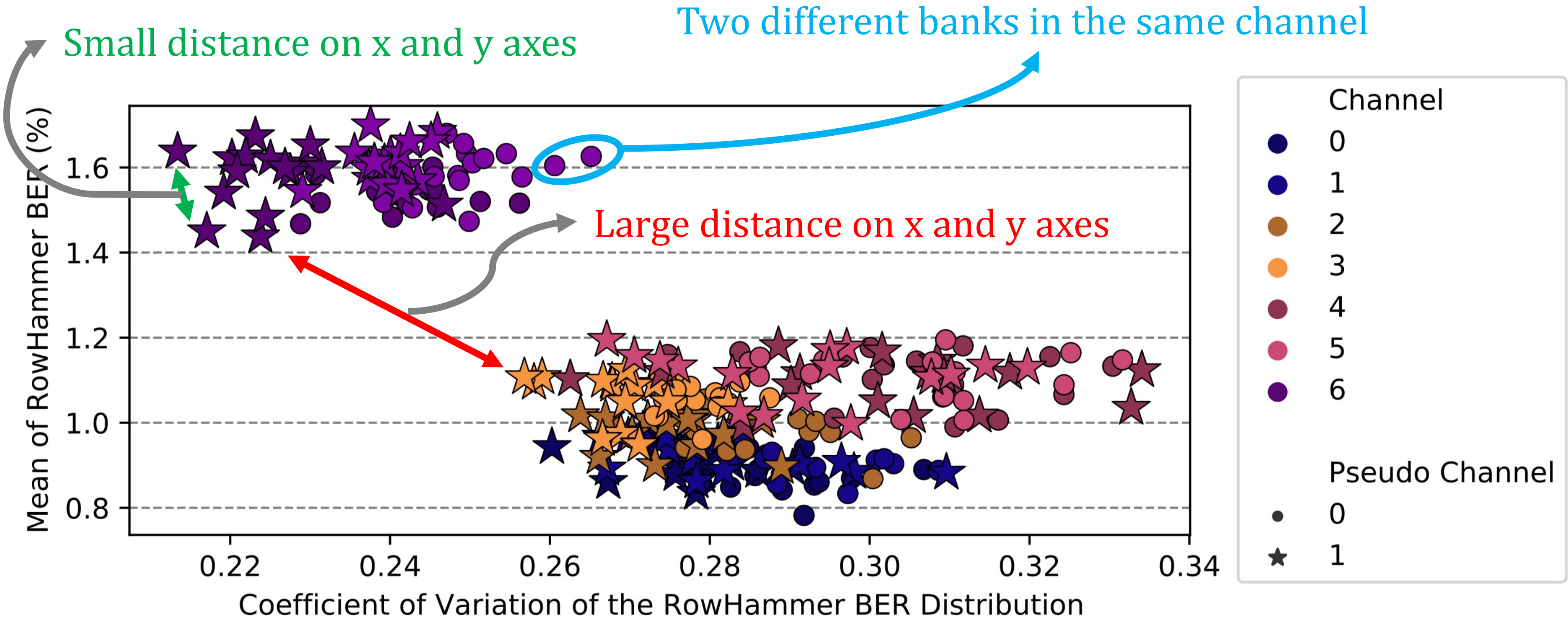
Spatial Distribution of HC_{first}



HC_{first} is as low as 14531 across all tested rows/channels:
Only ~1.3 ms to induce a RowHammer bitflip

HC_{first} distribution heavily depends on the data pattern

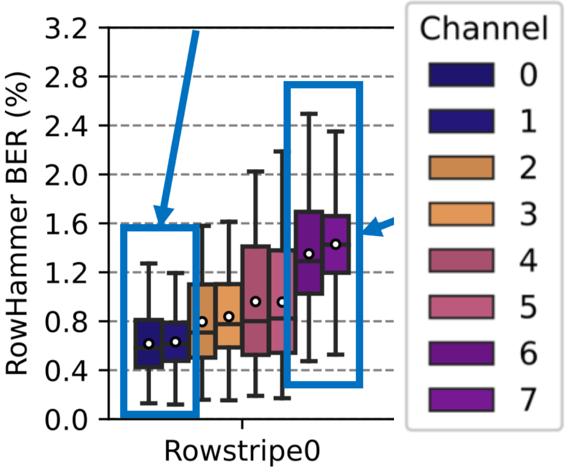
Variation in Bit Error Rate



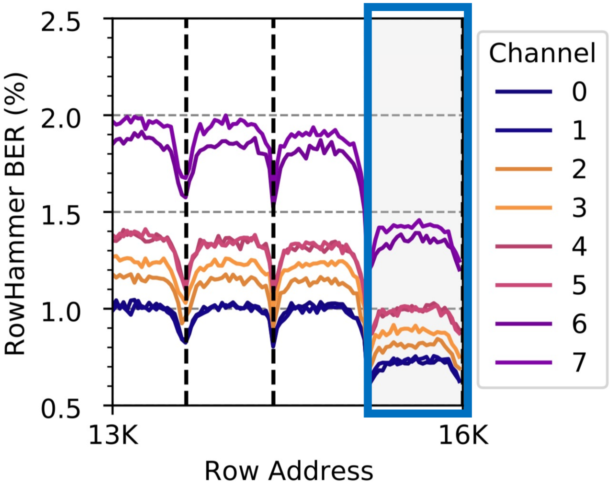
Banks in the same channel have similar variation in BER

Hypotheses from Characterization

1. Similar BER & HC$_{first}$ within groups of two channels suggests these channels share DRAM dies



2. RowHammer BER changes with the row's proximity to sense amplifiers and bank I/O



Implications on Attacks and Mitigations

Key Observation: RowHammer BER and HC_{first} vary across channels

Two implications for RowHammer attacks and mitigations

A RowHammer attack can use the most-RH-vulnerable HBM2 channel to prepare for and perform the attack faster

A RowHammer mitigation can allocate fewer resources for RowHammer-resilient channels and more efficiently prevent RowHammer bitflips

Outline

1. HBM DRAM Organization & Operation

2. DRAM Cell Leakage & RowHammer

3. HBM DRAM Testing Methodology

4. RowHammer Spatial Variation Analysis

5. On-die RowHammer Mitigation Analysis

6. Conclusion

Key Takeaways from on-die Mitigation Analysis

Takeaway 1

A modern HBM2 chip implements an undisclosed on-DRAM-die RowHammer mitigation

Takeaway 2

This mitigation resembles the one in DDR4 chips from one major manufacturer as shown in prior work

Hassan et al., "Uncovering In-DRAM RowHammer Protection Mechanisms: A New Methodology, Custom RowHammer Patterns, and Implications," in MICRO, 2021.

On-Die RowHammer Mitigation Analysis (I)

HBM2 standard defines a “Target Row Refresh (TRR)-mode”

- Memory controller and DRAM cooperate to prevent RH bitflips

Real DDR4 chips implement on-die mitigation mechanisms

- Memory-controller-transparent, hidden behind periodic REF

Does a similar hidden mitigation mechanism exist in HBM2?

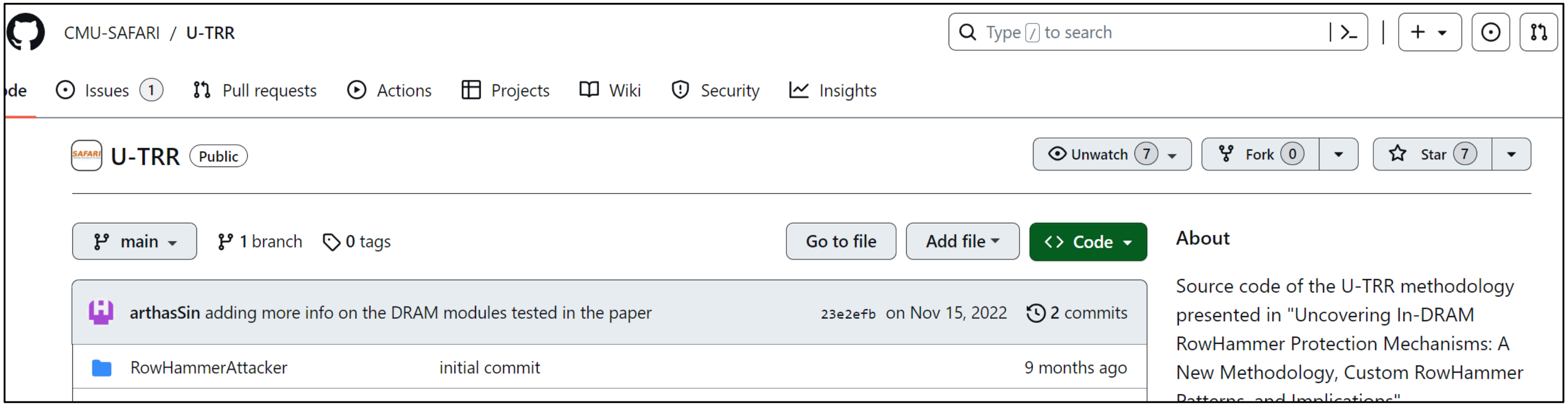
On-Die RowHammer Mitigation Analysis (II)

Hassan et al., "Uncovering In-DRAM RowHammer Protection Mechanisms: A New Methodology, Custom RowHammer Patterns, and Implications," in MICRO, 2021.

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

Hasan Hassan[†] Yahya Can Tuğrul[†‡] Jeremie S. Kim[†] Victor van der Veen[σ]
Kaveh Razavi[†] Onur Mutlu[†]
[†]*ETH Zürich* [‡]*TOBB University of Economics & Technology* [σ]*Qualcomm Technologies Inc.*

Key idea: Use data retention failures as a side channel to detect when a row is refreshed by on-die mitigation



U-TRR Public

arthasSin adding more info on the DRAM modules tested in the paper 23e2efb on Nov 15, 2022 2 commits

RowHammerAttacker initial commit 9 months ago

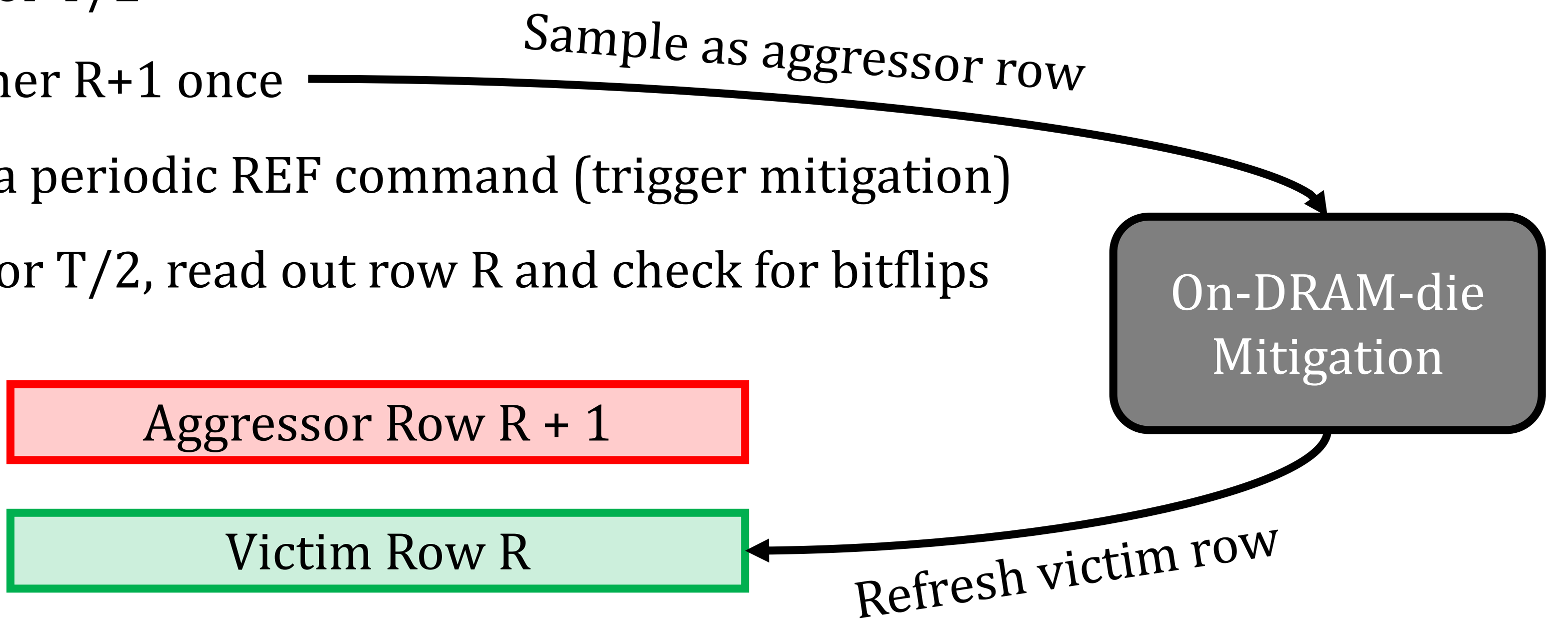
About

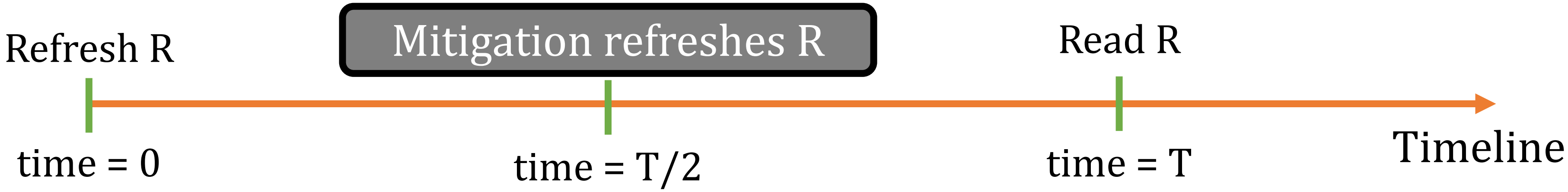
Source code of the U-TRR methodology presented in "Uncovering In-DRAM RowHammer Protection Mechanisms: A New Methodology, Custom RowHammer Patterns, and Implications"

