

ColumnDisturb

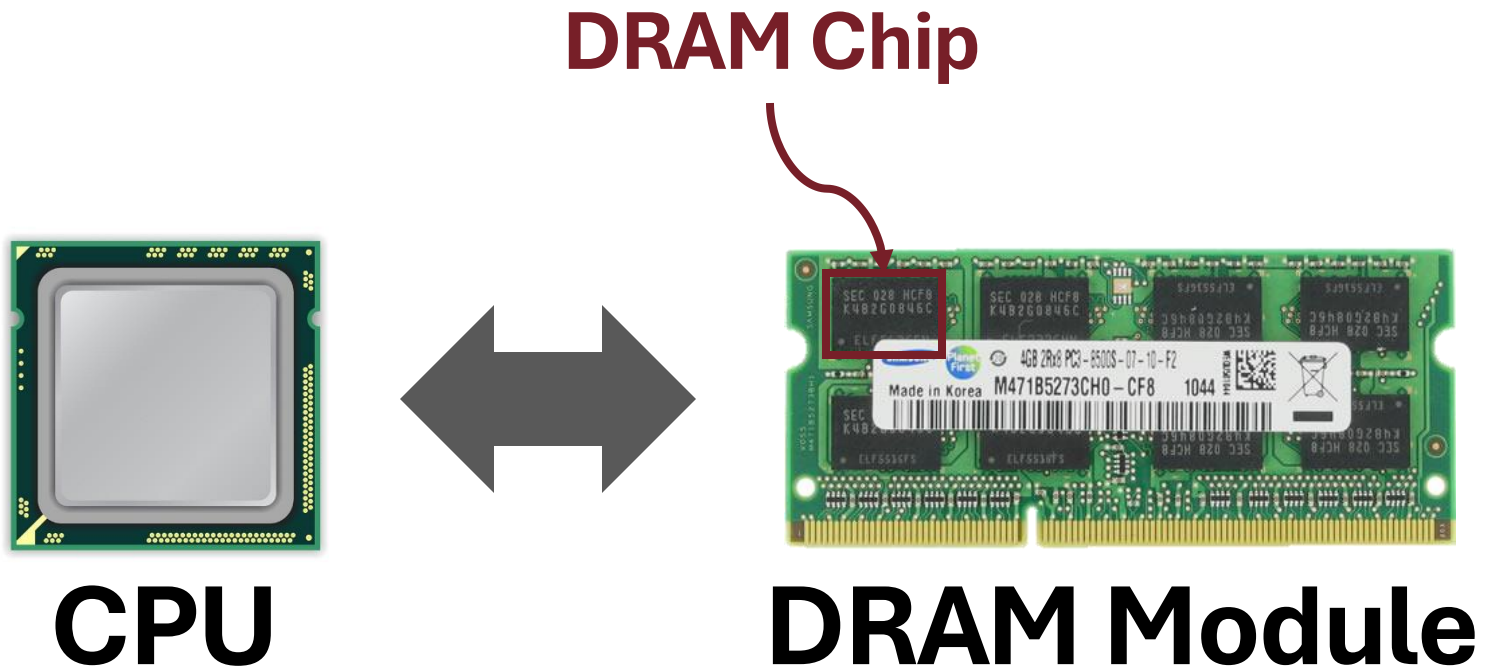
Understanding Column-based Read Disturbance in Real DRAM Chips and Implications for Future Systems

İsmail Emir Yüksel

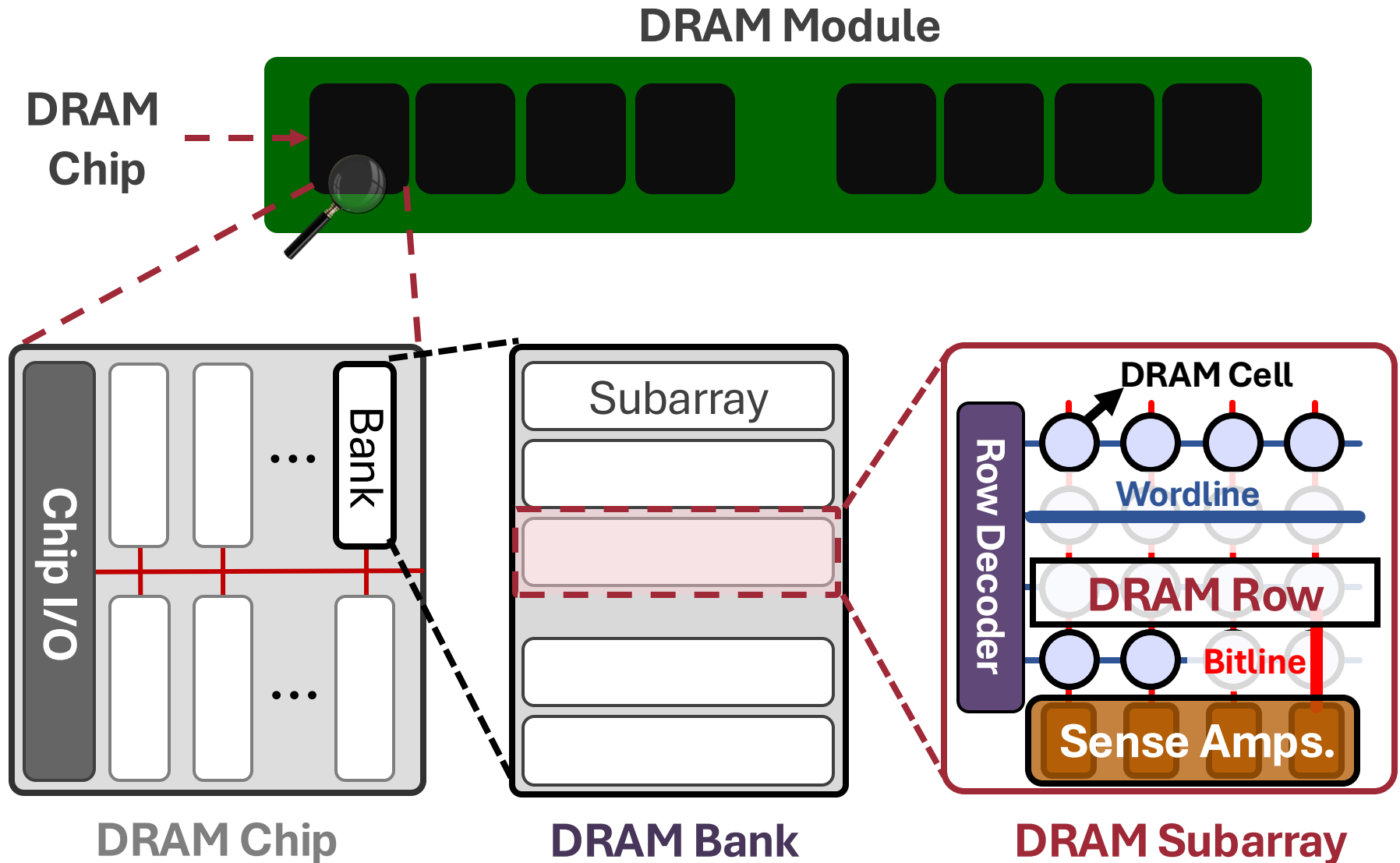
Ataberk Olgun F. Nisa Bostancı Haocong Luo