SAFARI **[Hassan+, MICRO'21, source code available at https://github.com/CMU-SAFARI/U-TRR]**

Experimental Methodology

1. Identify a row (R) with **T** retention time
2. Wait for T/2
3. Hammer R+1 once
4. Issue a periodic REF command (trigger mitigation)
5. Wait for T/2, read out row R and check for bitflips





[Hassan+, MICRO'21]

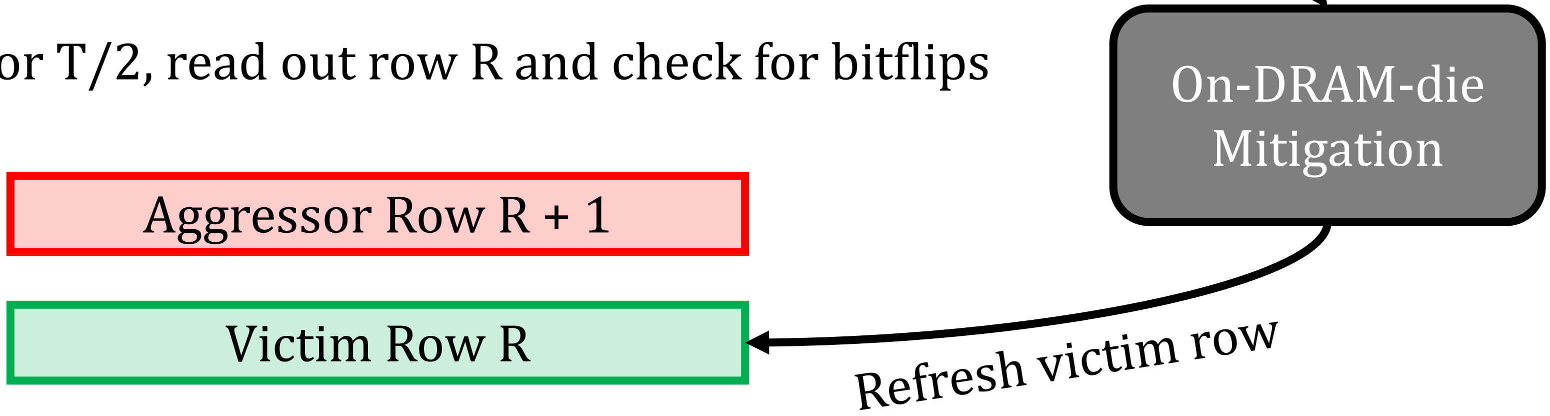
Experimental Methodology

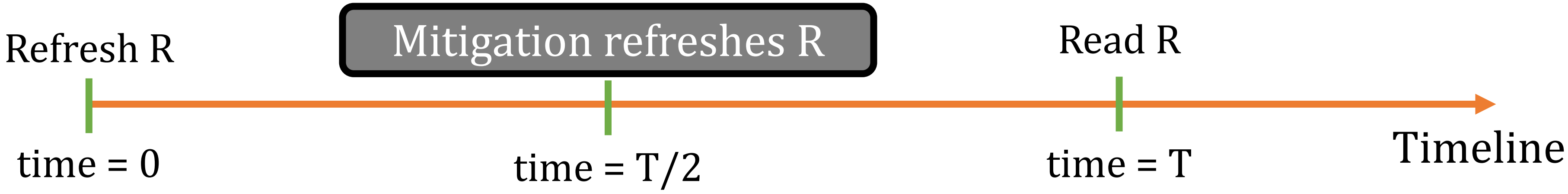
1. Identify a row (R) with **T** retention time

Row R experiences no bitflips
only if on-DRAM-die mitigation exists

4. Issue a periodic REF command (trigger mitigation)

5. Wait for T/2, read out row R and check for bitflips





[Hassan+, MICRO'21]

Experimental Methodology

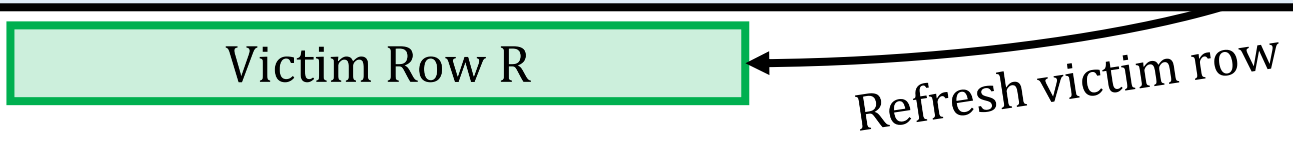
1. Identify a row (R) with **T** retention time

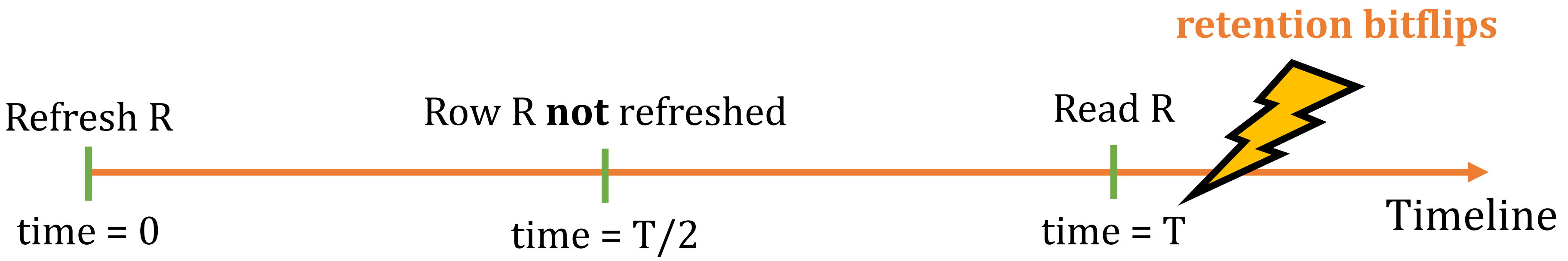
Row R experiences no bitflips
only if on-DRAM-die mitigation exists

4. Issue a periodic REF command (trigger mitigation)

5. Wait for T/2, read out row R and check for bitflips

Row R experiences retention bitflips
if not refreshed at T/2





[Hassan+, MICRO'21]

HBM2 DRAM Chips Implement Undisclosed TRR

The HBM2 chip implements an undisclosed on-die RowHammer mitigation mechanism

This mechanism performs a victim row refresh operation every 17 periodic refresh (REF) operations

This mitigation resembles the one in DDR4 chips from one major manufacturer

Outline

1. HBM DRAM Organization & Operation

2. DRAM Cell Leakage & RowHammer

3. HBM DRAM Testing Methodology

4. RowHammer Spatial Variation Analysis

5. On-die RowHammer Mitigation Analysis

6. Conclusion

Conclusion

We provide the first detailed experimental characterization of RowHammer in a modern HBM2 DRAM chip

Different channels in 3D-stacked HBM chips exhibit different RowHammer vulnerability

DRAM rows near the end of a DRAM bank are more RowHammer resilient

Two implications for RowHammer attacks and mitigations:

1. Faster and more effective attacks
2. More efficient mitigations

A modern HBM chip implements undisclosed on-DRAM-die RowHammer mitigation (e.g., similar to DDR4 chips)

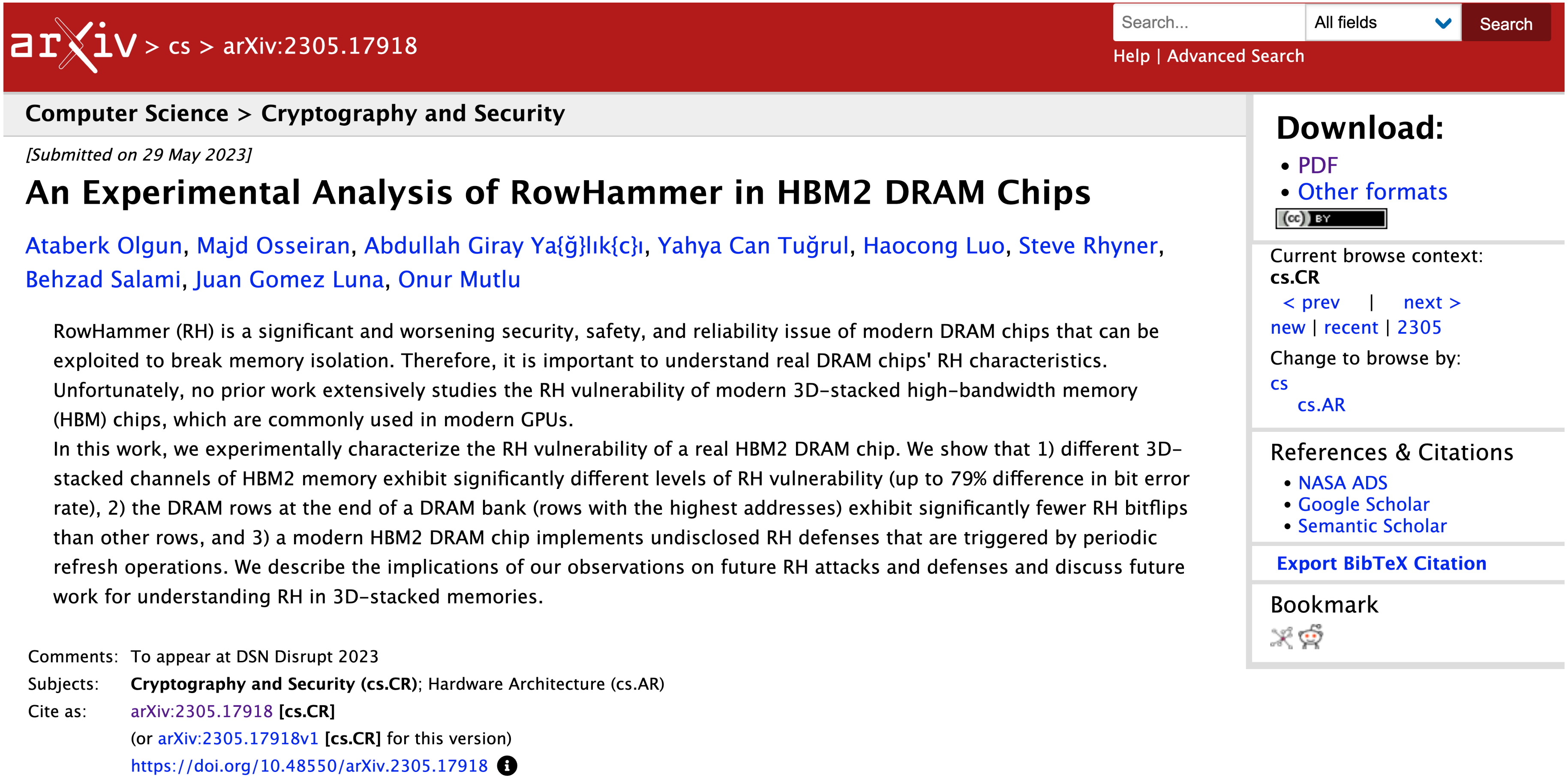
Future Directions: To present more insights into how RowHammer behaves in HBM

1. Test more HBM DRAM chips, data patterns, at different temperature and voltage levels
2. Investigate read-disturb-based interference across different 3D-stacked HBM DRAM channels
3. Study the effects of the new read-disturb phenomenon, RowPress [Luo+, ISCA'23]

Luo et al., "RowPress: Amplifying Read Disturbance in Modern DRAM Chips," in ISCA, 2023.

Available on ArXiv

https://arxiv.org/abs/2305.17918



Computer Science > Cryptography and Security

[Submitted on 29 May 2023]

An Experimental Analysis of RowHammer in HBM2 DRAM Chips

Ataberk Olgun, Majd Osseiran, Abdullah Giray Ya{ğ}lık{c}ı, Yahya Can Tuğrul, Haocong Luo, Steve Rhyner, Behzad Salami, Juan Gomez Luna, Onur Mutlu

RowHammer (RH) is a significant and worsening security, safety, and reliability issue of modern DRAM chips that can be exploited to break memory isolation. Therefore, it is important to understand real DRAM chips' RH characteristics. Unfortunately, no prior work extensively studies the RH vulnerability of modern 3D-stacked high-bandwidth memory (HBM) chips, which are commonly used in modern GPUs.

In this work, we experimentally characterize the RH vulnerability of a real HBM2 DRAM chip. We show that 1) different 3D-stacked channels of HBM2 memory exhibit significantly different levels of RH vulnerability (up to 79% difference in bit error rate), 2) the DRAM rows at the end of a DRAM bank (rows with the highest addresses) exhibit significantly fewer RH bitflips than other rows, and 3) a modern HBM2 DRAM chip implements undisclosed RH defenses that are triggered by periodic refresh operations. We describe the implications of our observations on future RH attacks and defenses and discuss future work for understanding RH in 3D-stacked memories.

Comments:	To appear at DSN Disrupt 2023
Subjects:	**Cryptography and Security (cs.CR)**; Hardware Architecture (cs.AR)
Cite as:	arXiv:2305.17918 **[cs.CR]**
	(or arXiv:2305.17918v1 **[cs.CR]** for this version)
	https://doi.org/10.48550/arXiv.2305.17918 ℹ

Download:

- PDF
- Other formats

(cc) BY

Current browse context:
cs.CR
< prev | next >
new | recent | 2305
Change to browse by:
cs
cs.AR

References & Citations

- NASA ADS
- Google Scholar
- Semantic Scholar

Export BibTeX Citation

Bookmark

An Experimental Analysis of RowHammer in HBM2 DRAM Chips

Link/QR code to full paper
https://arxiv.org/pdf/2305.17918

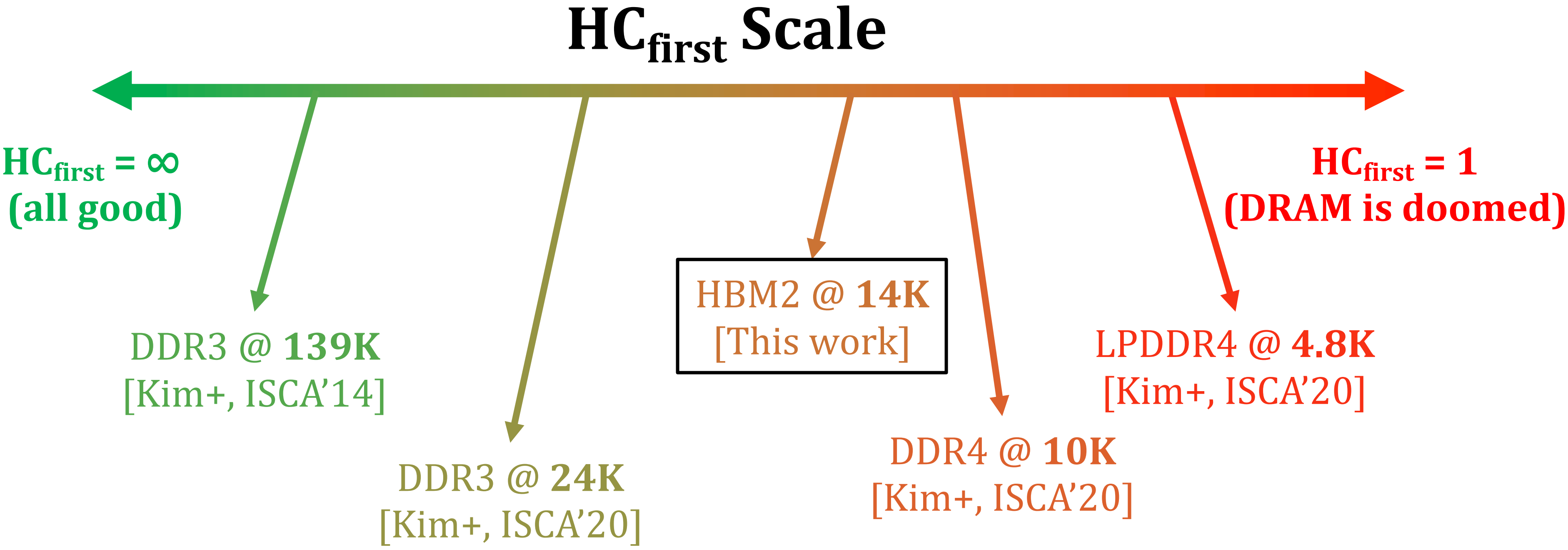
Ataberk Olgun Majd Osseiran

A. Giray Yağlıkçı Yahya Can Tuğrul Haocong Luo Steve Rhyner

Behzad Salami Juan Gomez Luna Onur Mutlu

ETH zürich SAFARI AMERICAN UNIVERSITY OF BEIRUT

Publicly-available HC$_{first}$ Values



HC$_{first}$ Scale

HC$_{first}$ = ∞
(all good)

HC$_{first}$ = 1
(DRAM is doomed)

DDR3 @ **139K**
[Kim+, ISCA'14]

DDR3 @ **24K**
[Kim+, ISCA'20]

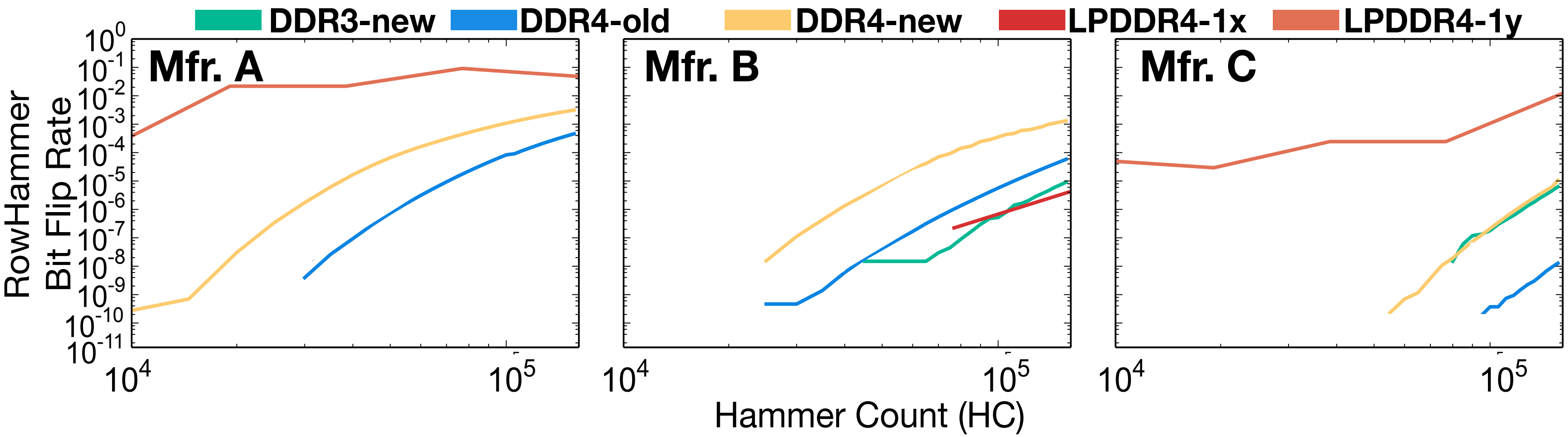
HBM2 @ **14K**
[This work]

DDR4 @ **10K**
[Kim+, ISCA'20]

LPDDR4 @ **4.8K**
[Kim+, ISCA'20]

*Not shown: Significant variance in HC$_{first}$ across vendors and die variations

3. Hammer Count (HC) Effects

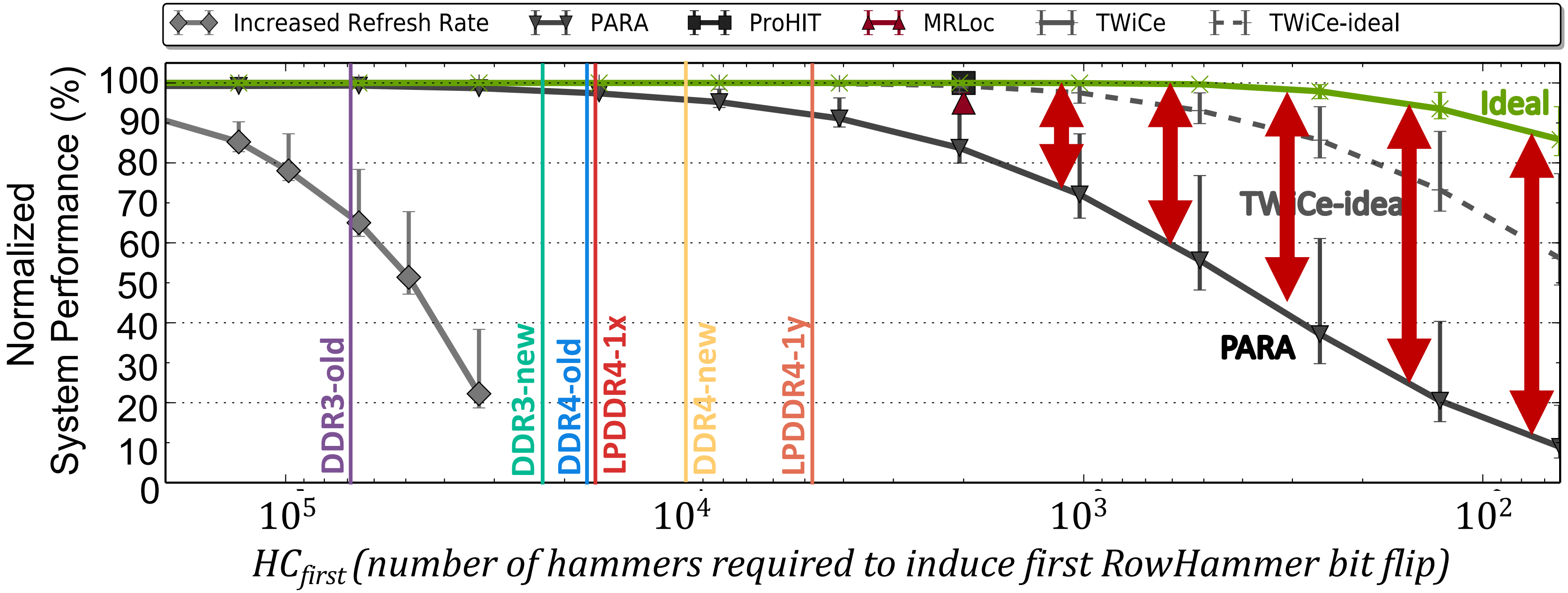


RowHammer bit flip rates **increase**
when going **from old to new** DDR4 technology node generations

RowHammer bit flip rates (i.e., RowHammer vulnerability) increase with technology node generation

[Kim+, ISCA'20]

Mitigation Mechanism Evaluation



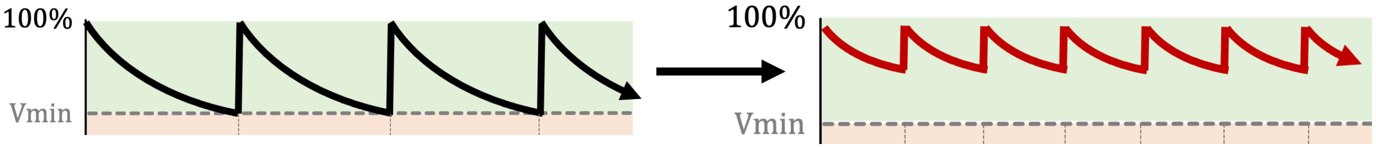
Ideal mechanism is significantly better than any existing mechanism for $HC_{first} < 1024$

Significant opportunity for developing a RowHammer solution with low performance overhead that supports low HC_{first}

[Kim+, ISCA'20]

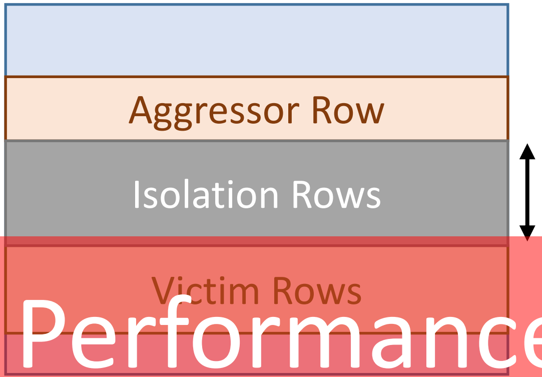
RowHammer Solution Approaches

- More robust DRAM chips **and/or** error-correcting codes
- Increased refresh rate



Fewer activations possible
in a refresh interval

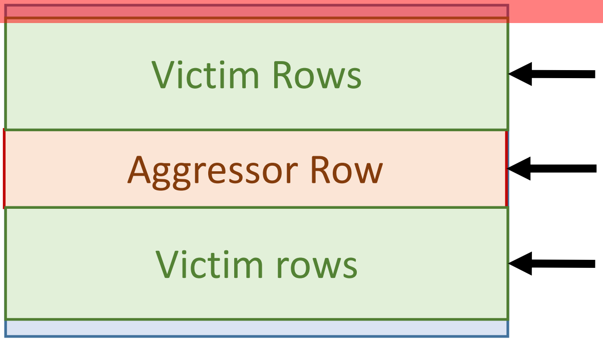
- Physical isolation



Large-enough distance

Cost, Power, Performance, Complexity

- Reactive refresh

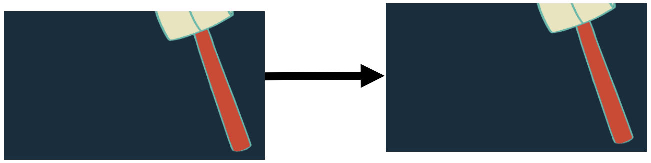


Refresh

Rapidly activated (hammered)