A. Giray Yağlıkçı Onur Mutlu

A Typical DRAM-Based Computing System

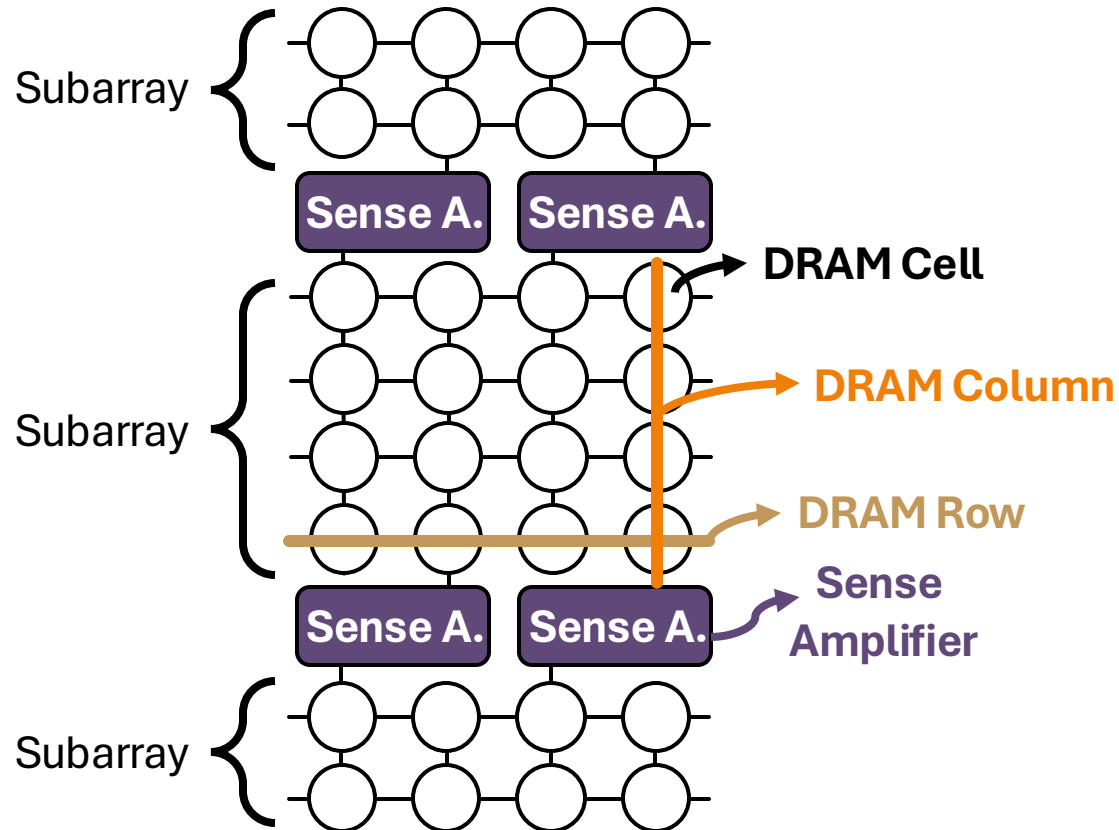


DRAM Organization



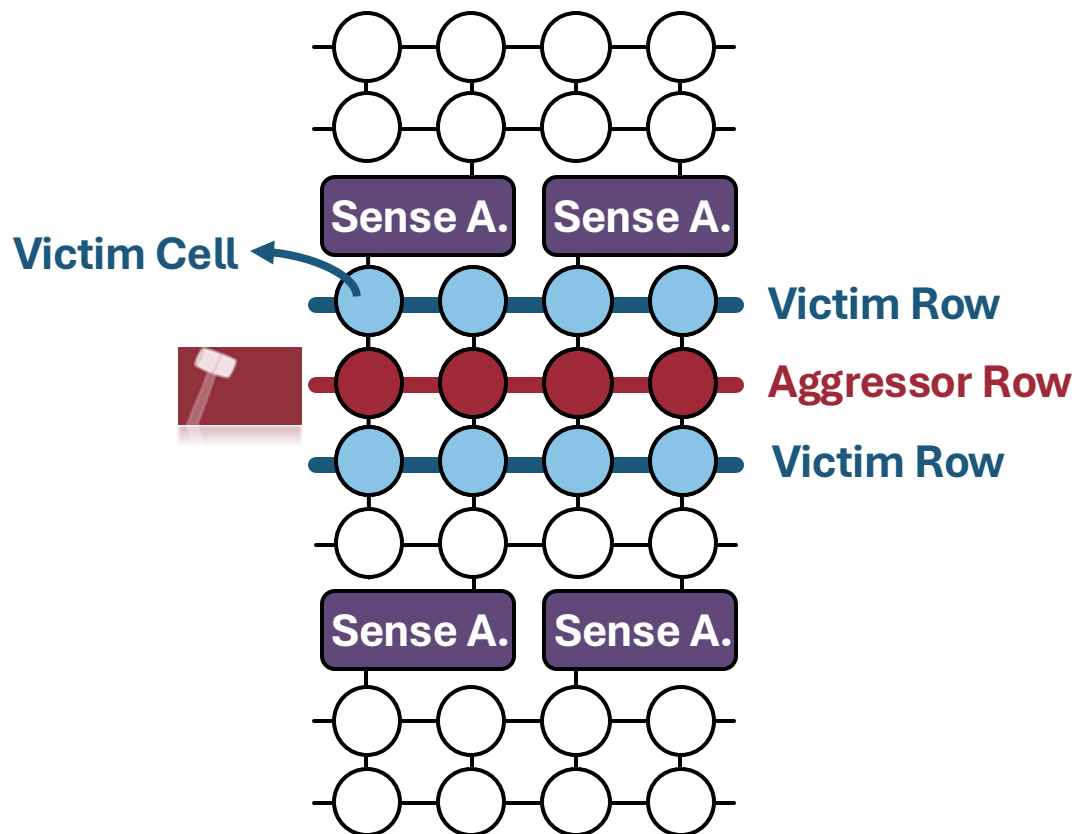
Read Disturbance in DRAM (I)

- Read disturbance in DRAM **breaks memory isolation**
- Prominent example: **RowHammer**



Read Disturbance in DRAM (I)

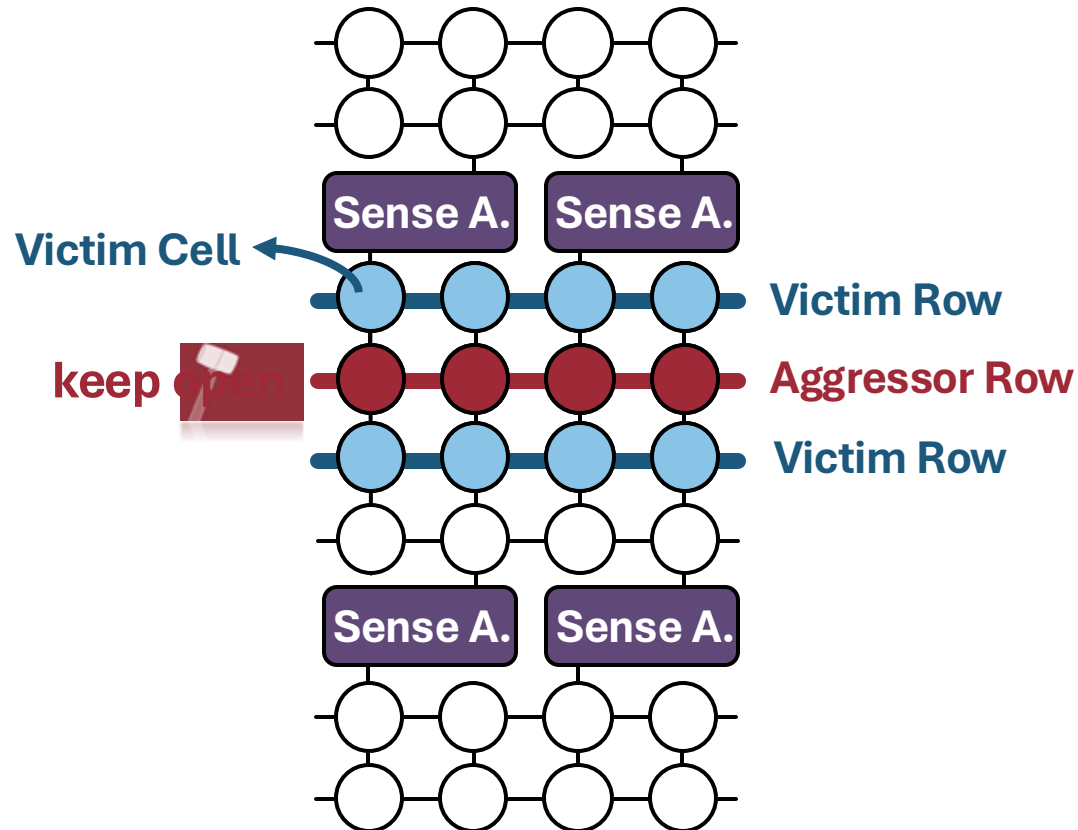
- Read disturbance in DRAM **breaks memory isolation**
- Prominent example: **RowHammer**



Repeatedly **opening** and **closing** a DRAM row **many times** causes **bitflips** in adjacent rows in the same subarray

Read Disturbance in DRAM (II)

- Read disturbance in DRAM **breaks memory isolation**
- A new read disturbance phenomenon: **RowPress**



Keeping a DRAM row **open for a long time** causes bitflips in adjacent rows in the same subarray

Read Disturbance in DRAM (III)

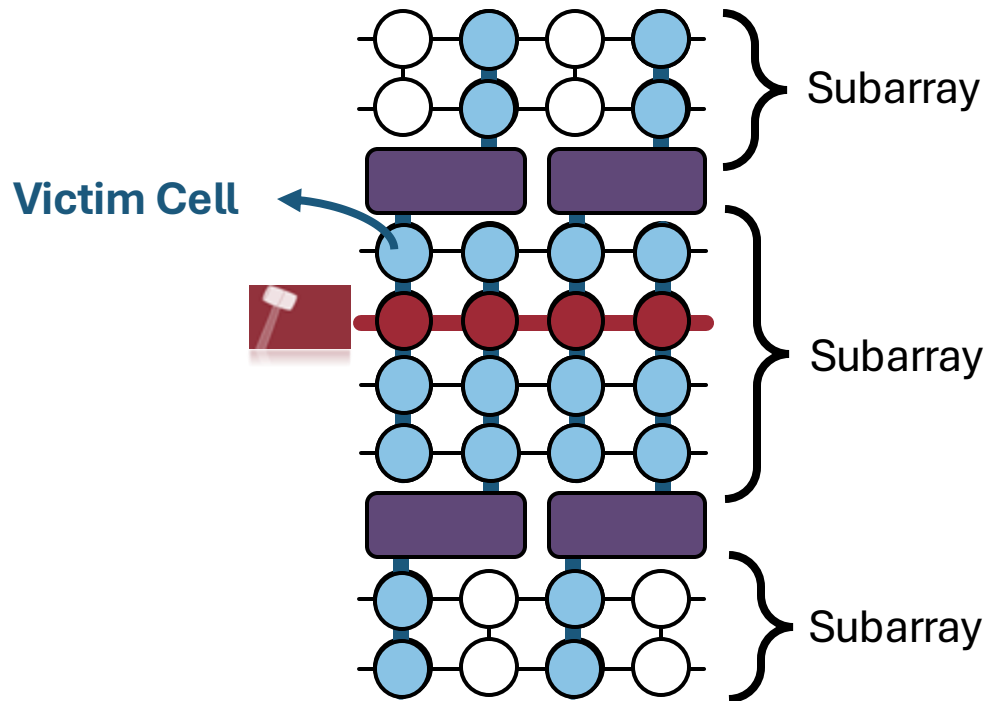
- RowHammer and RowPress are **row-based read disturbance phenomena** that affect **only a few neighboring victim rows**
- Most of the mitigations work by
 - 1) detecting **high row activation count** or **high row open time**
 - 2) refreshing **a few neighboring rows** (e.g., +/-4)

Detecting and refreshing **even only a few** neighboring rows is shown to be **a difficult problem**

What if there is a **much worse**
read disturbance phenomenon
that affects
thousands of DRAM rows
through the **bitlines**?

ColumnDisturb

By repeatedly opening or keeping a DRAM row open, it is possible to **disturb** many DRAM cells on the same DRAM column



With **ColumnDisturb**, the activation of a single row **concurrently** affects DRAM cells across **as many as three DRAM subarrays** (e.g., up to 3072 DRAM rows)

Outline

What is ColumnDisturb?

Testing Infrastructure

Real DRAM Chip Results

Implications

Conclusion

Outline

What is ColumnDisturb?