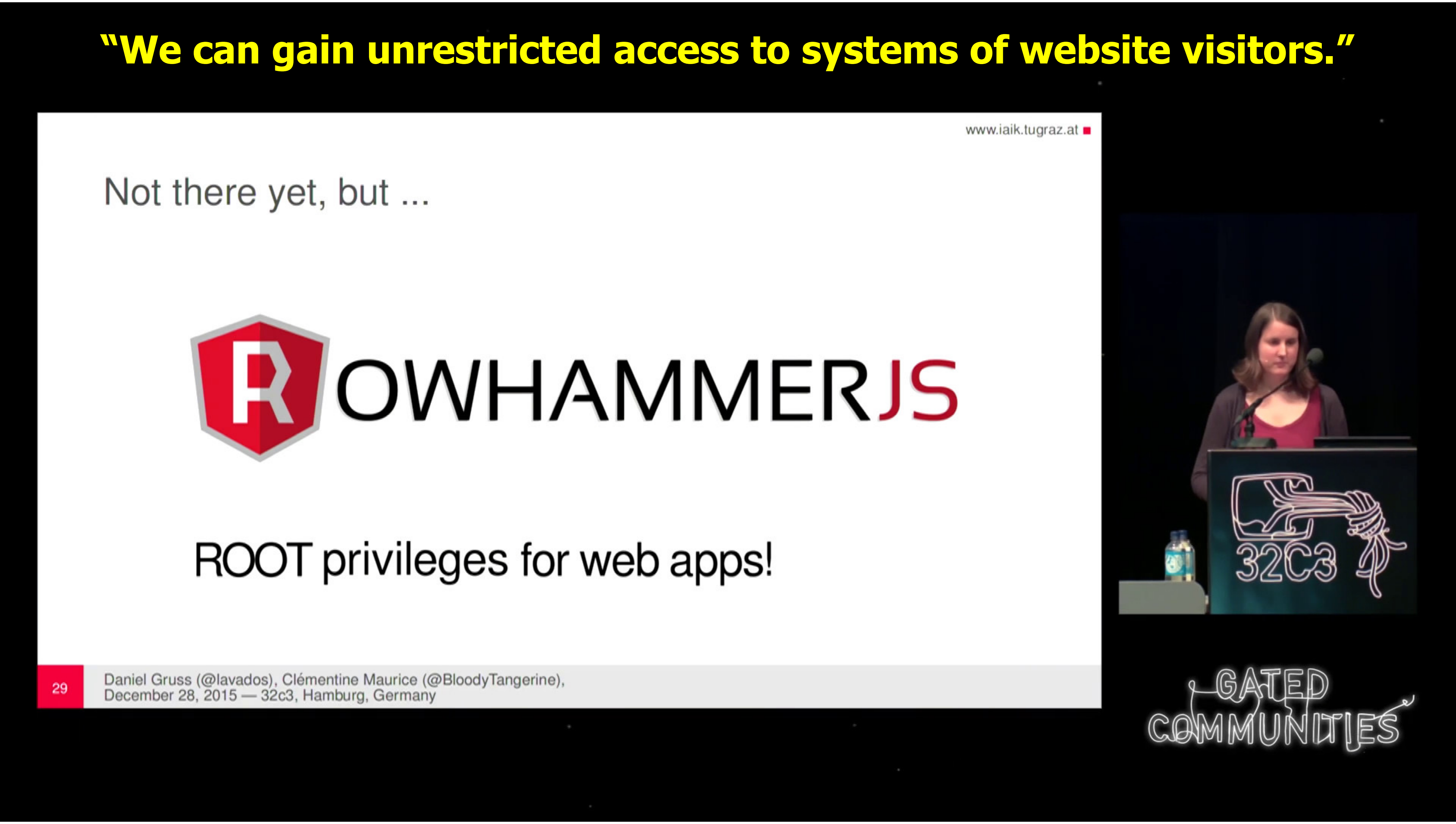
Refresh

- Proactive throttling



Fewer activations allowed for aggressive applications

More Security Implications (I)



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

www.iaik.tugraz.at

Not there yet, but ...

ROWHAMMERJS

ROOT privileges for web apps!

29

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

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

Source: https://lab.dsst.io/32c3-slides/7197.html

More Security Implications (II)

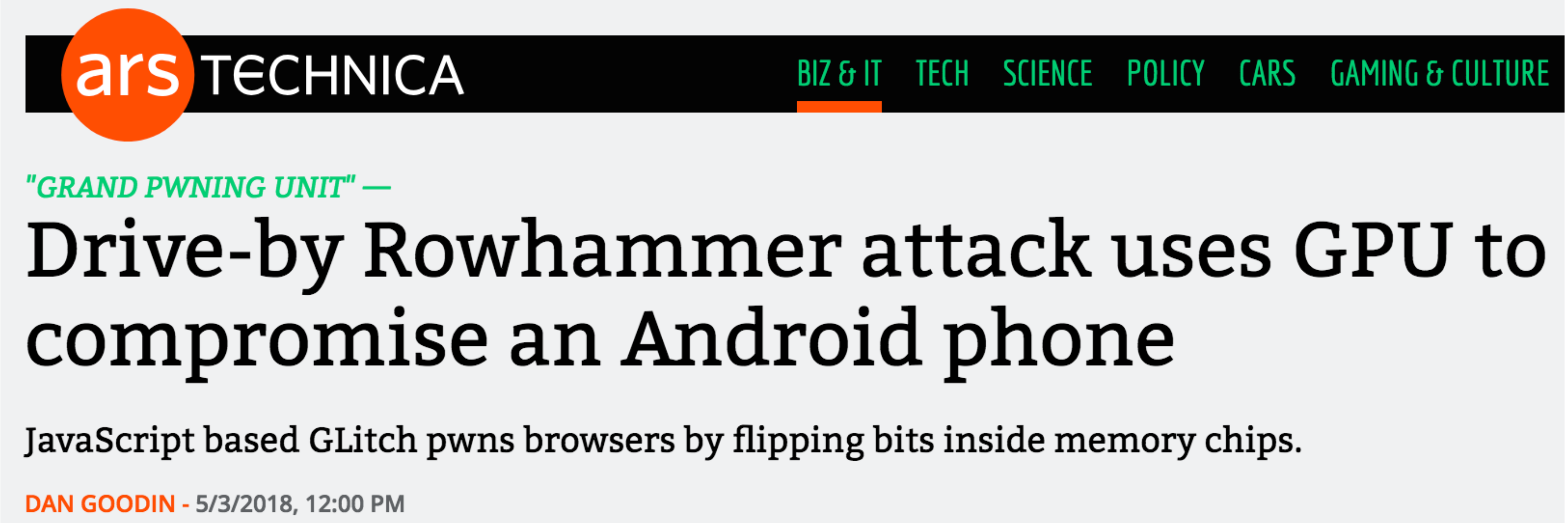
"Can gain control of a smart phone deterministically"

Drammer: Deterministic Rowhammer Attacks on Mobile Platforms, CCS'16

Source: https://fossbytes.com/drammer-rowhammer-attack-android-root-devices/

More Security Implications (III)

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



ars TECHNICA

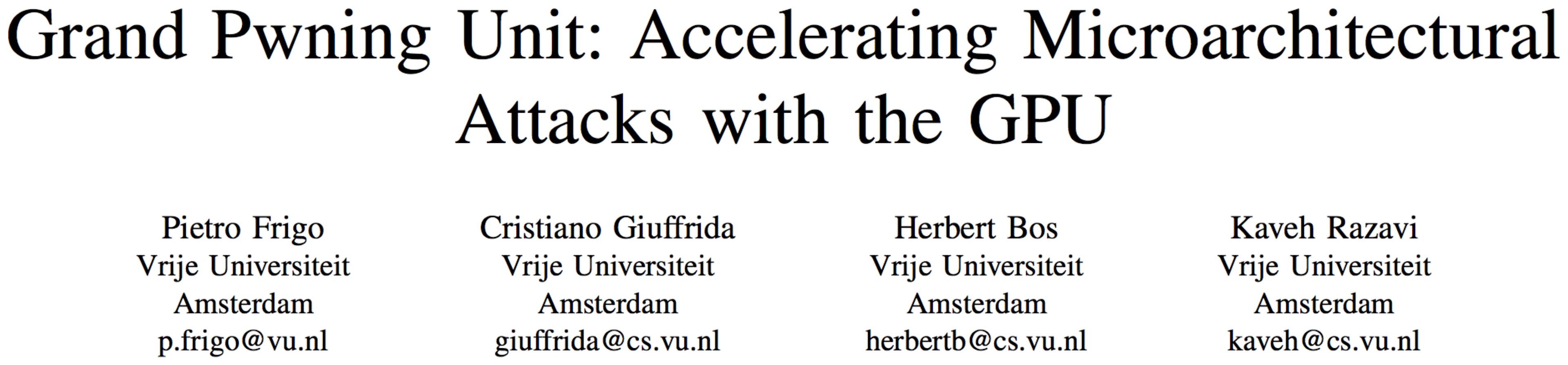
BIZ & IT TECH SCIENCE POLICY CARS GAMING & CULTURE

"GRAND PWNING UNIT" —

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

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

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



Grand Pwning Unit: Accelerating Microarchitectural Attacks with the GPU

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

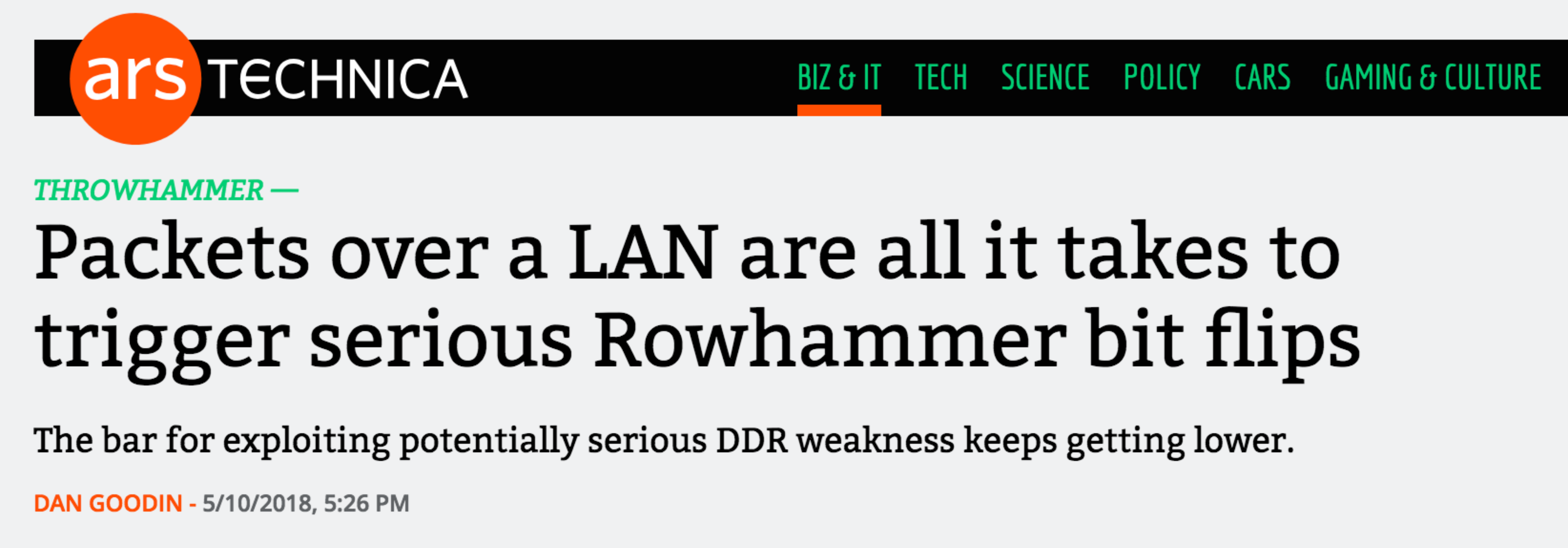
Cristiano Giuffrida
Vrije Universiteit
Amsterdam
giuffrida@cs.vu.nl

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

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

More Security Implications (IV)

- Rowhammer over RDMA (I) USENIX ATC 2018

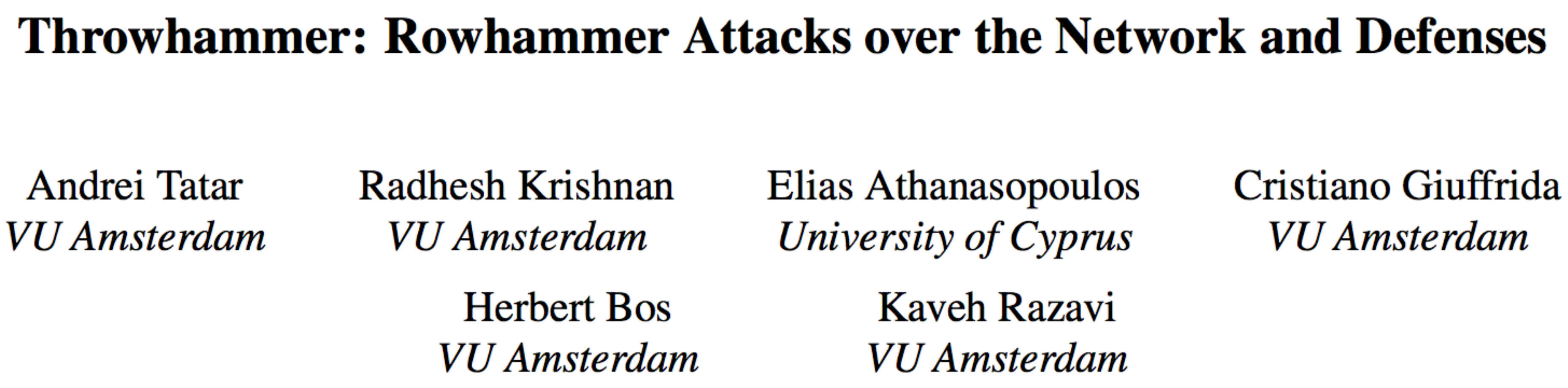


THROWHAMMER —

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

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

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



Throwhammer: Rowhammer Attacks over the Network and Defenses

Andrei Tatar
VU Amsterdam

Radhesh Krishnan
VU Amsterdam

Elias Athanasopoulos
University of Cyprus

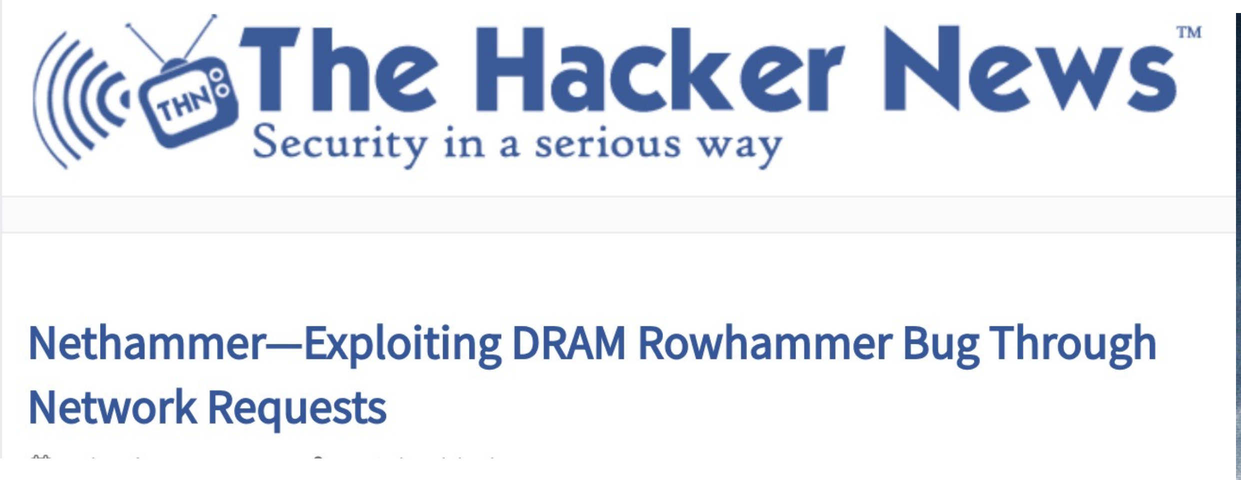
Cristiano Giuffrida
VU Amsterdam

Herbert Bos
VU Amsterdam

Kaveh Razavi
VU Amsterdam

More Security Implications (V)

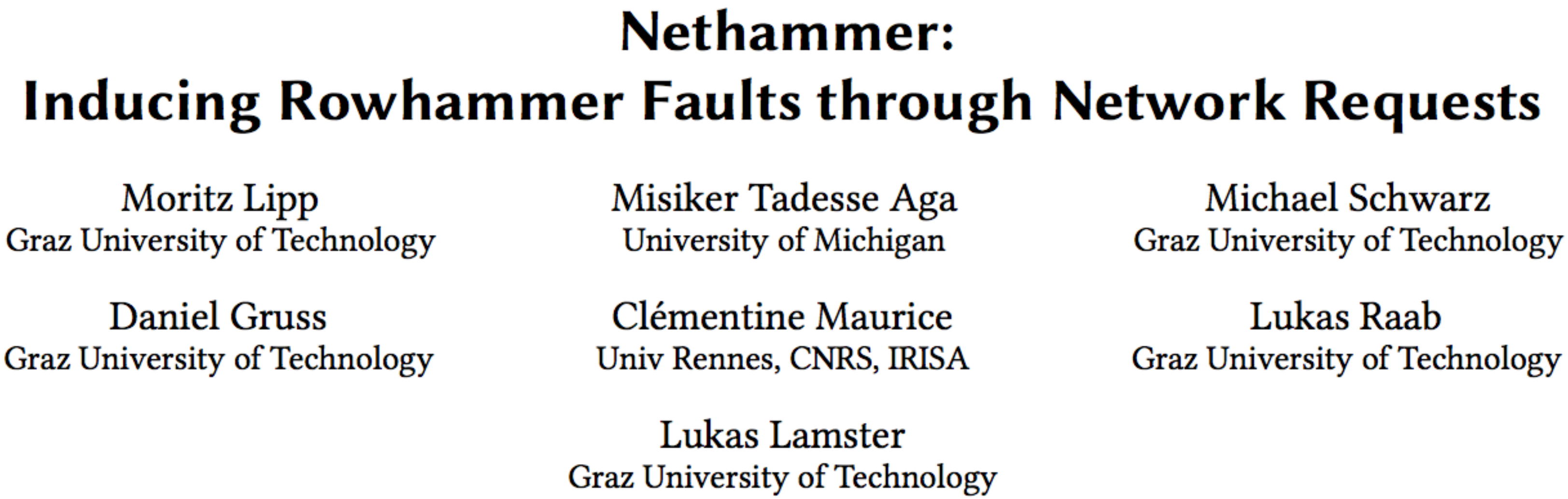
- Rowhammer over RDMA (II)



The Hacker News™

Security in a serious way

Nethammer—Exploiting DRAM Rowhammer Bug Through Network Requests



Nethammer:
Inducing Rowhammer Faults through Network Requests

Moritz Lipp
Graz University of Technology

Misiker Tadesse Aga
University of Michigan

Michael Schwarz
Graz University of Technology

Daniel Gruss
Graz University of Technology

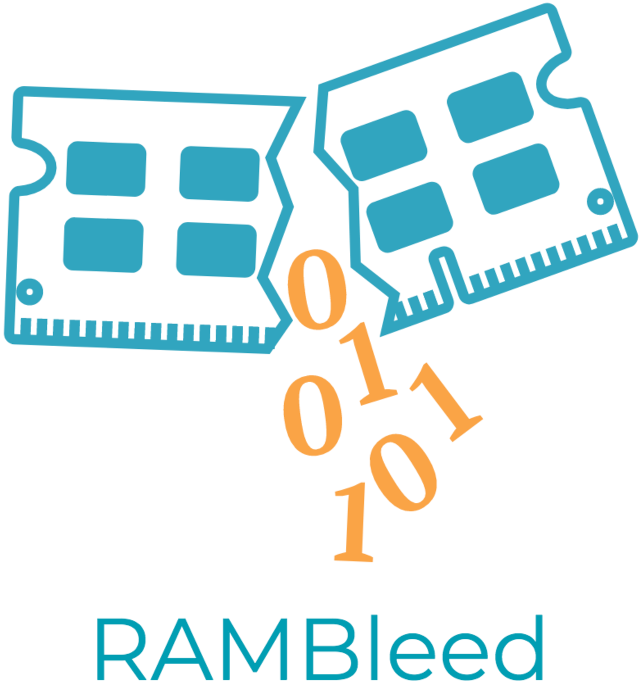
Clémentine Maurice
Univ Rennes, CNRS, IRISA

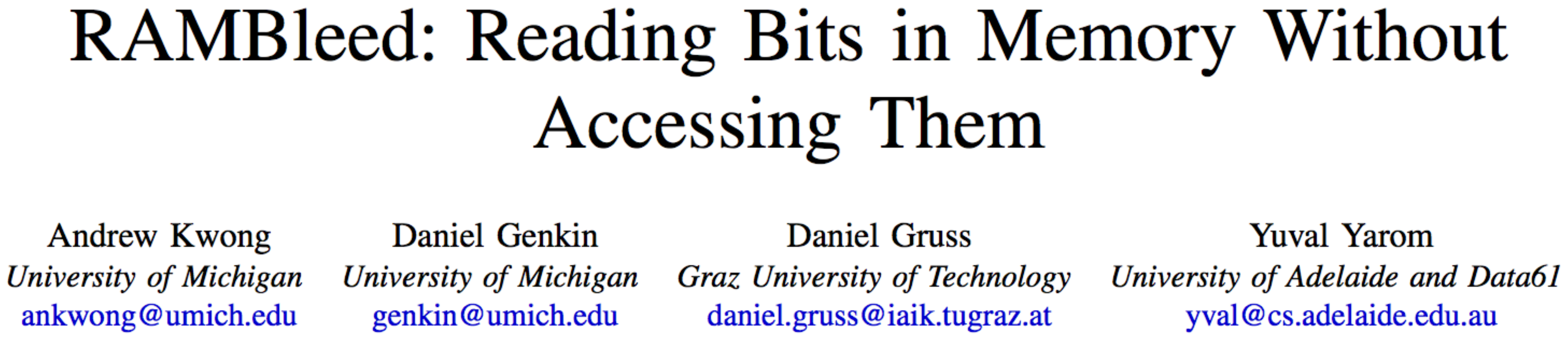
Lukas Raab
Graz University of Technology

Lukas Lamster
Graz University of Technology

More Security Implications (VI)

- IEEE S&P 2020





RAMBleed: Reading Bits in Memory Without Accessing Them

Andrew Kwong
University of Michigan
ankwong@umich.edu

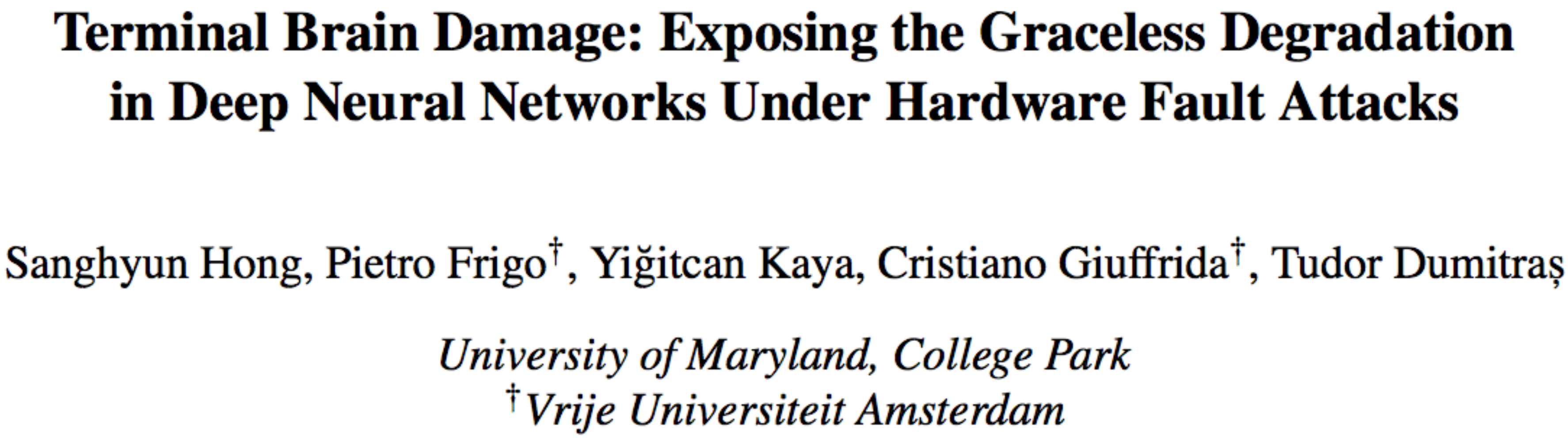
Daniel Genkin
University of Michigan
genkin@umich.edu

Daniel Gruss
Graz University of Technology
daniel.gruss@iaik.tugraz.at

Yuval Yarom
University of Adelaide and Data61
yval@cs.adelaide.edu.au

More Security Implications (VII)

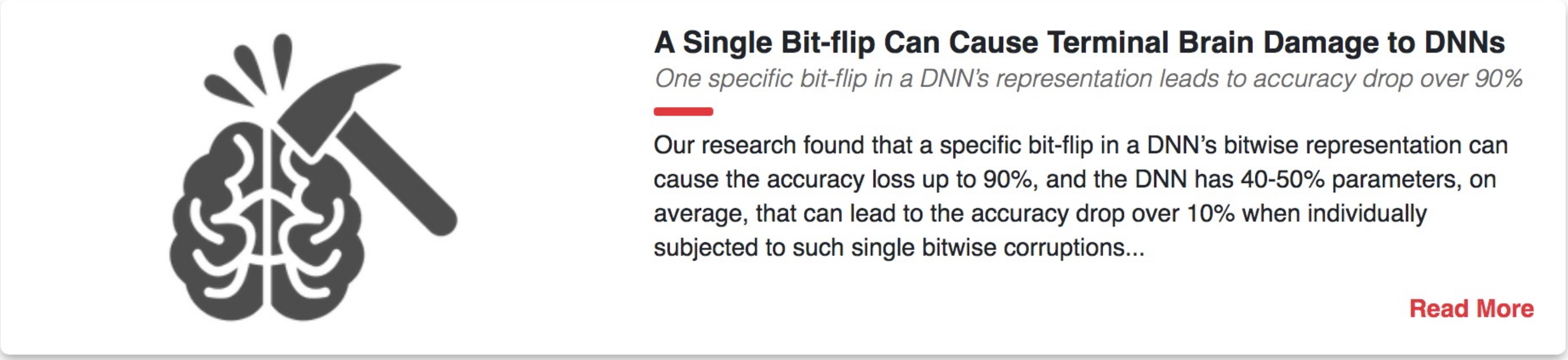
- USENIX Security 2019



Terminal Brain Damage: Exposing the Graceless Degradation in Deep Neural Networks Under Hardware Fault Attacks

Sanghyun Hong, Pietro Frigo[†], Yiğitcan Kaya, Cristiano Giuffrida[†], Tudor Dumitraș

University of Maryland, College Park
[†]Vrije Universiteit Amsterdam



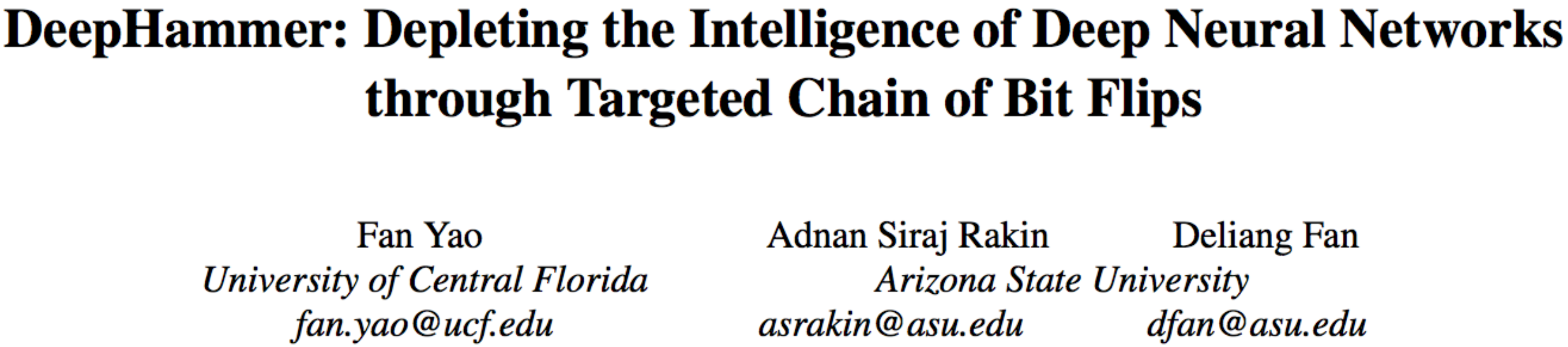
A Single Bit-flip Can Cause Terminal Brain Damage to DNNs

One specific bit-flip in a DNN's representation leads to accuracy drop over 90%

Our research found that a specific bit-flip in a DNN's bitwise representation can cause the accuracy loss up to 90%, and the DNN has 40-50% parameters, on average, that can lead to the accuracy drop over 10% when individually subjected to such single bitwise corruptions...