Testing Infrastructure

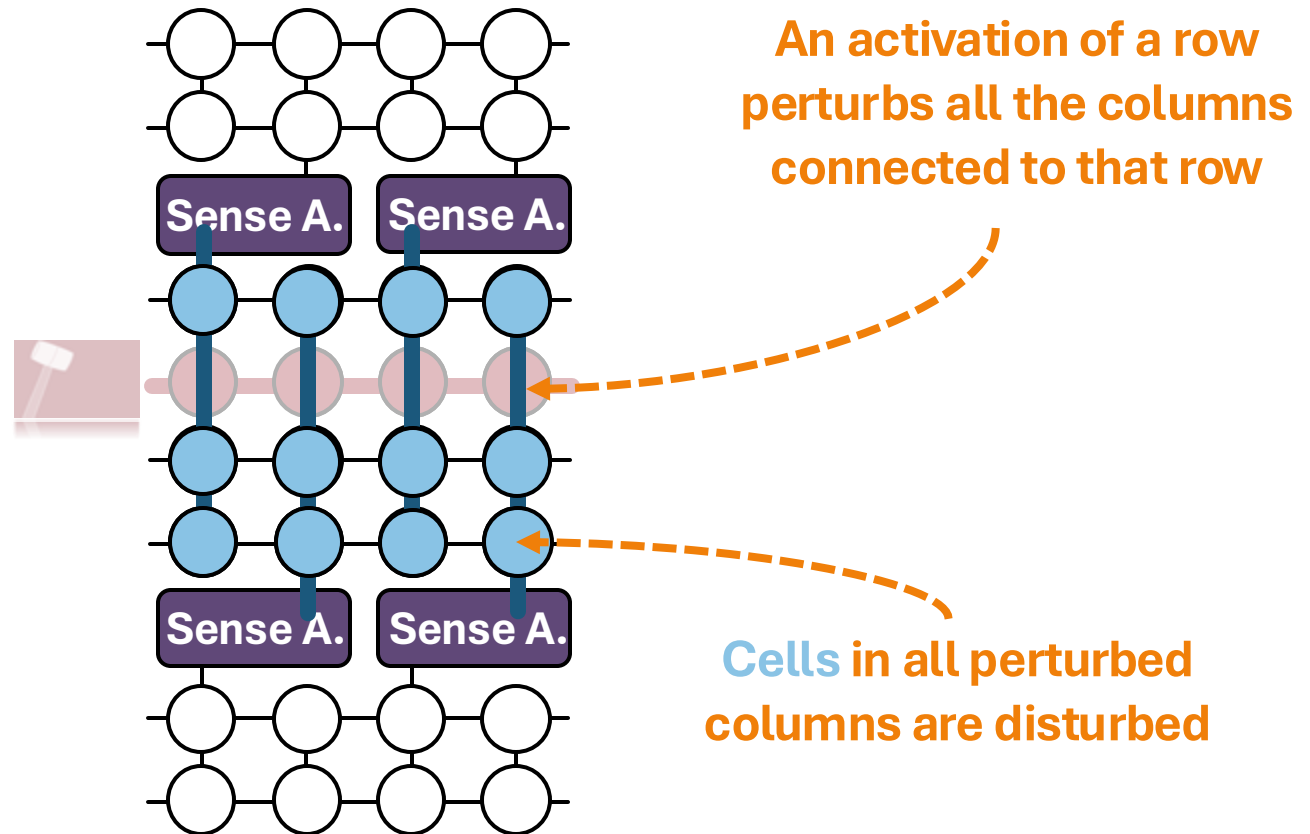
Real DRAM Chip Results

Implications

Conclusion

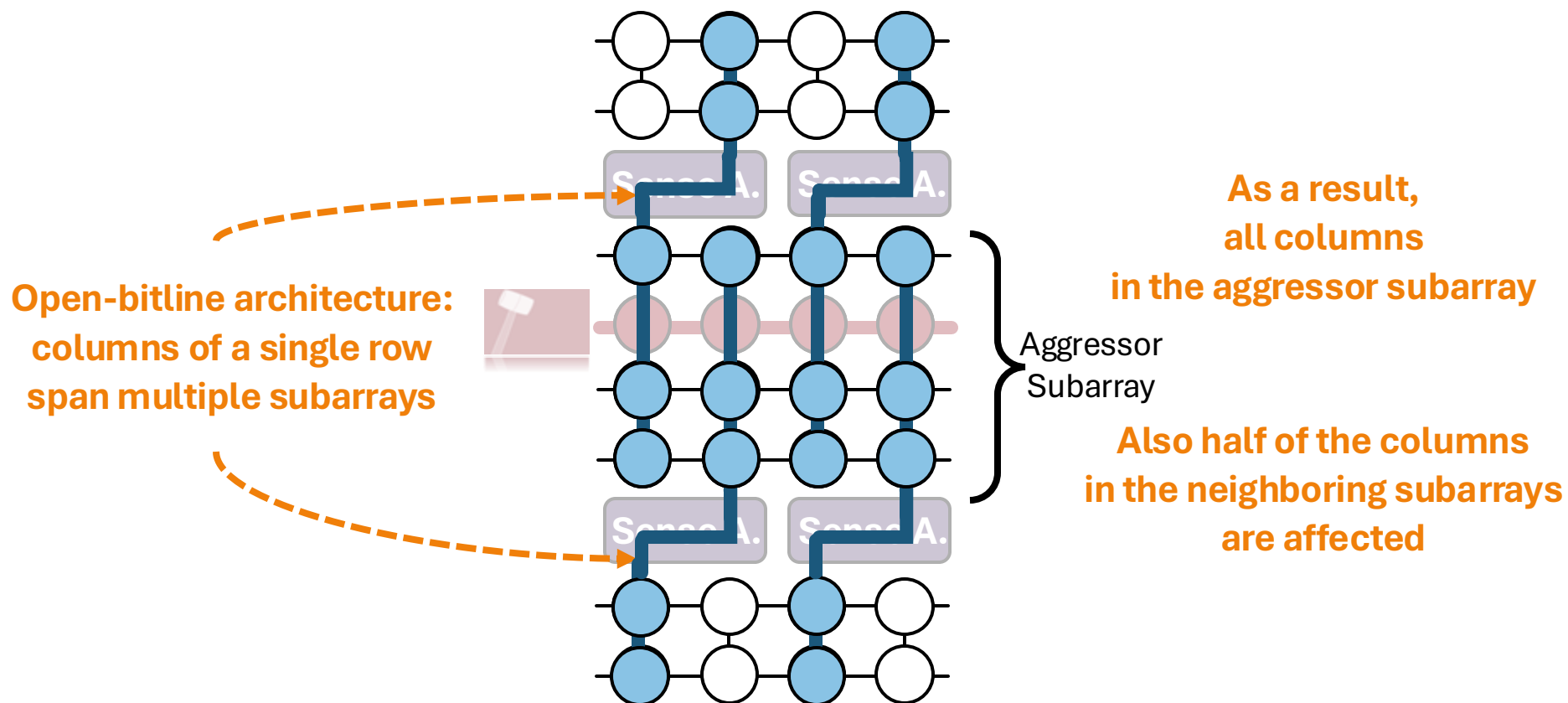
What is ColumnDisturb?

- ColumnDisturb is a **column-based** read disturbance phenomenon that **affects thousands of rows**



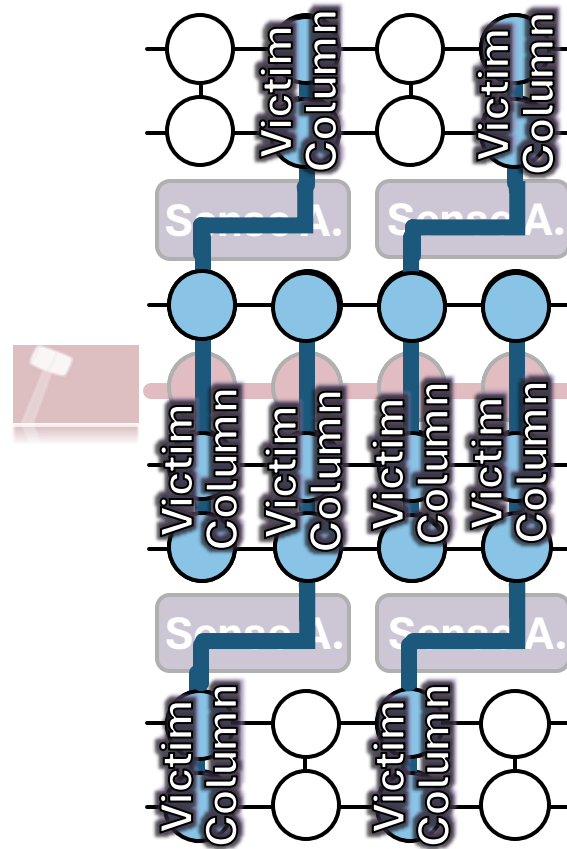
What is ColumnDisturb?

- ColumnDisturb is a **column-based** read disturbance phenomenon that **affects thousands of rows**



What is ColumnDisturb?

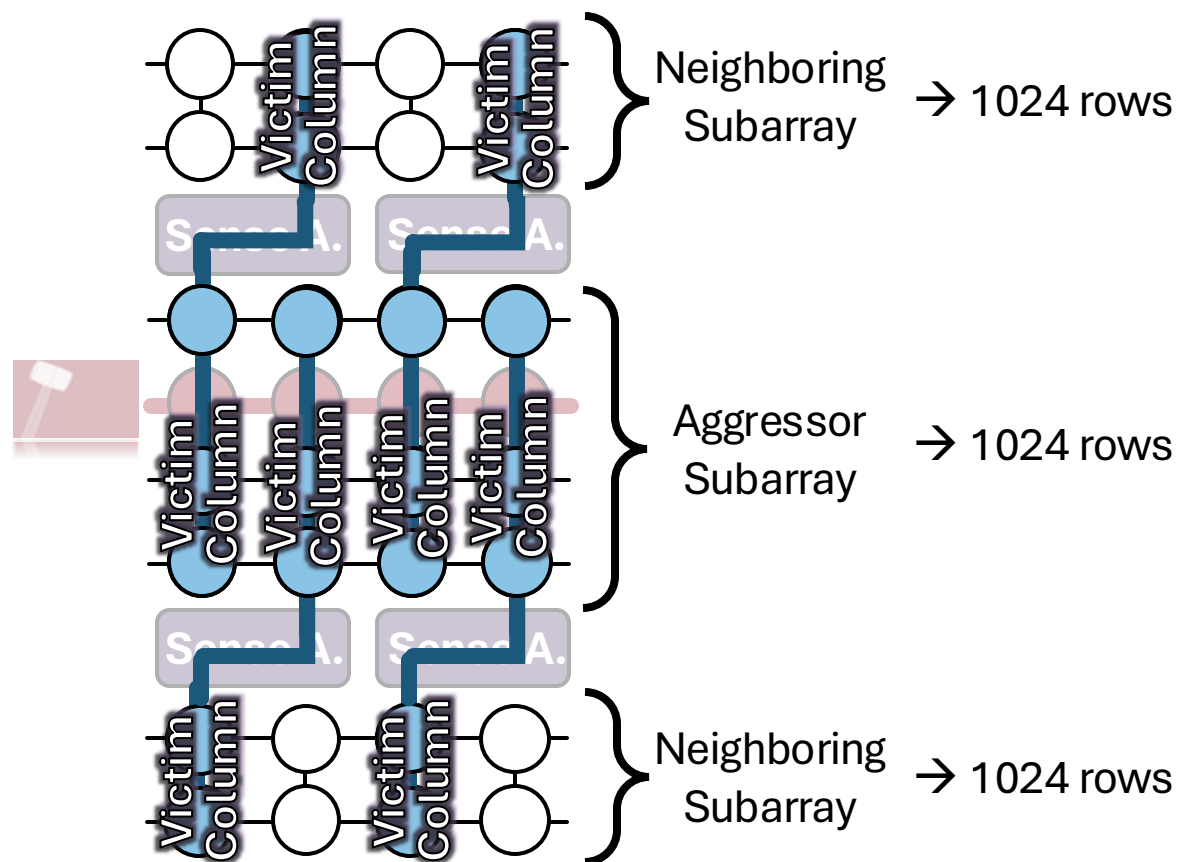
- ColumnDisturb is a **column-based** read disturbance phenomenon that **affects thousands of rows**