More Security Implications (VIII)

- USENIX Security 2020



DeepHammer: Depleting the Intelligence of Deep Neural Networks through Targeted Chain of Bit Flips

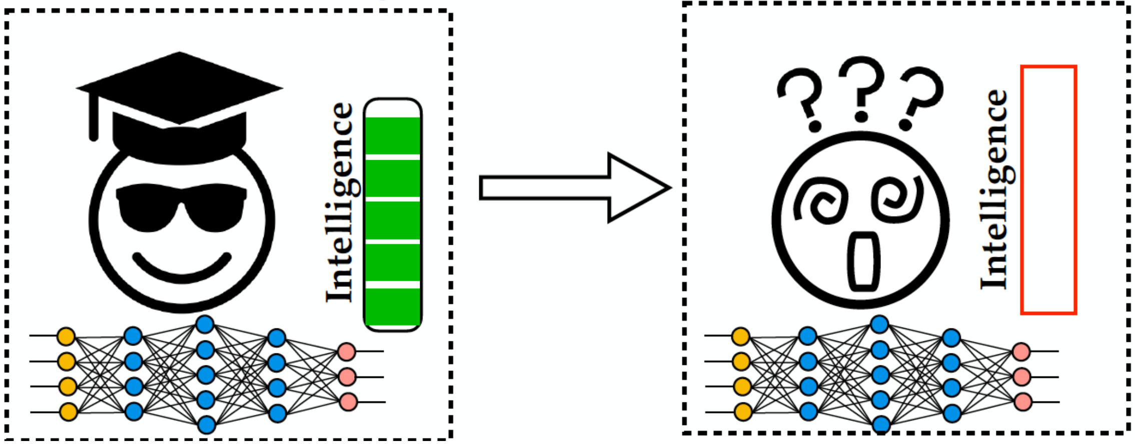
Fan Yao
University of Central Florida
fan.yao@ucf.edu

Adnan Siraj Rakin
Deliang Fan
Arizona State University
asrakin@asu.edu
dfan@asu.edu

Degrade the **inference accuracy** to the level of **Random Guess**

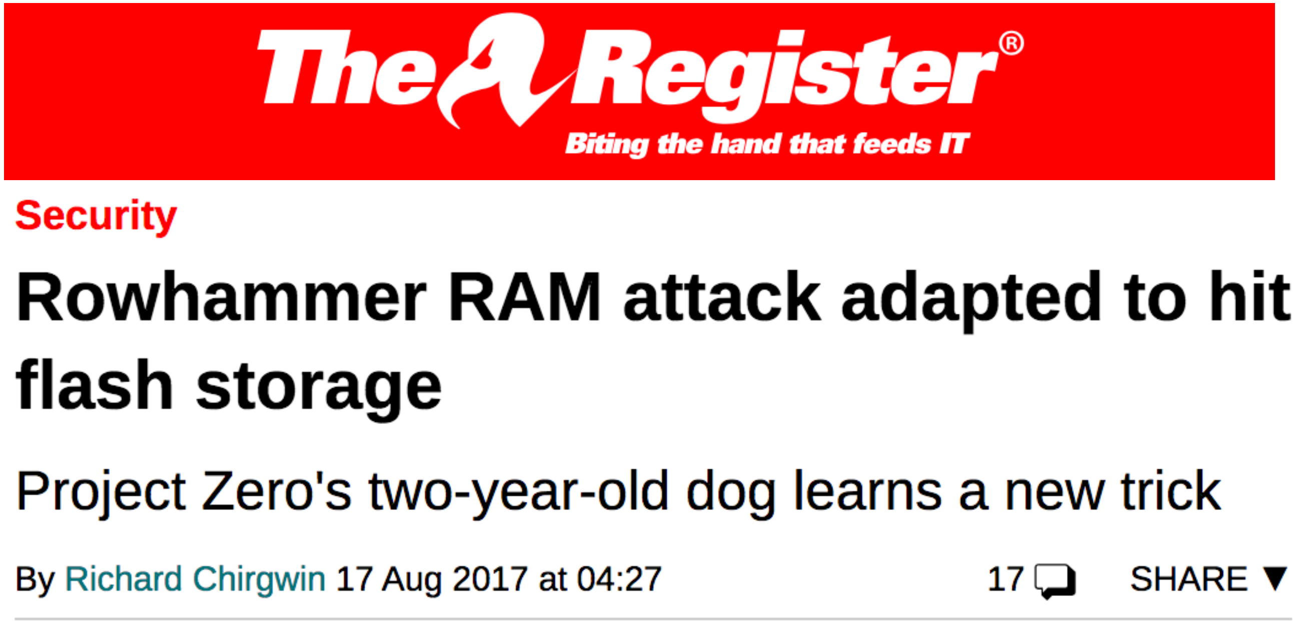
Example: ResNet-20 for CIFAR-10, **10** output classes

Before attack, **Accuracy: 90.2%** After attack, **Accuracy: ~10% (1/10)**



More Security Implications (IX)

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



The Register®

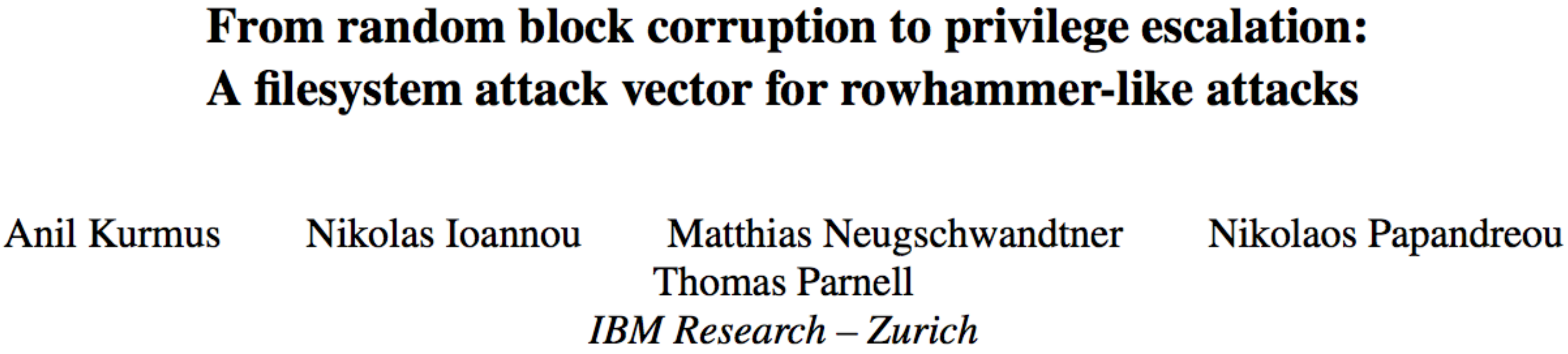
Biting the hand that feeds IT

Security

Rowhammer RAM attack adapted to hit flash storage

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

By Richard Chirgwin 17 Aug 2017 at 04:27 17 SHARE ▼

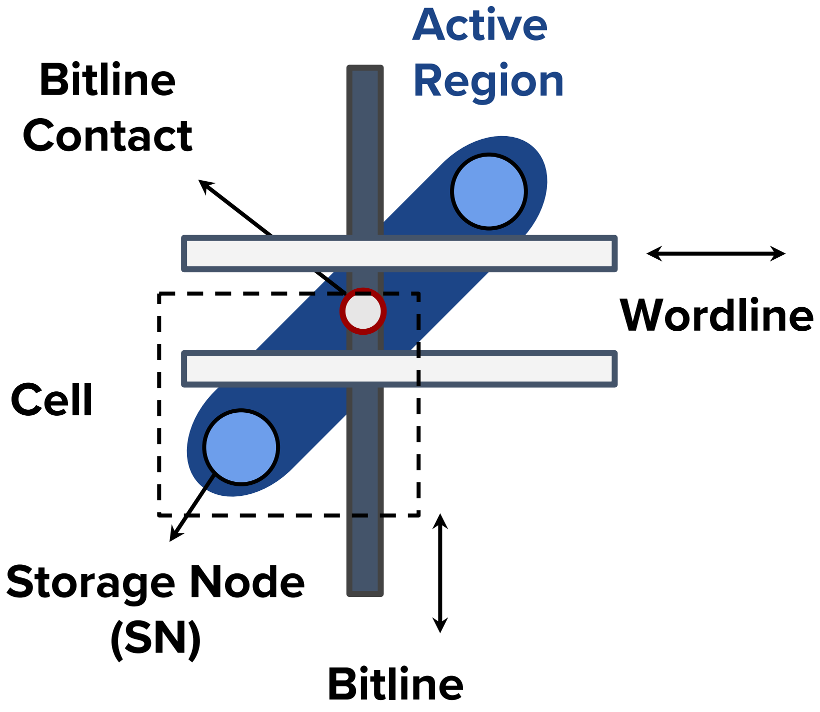


From random block corruption to privilege escalation: A filesystem attack vector for rowhammer-like attacks

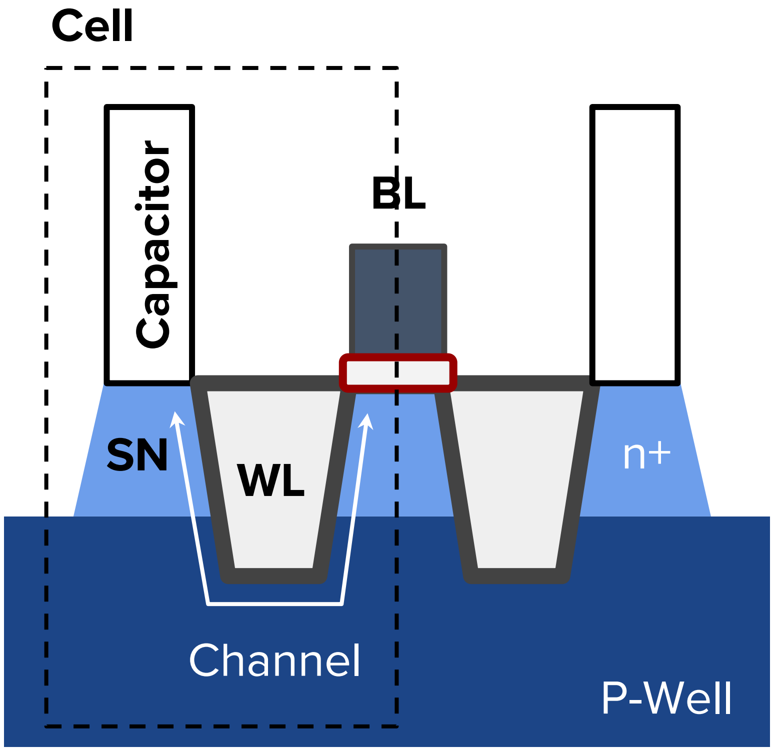
Anil Kurmus Nikolas Ioannou Matthias Neugschwandtner Nikolaos Papandreou
Thomas Parnell
IBM Research – Zurich