ColumnDisturb affects cells **vertically connected to the same bitline**.
DRAM columns are the victims

What is ColumnDisturb?

- ColumnDisturb is a **column-based** read disturbance phenomenon that **affects thousands of rows**



ColumnDisturb affects cells

across both **the same subarray and multiple neighboring subarrays**

Outline

What is ColumnDisturb?

Testing Infrastructure

Real DRAM Chip Results

Implications for Future Systems & Mechanisms

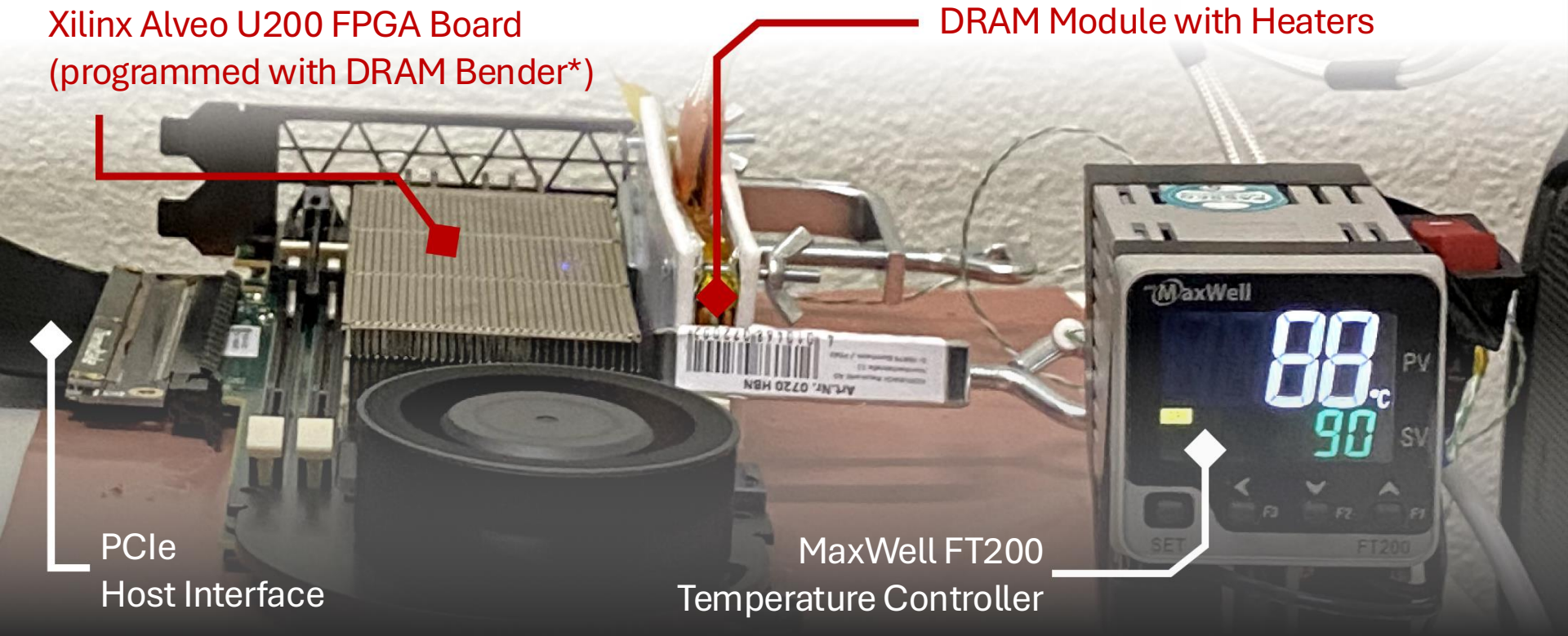
Conclusion

DDR4 DRAM Testing Infrastructure

DRAM Bender on a Xilinx Virtex UltraScale+ XCU200

Xilinx Alveo U200 FPGA Board
(programmed with DRAM Bender*)

DRAM Module with Heaters



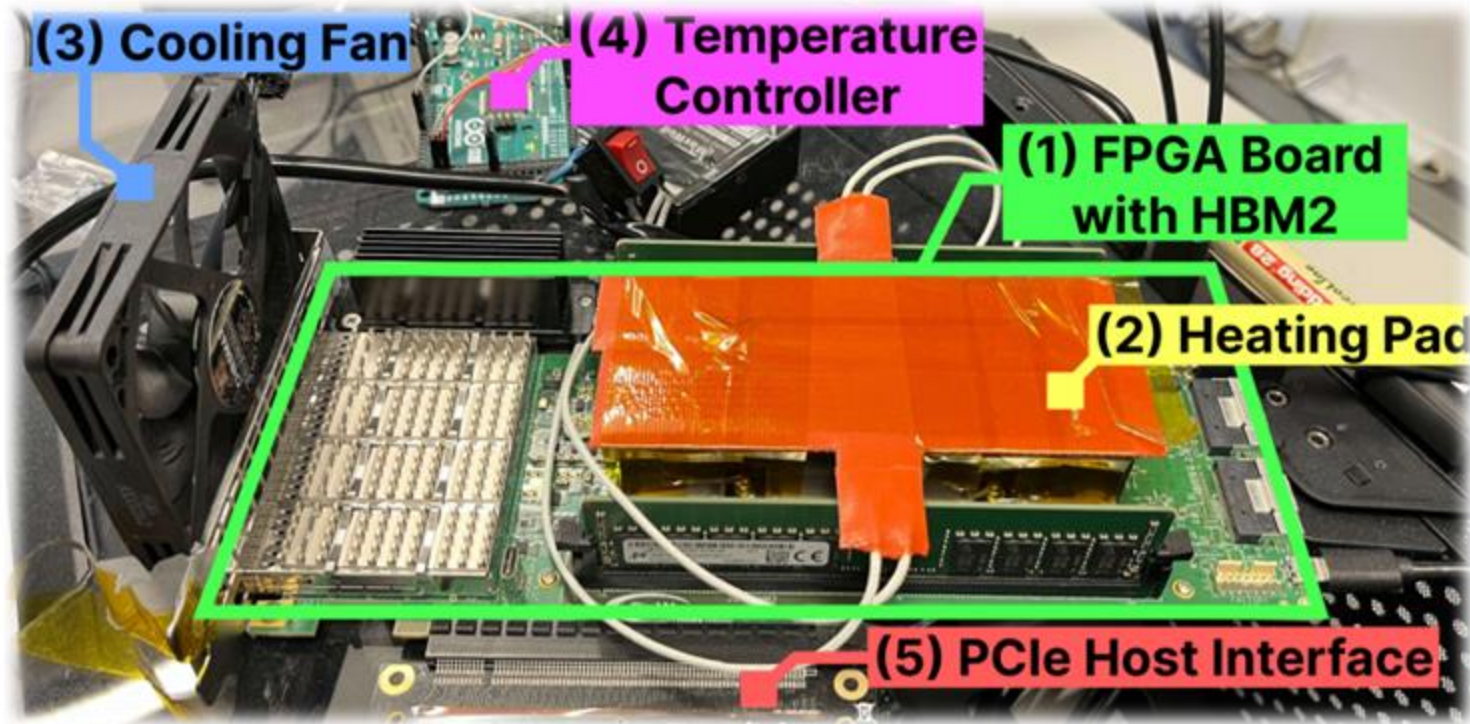
PCIe
Host Interface

MaxWell FT200
Temperature Controller

Fine-grained control over DRAM commands,
timing parameters ($\pm 1.5\text{ns}$), and temperature ($\pm 0.5^\circ\text{C}$)