DRAM Array Layout

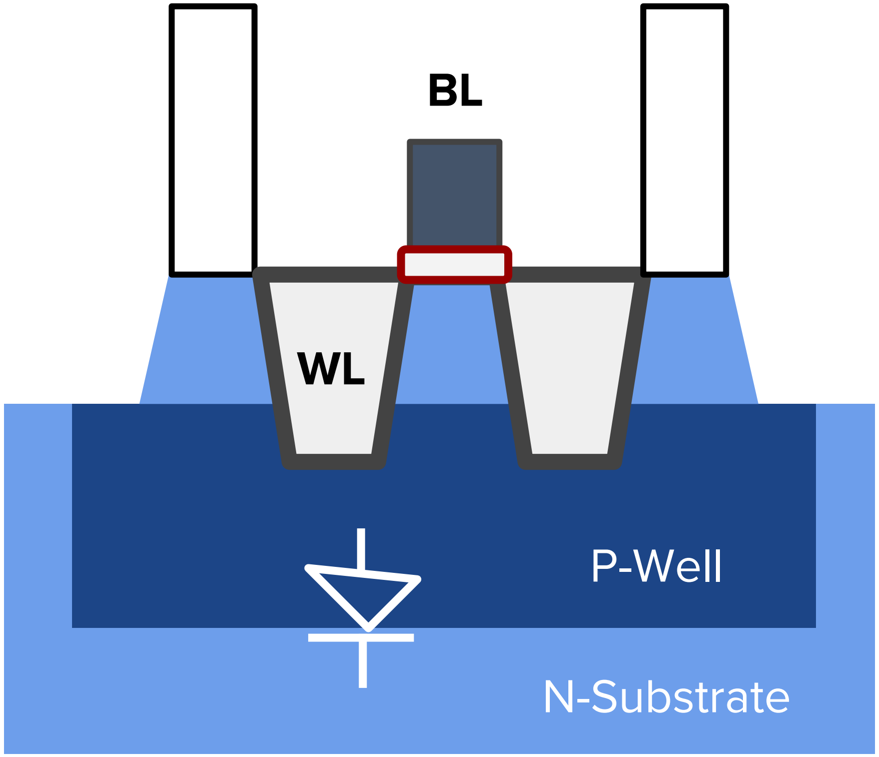
Top View



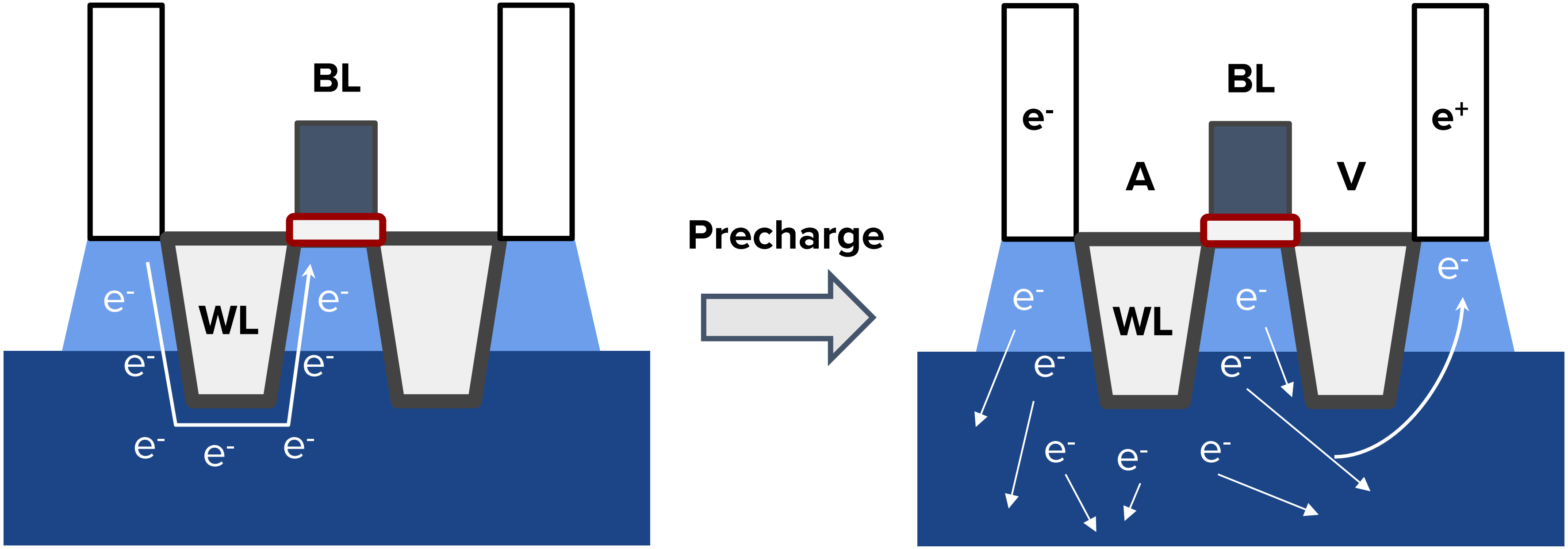
Cross Section



Mechanism 0: Reflecting Electric Field



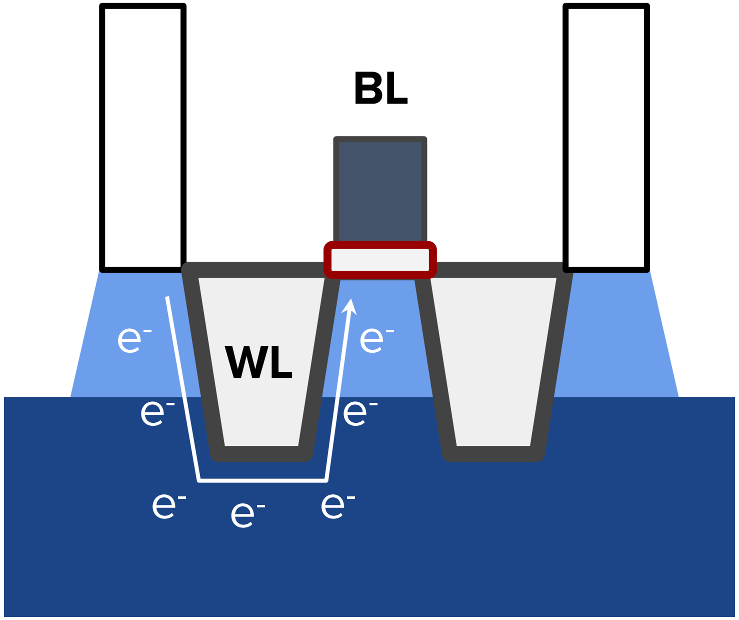
Mechanism 1: Electron Injection



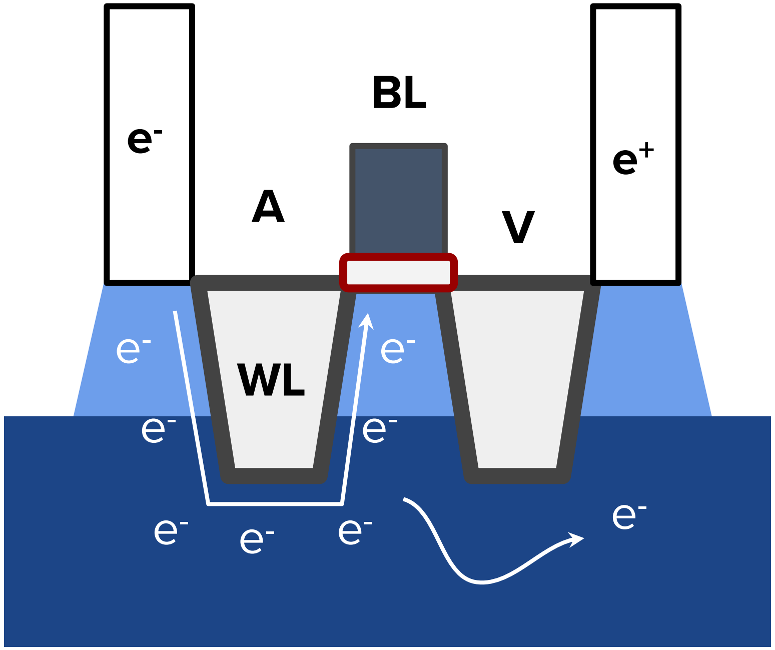
Aggressor ACT

Recombination

Mechanism 2: Electron Drift



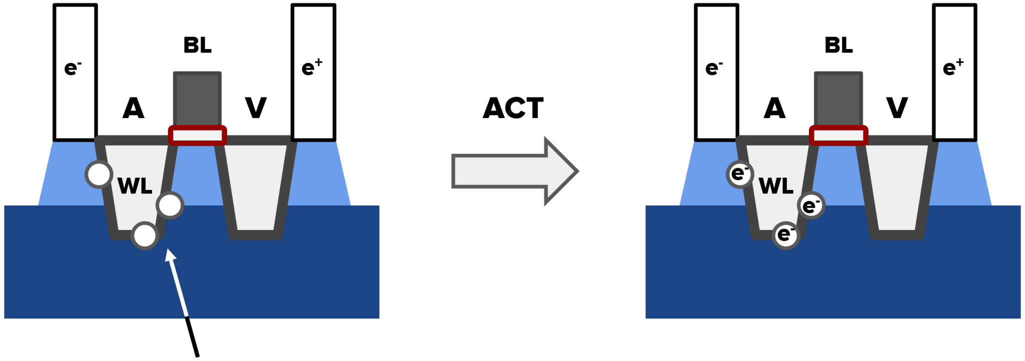
Aggressor ACT



Electron Drift

More

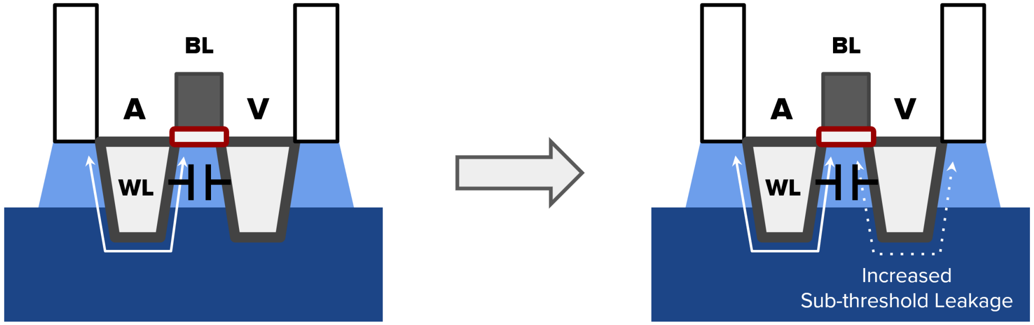
- Charge traps



Interface Charge Trap

Trap Charged

- Wordline Crosstalk

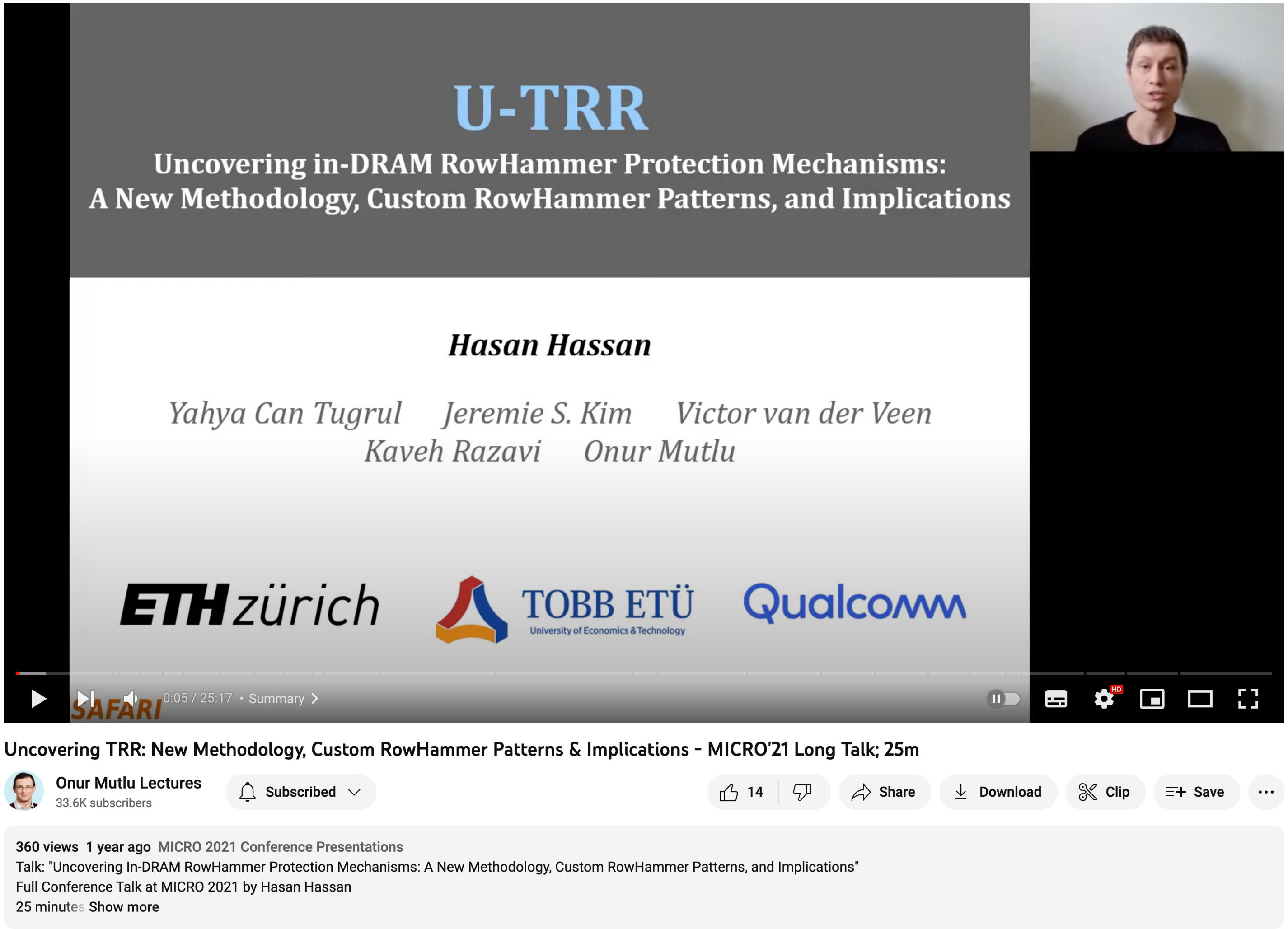


Aggressor ACT

Victim Leakage

More on U-TRR

https://youtu.be/YkBR9yeLHRs



U-TRR

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

Hasan Hassan

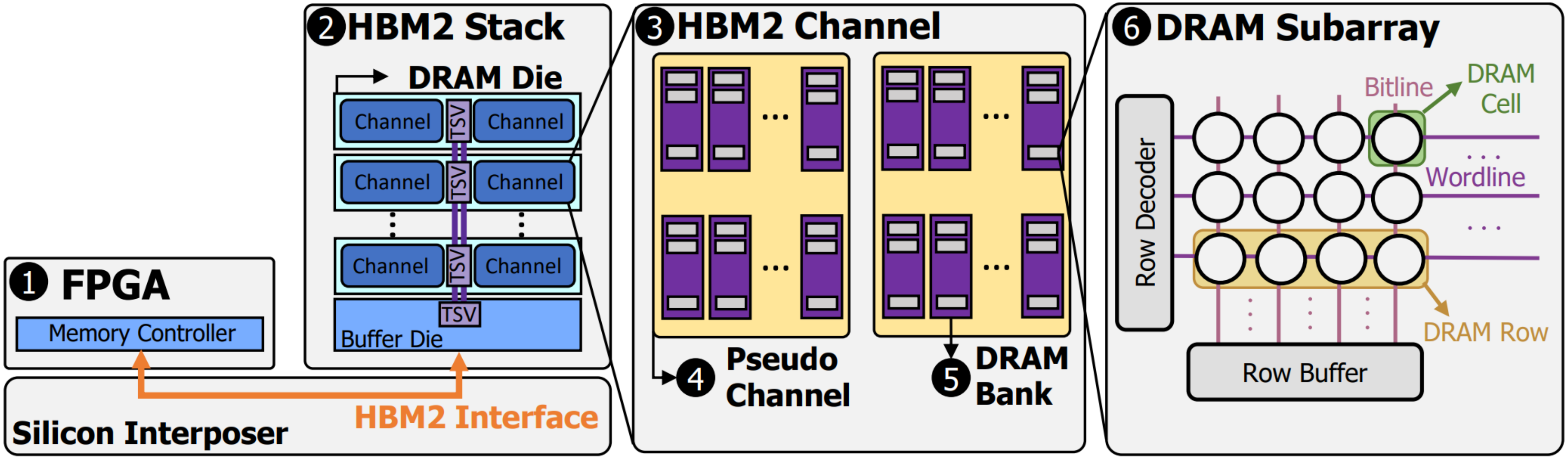
Yahya Can Tugrul   *Jeremie S. Kim*   *Victor van der Veen*
Kaveh Razavi   *Onur Mutlu*

Uncovering TRR: New Methodology, Custom RowHammer Patterns & Implications - MICRO'21 Long Talk; 25m