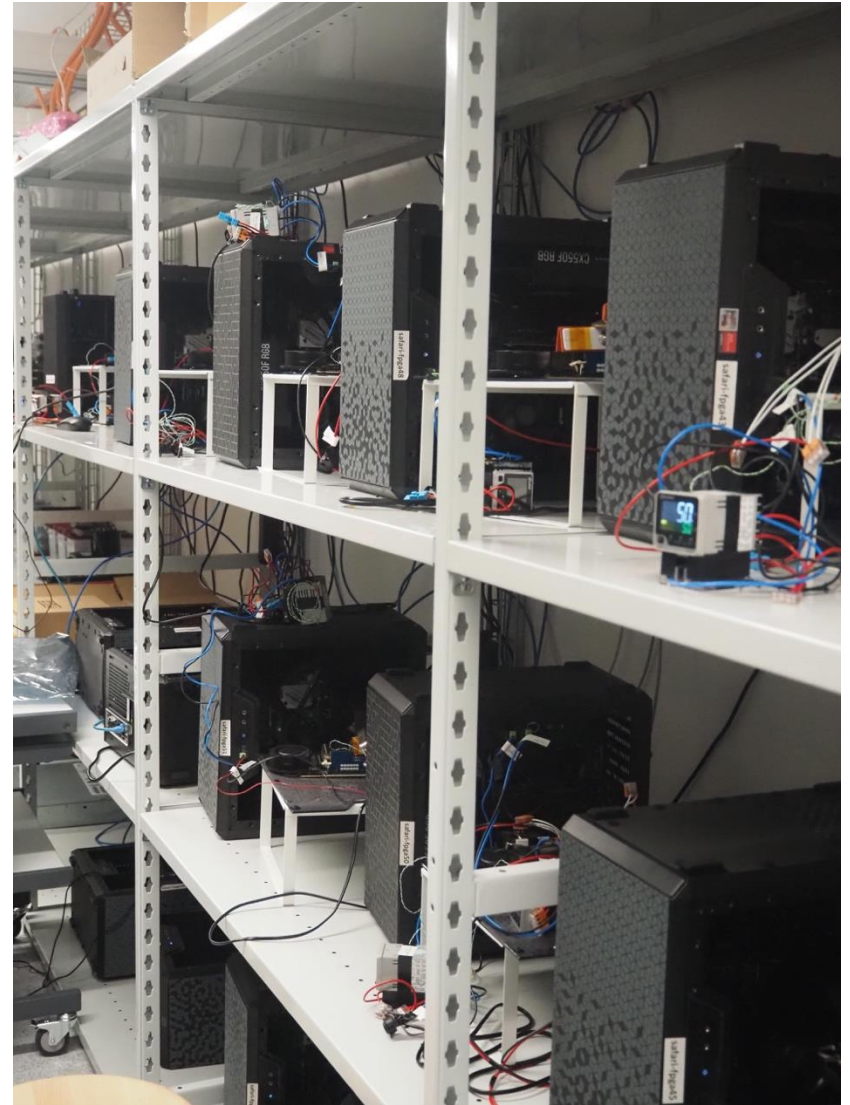
HBM2 DRAM Testing Infrastructure

DRAM Bender on a Bittware XUPVH

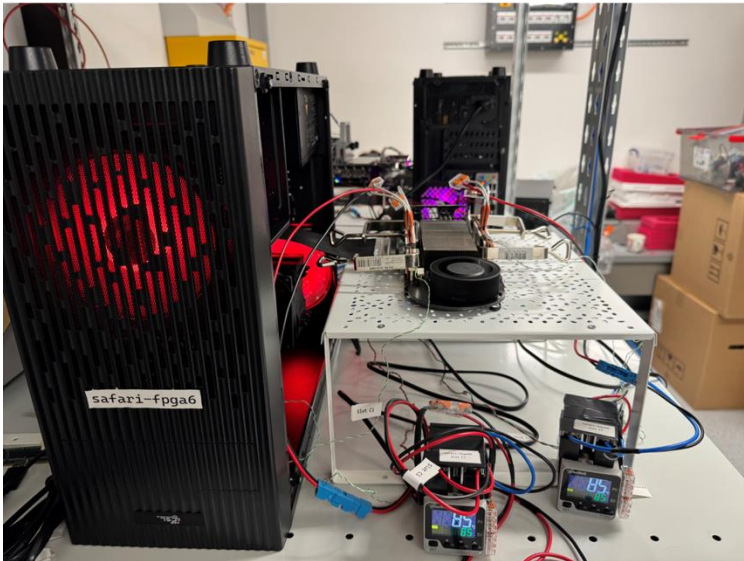


Fine-grained control over DRAM commands,
timing parameters ($\pm 1.67\text{ns}$), and temperature ($\pm 0.5^\circ\text{C}$)

Laboratory for Understanding Memory



Laboratory for Understanding Memory



Tested DRAM Chips

216 DDR4 and 4 HBM2 Chips from SK Hynix, Micron, Samsung

Chip Mfr.	Module IDs	#Chips	Die Rev.	Density	Org.
SK Hynix	H0,H1,H2	24	A	8Gb	x8
	H3,H4,H5,H6	32	D	8Gb	x8
	H7	8	A	16Gb	x8
	H8,H9	16	C	16Gb	x8
Micron	M0	8	B	4Gb	x8
	M1,M2,M3	24	R	8Gb	x8
	M4, M5	16	B	16Gb	x8
	M6, M7	8	E	16Gb	x16
	M8,M9,M10,M11	32	F	16Gb	x8
Samsung	S0, S1	16	A	16Gb	x8
	S2, S3	16	B	16Gb	x8
	S4, S5	16	C	16Gb	x16
Samsung	HBM2 Chips	4	N/A	N/A	N/A

^a We report “N/A” if the information is not publicly available.

Testing Methodology: Sources of Interference

Prevent sources of interference during core test loop

- **No DRAM refresh**
 - to avoid refreshing victim rows
- **No read disturbance mitigation mechanisms**
 - to observe circuit-level effects
- **No error correcting codes (ECC)**
 - to observe all bitflips
- **Filter out Retention & RowHammer/RowPress bitflips**
 - to demonstrate that ColumnDisturb bitflips are new and unique

ColumnDisturb: Understanding Column-based Read Disturbance in Real DRAM Chips and Implications for Future Systems

İsmail Emir Yüksel¹

Ataberk Olgun¹

F. Nisa Bostancı¹

Haocong Luo¹

A. Giray Yağlıkçı^{1,2}

Onur Mutlu¹

¹ETH Zürich

²CISPA

We experimentally demonstrate a new widespread read disturbance phenomenon, ColumnDisturb, in real commodity DRAM chips. By repeatedly opening or keeping a DRAM row (aggressor row) open, we show that it is possible to disturb DRAM cells through a DRAM column (i.e., bitline) and induce bitflips in DRAM cells sharing the same columns as the aggressor row (across multiple DRAM subarrays). With ColumnDisturb, the activation of a single row concurrently disturbs DRAM cells across as many as three DRAM subarrays (e.g., up to 3072 DRAM rows in tested DDR4 DRAM chips) as opposed to RowHammer & RowPress, which affect only a few neighboring rows of the aggressor row in a single subarray. We rigorously and comprehensively characterize ColumnDisturb and its characteristics under various operational conditions (i.e., temperature, data pattern, DRAM timing parameters, average voltage level of the bitline, memory access pattern, and spatial variation) using 216 DDR4 and 4 HBM2 chips from three major DRAM manufacturers. Among our 27 key experimental observations, we highlight two major results and their implications.

1. Introduction

Dynamic random access memory (DRAM) [1] is the dominant main memory technology in nearly all modern computing systems due to its latency and cost characteristics. Continuing to increase DRAM capacity requires increasing the density of DRAM cells by reducing (i.e., scaling) the technology node size of DRAM, which in turn reduces DRAM cell size, cell-to-cell spacing, and cell-to-bitline spacing [2]. Unfortunately, as a result of this aggressive technology node scaling, DRAM suffers from worsening read disturbance problems [3–6].

Read disturbance in modern DRAM chips [4–9] is a widespread phenomenon and is reliably used for breaking memory isolation [5, 7, 9–70], a fundamental building block of robust (i.e., safe, secure, reliable, available) systems. RowHammer and RowPress are two prominent examples of DRAM read disturbance phenomena where a victim DRAM row experiences bitflips when a nearby aggressor DRAM row is 1) repeatedly activated (i.e., hammered) [5–7] or 2) kept open for a long period (i.e., nressed) [4, 71]. Many prior works [3–5, 7, 9, 41, 42]

<https://arxiv.org/pdf/2510.14750>

Testing Methodology: Metric & Access Pattern

Minimum time required to induce the first **ColumnDisturb** bitflip

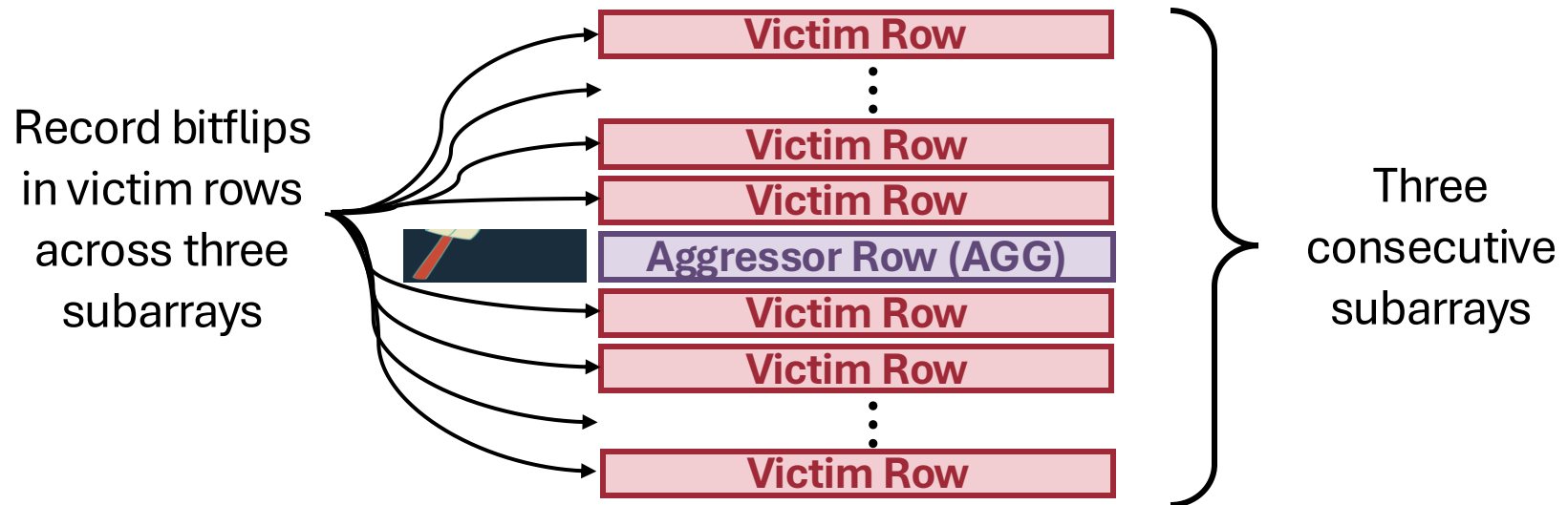
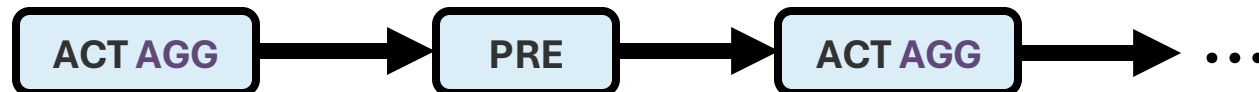
Lower is worse

Number of ColumnDisturb bitflips in a subarray

Higher is worse

Number of rows with ColumnDisturb bitflips in a subarray

Higher is worse



Testing Methodology: Test Parameters

We carefully sweep

- Data Pattern
 - 0x00, 0xAA, 0x33, and 0x77 (and their inverse pattern)
- Temperature (°C)
 - 45, 65, 85, and 95
- Aggressor Row On Time (tAggOn)*
 - 36ns, 7.8us, 70.2us, and 1ms
- DRAM Refresh Interval
 - 64ms, 128ms, 256ms, 512ms, 1s, 2s, 4s, 8s, and 16s
- Aggressor Row Location
 - Aggressor subarray's first, middle, and last rows
- Tested Subarrays (Rows)
 - Up to +49K (+4.5M): *all subarrays in all banks across all DRAM chips*

Outline

What is ColumnDisturb?

Testing Infrastructure

Real DRAM Chip Results

Implications

Conclusion

Key Takeaways from Major Characterization Experiments

Takeaway 1

ColumnDisturb is a **widespread read disturbance phenomenon** and **becomes worse** as DRAM technology **scales down to smaller** node sizes

Takeaway 2

ColumnDisturb induces bitflips in **thousands of DRAM rows** and is **different** from **RowHammer, RowPress, and Retention** failures

Takeaway 3

The **bitflip directionalities** of ColumnDisturb and RowHammer & RowPress are **significantly different**

Takeaway 4

Significantly more rows are vulnerable to ColumnDisturb **than** retention failures

Takeaway 5

As **temperature increases**,
DRAM chips **become more vulnerable** to ColumnDisturb

Key Takeaways from Major Characterization Experiments

Takeaway 1

ColumnDisturb is a **widespread read disturbance phenomenon** and **becomes worse** as DRAM technology **scales down to smaller** node sizes

Takeaway 2

ColumnDisturb induces bitflips in **thousands of DRAM rows** and is **different** from **RowHammer, RowPress, and Retention** failures

Takeaway 3

The **bitflip directionalities** of ColumnDisturb and RowHammer & RowPress are **significantly different**

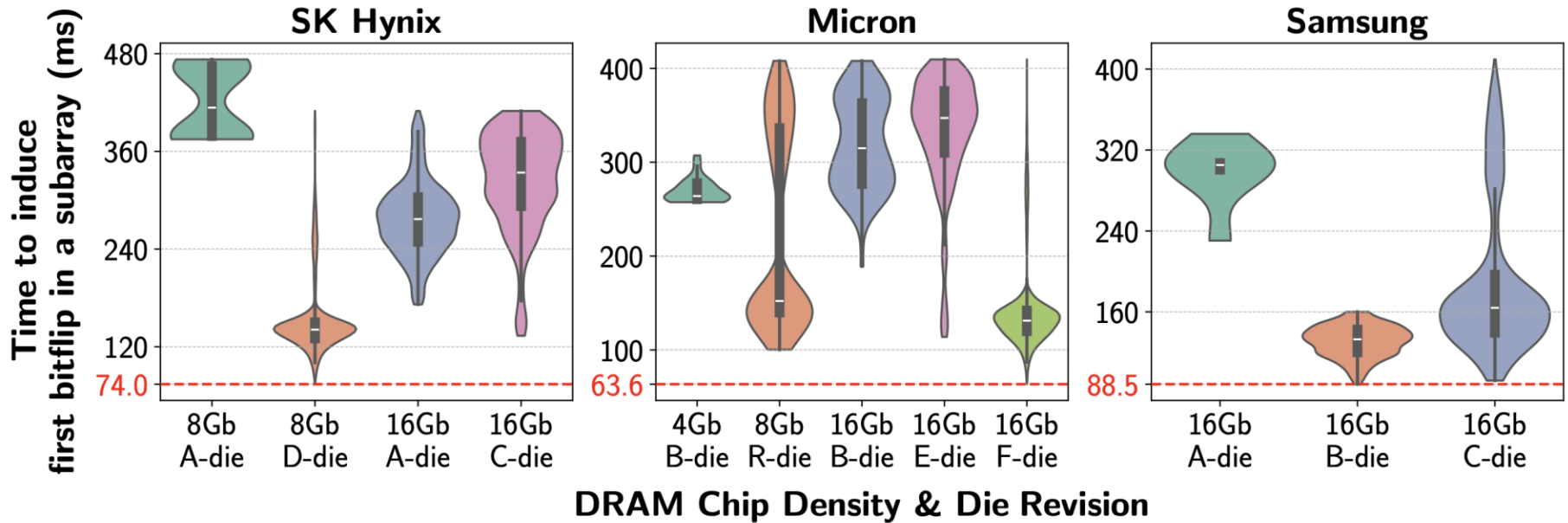
Takeaway 4

Significantly more rows are vulnerable to ColumnDisturb **than** retention failures

Takeaway 5

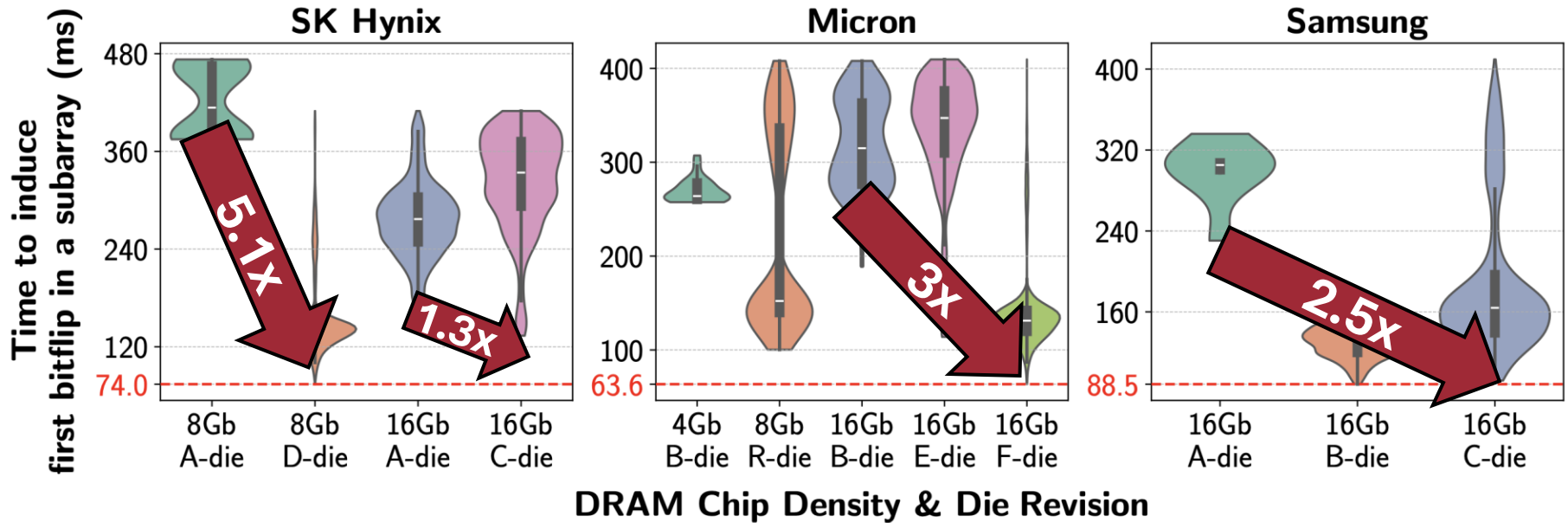
As **temperature increases**, DRAM chips **become more vulnerable** to ColumnDisturb

Minimum Time to Induce First Bitflip



All tested 216 DDR4 chips are **vulnerable** to ColumnDisturb

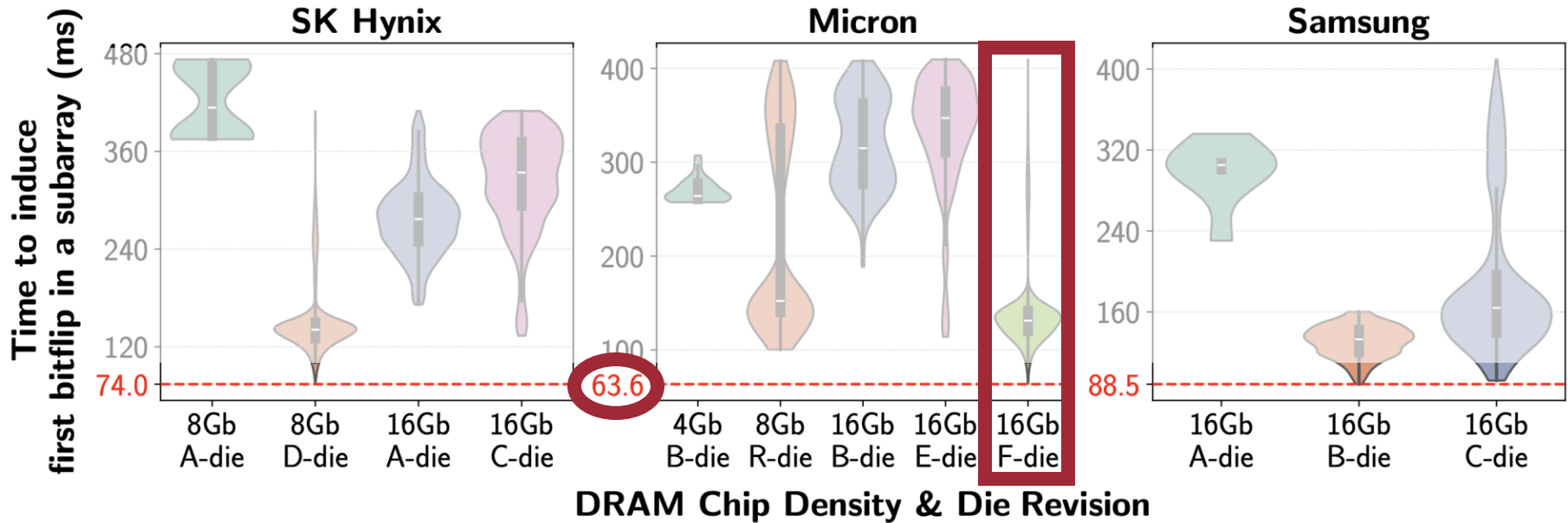
Minimum Time to Induce First Bitflip



Newer chips tend to be more vulnerable to ColumnDisturb bitflips

This trend is consistent across all tested manufacturers

Minimum Time to Induce First Bitflip



Some **real DRAM chips** experience ColumnDisturb bitflips within the nominal refresh window (64ms) under nominal operating conditions