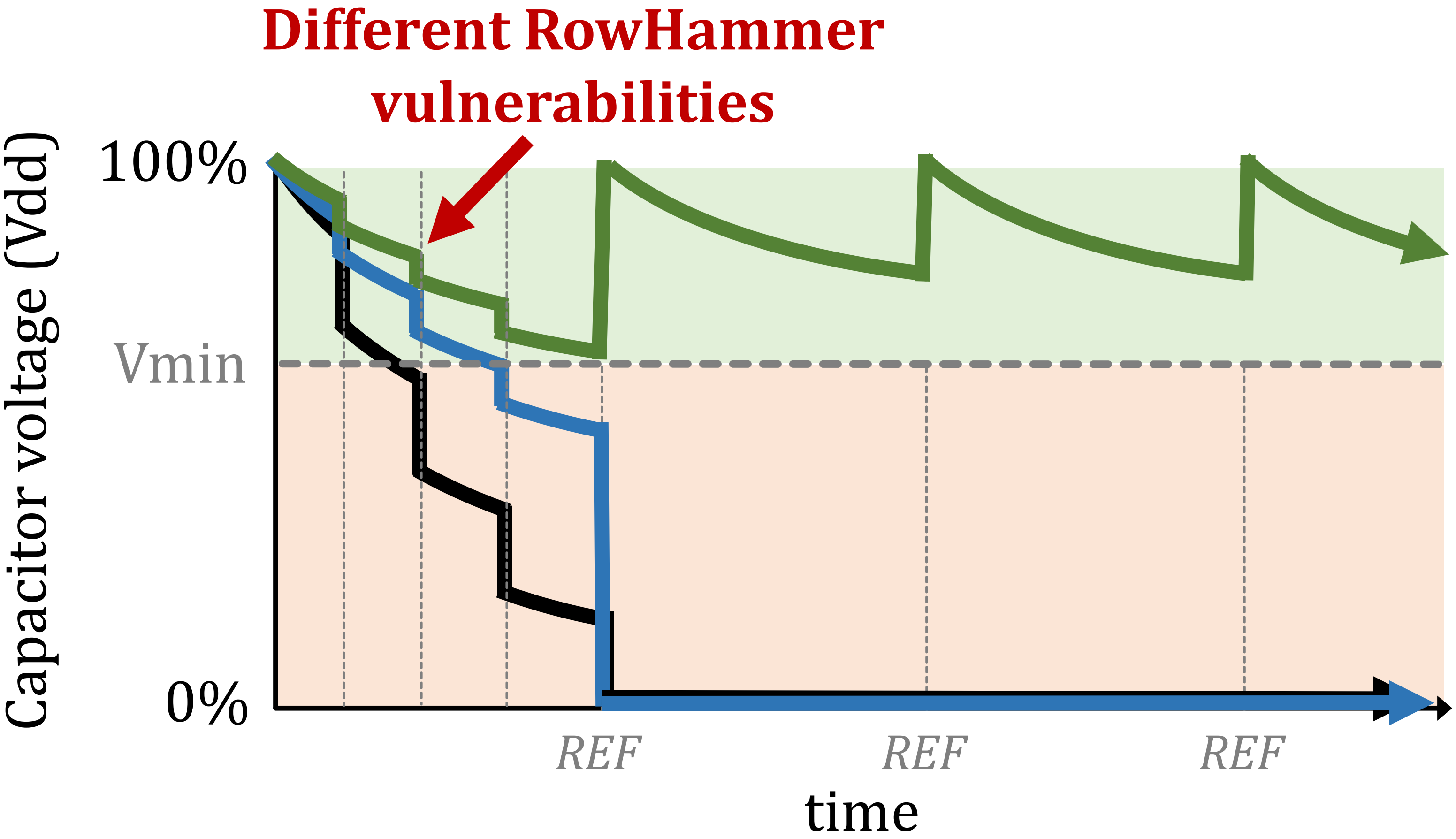
Onur Mutlu Lectures
33.6K subscribers

360 views 1 year ago MICRO 2021 Conference Presentations
Talk: "Uncovering In-DRAM RowHammer Protection Mechanisms: A New Methodology, Custom RowHammer Patterns, and Implications"
Full Conference Talk at MICRO 2021 by Hasan Hassan
25 minutes

HBM2 Organization

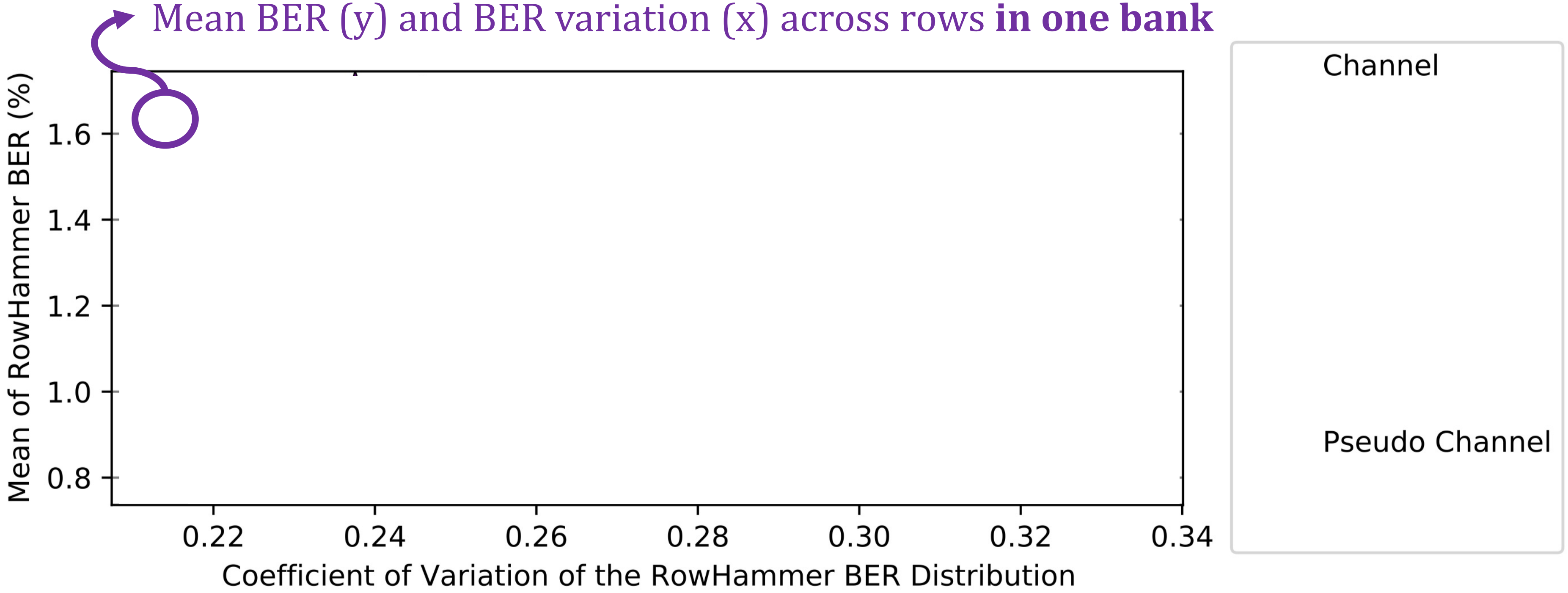


Cell-to-Cell Variation



Some cells are more vulnerable
due to **process variation** and **design-induced variation**

Variation in Bit Error Rate

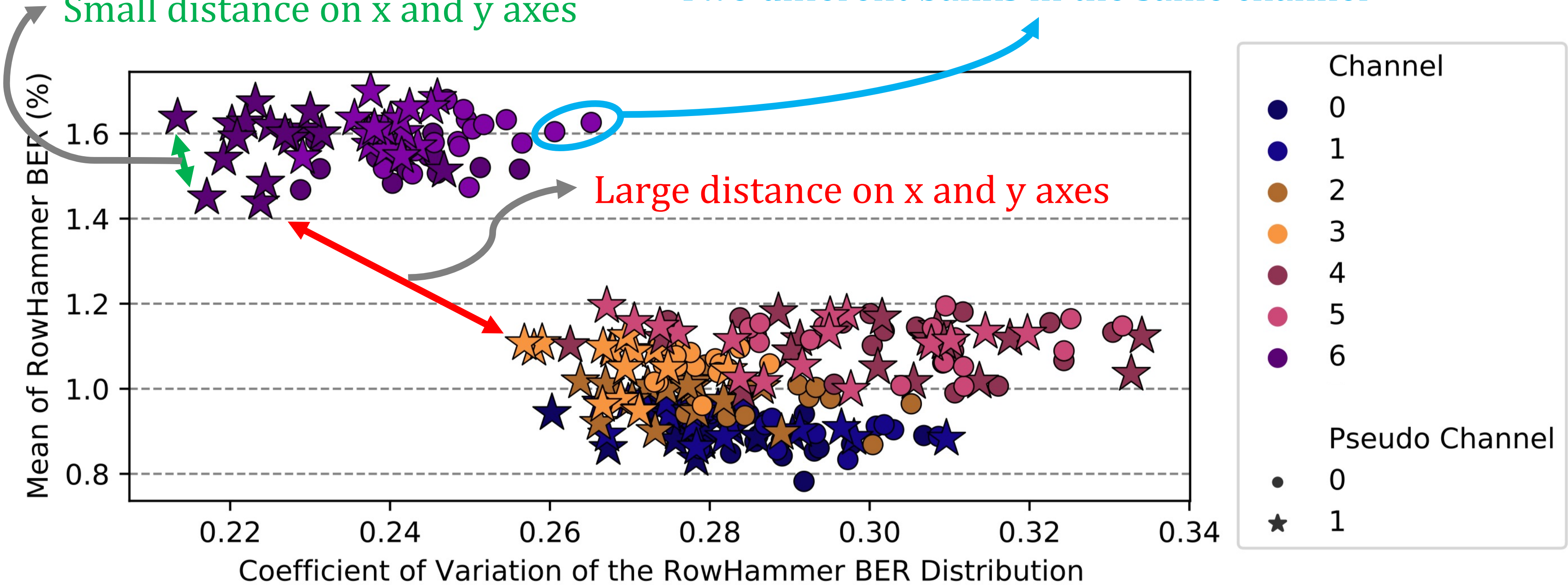


Mean BER (y) and BER variation (x) across rows **in one bank**

Variation in BER
across rows decrease

Variation in BER
across rows increase

Variation in Bit Error Rate



Small distance on x and y axes

Two different banks in the same channel

Large distance on x and y axes

Banks in the same channel have similar variation in BER