Key Takeaways from Major Characterization Experiments

Takeaway 1

ColumnDisturb is a **widespread read disturbance phenomenon** and **becomes worse** as DRAM technology **scales down to smaller** node sizes

Takeaway 2

ColumnDisturb induces bitflips in **thousands of DRAM rows** and is **different** from **RowHammer, RowPress, and Retention** failures

Takeaway 3

The **bitflip directionalities** of ColumnDisturb and RowHammer & RowPress are **significantly different**

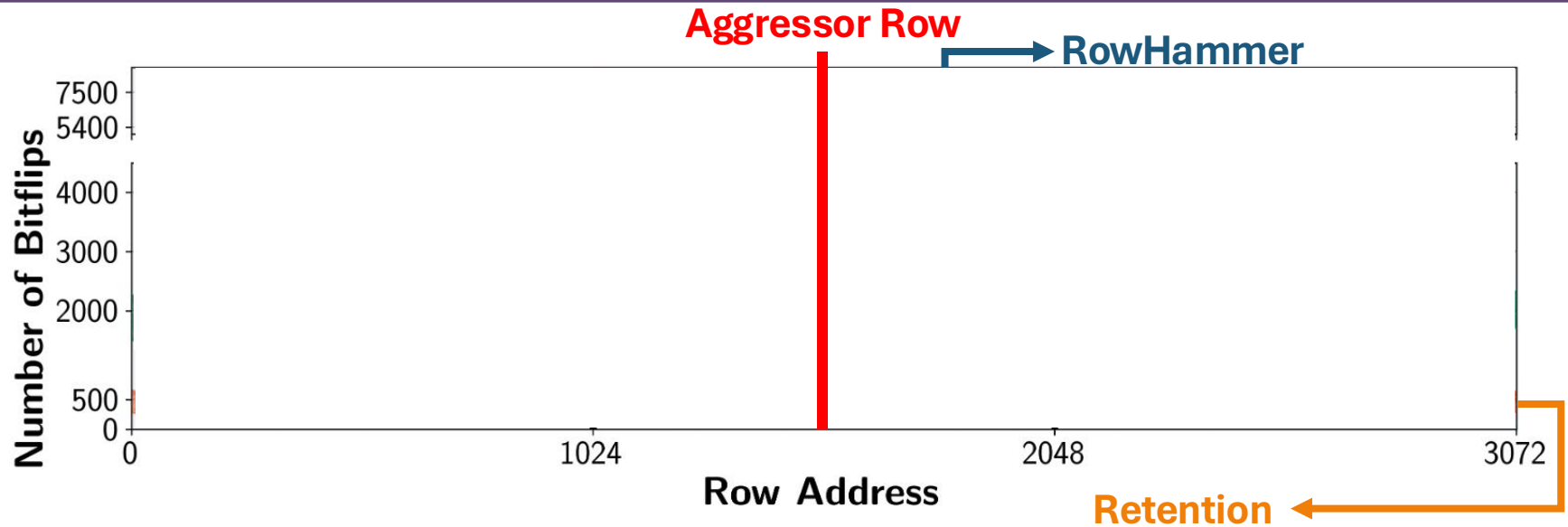
Takeaway 4

Significantly more rows are vulnerable to ColumnDisturb **than** retention failures

Takeaway 5

As **temperature increases**, DRAM chips **become more vulnerable** to ColumnDisturb

ColumnDisturb vs. RH/RP & Retention



RowHammer & RowPress induce bitflips **only**
in the immediate neighboring rows of the aggressor row

Retention failures are almost **constant** (around 500) for every row

ColumnDisturb induces **many more** bitflips than retention failures
and **affects many more rows** than RowHammer & RowPress

Key Takeaways from Major Characterization Experiments

Takeaway 1

ColumnDisturb is a **widespread read disturbance phenomenon** and **becomes worse** as DRAM technology **scales down to smaller** node sizes

Takeaway 2

ColumnDisturb induces bitflips in **thousands of DRAM rows** and is **different** from **RowHammer, RowPress, and Retention** failures

Takeaway 3

The **bitflip directionalities** of ColumnDisturb and RowHammer & RowPress are **significantly different**

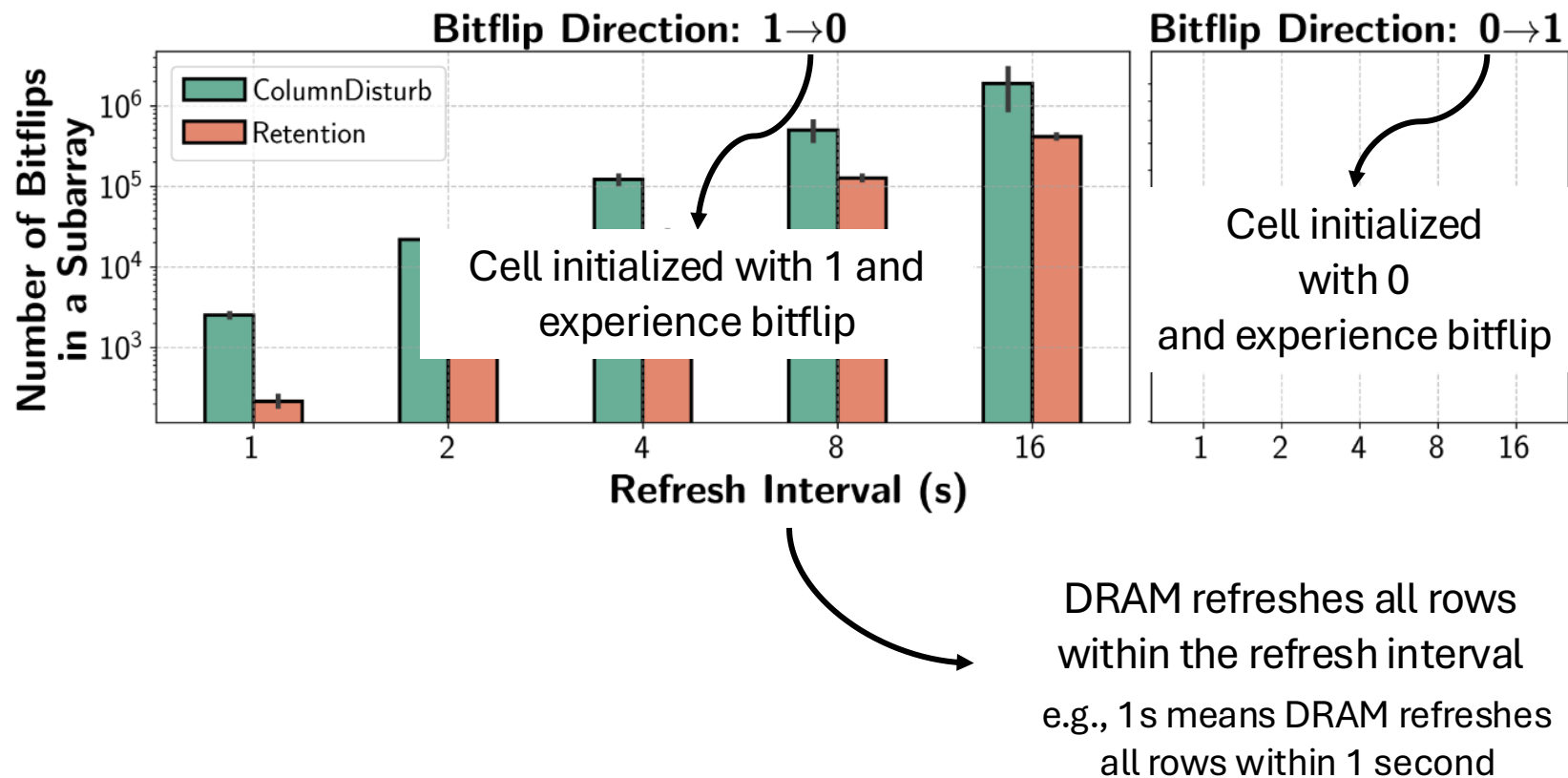
Takeaway 4

Significantly more rows are vulnerable to ColumnDisturb **than** retention failures

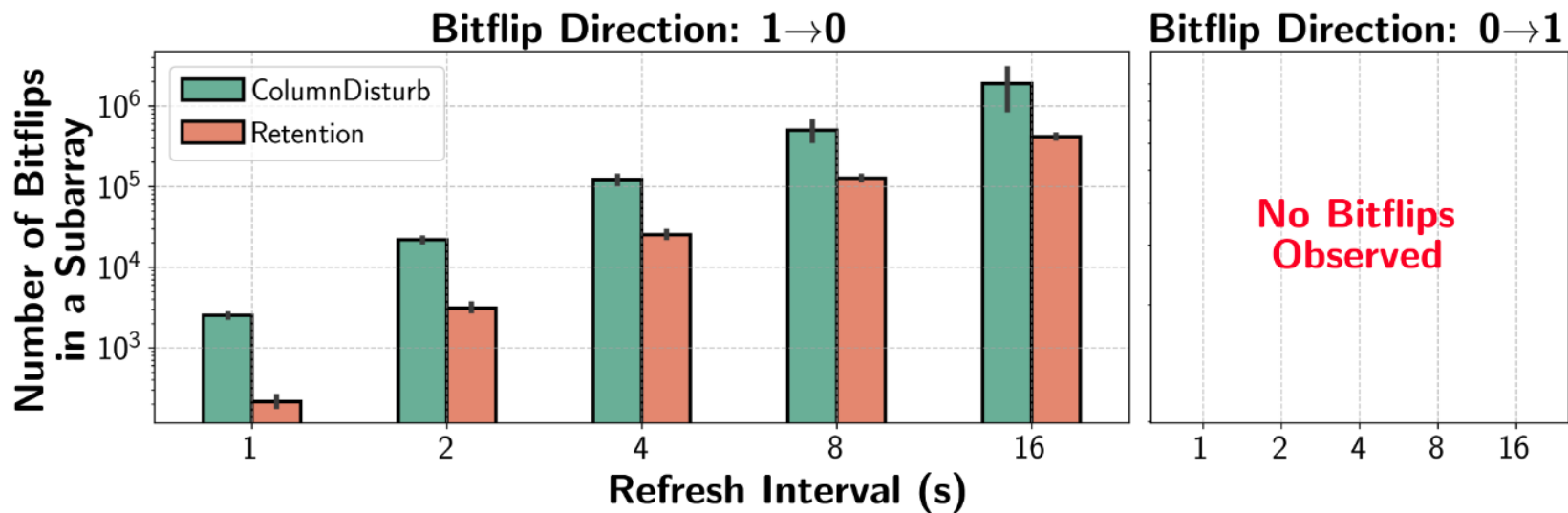
Takeaway 5

As **temperature increases**, DRAM chips **become more vulnerable** to ColumnDisturb

Bitflip Direction



Bitflip Direction



ColumnDisturb induces **only 1 to 0** bitflips, which are **the same direction** as retention failures

This is in stark contrast to **RowHammer and RowPress**, which induce bitflips in **both 1 to 0 and 0 to 1** directions

Key Takeaways from Major Characterization Experiments

Takeaway 1

ColumnDisturb is a **widespread read disturbance phenomenon** and **becomes worse** as DRAM technology **scales down to smaller** node sizes

Takeaway 2

ColumnDisturb induces bitflips in **thousands of DRAM rows** and is **different** from **RowHammer, RowPress, and Retention** failures

Takeaway 3

The **bitflip directionalities** of ColumnDisturb and RowHammer & RowPress are **significantly different**

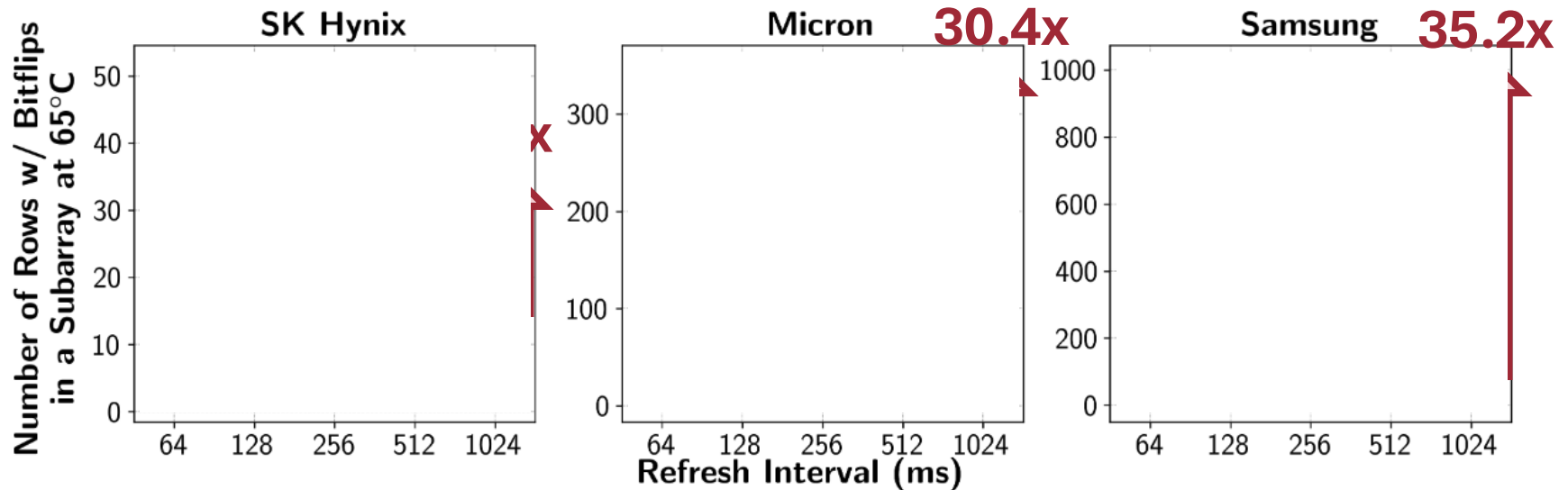
Takeaway 4

Significantly more rows are vulnerable to ColumnDisturb **than** retention failures

Takeaway 5

As **temperature increases**, DRAM chips **become more vulnerable** to ColumnDisturb

Number of Rows w/ Bitflips



ColumnDisturb induces bitflips
in significantly more **DRAM rows** than retention failures

This trend is **consistent** across **all tested manufacturers**

Key Takeaways from Major Characterization Experiments

Takeaway 1

ColumnDisturb is a **widespread read disturbance phenomenon** and **becomes worse** as DRAM technology **scales down to smaller** node sizes

Takeaway 2

ColumnDisturb induces bitflips in **thousands of DRAM rows** and is **different** from **RowHammer, RowPress, and Retention** failures

Takeaway 3

The **bitflip directionalities** of ColumnDisturb and RowHammer & RowPress are **significantly different**

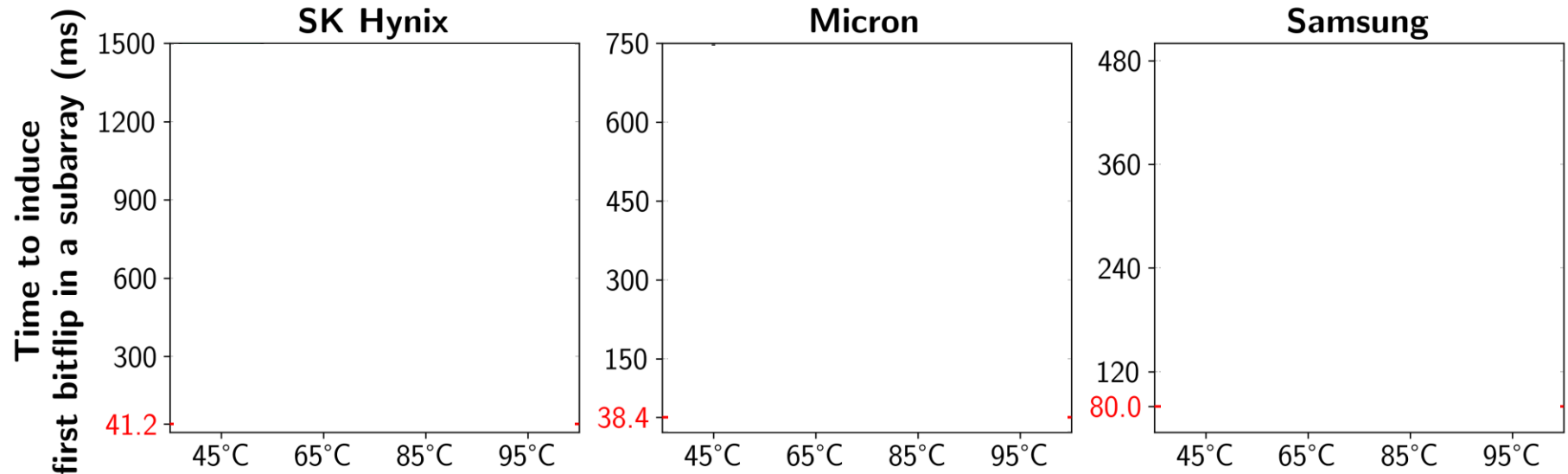
Takeaway 4

Significantly more rows are vulnerable to ColumnDisturb **than** retention failures

Takeaway 5

As **temperature increases**,
DRAM chips **become more vulnerable** to ColumnDisturb

Effect of Temperature



Time to induce the first ColumnDisturb bitflip
significantly reduces as **temperature increases**

This trend is **consistent** across **all tested manufacturers**

Outline

What is ColumnDisturb?

Testing Infrastructure

Real DRAM Chip Results

Implications

Conclusion

Implications Summary

1. Implications on Future Computing Systems

ColumnDisturb could jeopardize the robustness of computing systems

- ColumnDisturb bitflips occur in very far away rows from the aggressor row, even **in nominal operating conditions**
- ColumnDisturb is **getting worse**: newer chips are **more vulnerable**

2. Implications on Retention-Aware Refresh Mechanisms

ColumnDisturb dramatically decreases the benefits of these mechanisms

- At long refresh intervals (e.g., >512ms), **a large majority of rows** experience ColumnDisturb bitflips
- ColumnDisturb induces **significantly more bitflips** than retention failures

Implications for System Robustness

- **ColumnDisturb bitflips** occur in **very far away rows** from the aggressor row
 - E.g., one of the bitflips we observe in nominal refresh window is **374 rows away** from the aggressor row
- Most of the read disturbance mitigations refresh only **a few neighboring rows** as RowHammer and RowPress induces bitflips **in close proximity**
- **Even refreshing only a few rows** is shown to be **a difficult problem**
- **We describe and analytically evaluate two solutions**
 - **Increasing DRAM Refresh Rate**
 - Reduces DRAM **throughput** by 42.1%
 - **Proactively Refreshing Victim Rows**
 - Reduces DRAM **throughput** by 12.4%

Implications for System Robustness

ColumnDisturb: Understanding Column-based Read Disturbance in Real DRAM Chips and Implications for Future Systems

İsmail Emir Yüksel¹

Ataberk Olgun¹

F. Nisa Bostancı¹

Haocong Luo¹

A. Giray Yağlıkçı^{1,2}

Onur Mutlu¹

¹ETH Zürich

²CISPA

We experimentally demonstrate a new widespread read disturbance phenomenon, ColumnDisturb, in real commodity DRAM chips. By repeatedly opening or keeping a DRAM row (aggressor row) open, we show that it is possible to disturb DRAM cells through a DRAM column (i.e., bitline) and induce bitflips in DRAM cells sharing the same columns as the aggressor row (across multiple DRAM subarrays). With ColumnDisturb, the activation of a single row concurrently disturbs DRAM cells across as many as three DRAM subarrays (e.g., up to 3072 DRAM rows in tested DDR4 DRAM chips) as opposed to RowHammer & RowPress, which affect only a few neighboring rows of the aggressor row in a single subarray. We rigorously and comprehensively characterize ColumnDisturb and its characteristics under various operational conditions (i.e., temperature, data pattern, DRAM timing parameters, average voltage level of the bitline, memory access pattern, and spatial variation) using 216 DDR4 and 4 HBM2 chips from three major DRAM manufacturers. Among our 27 key experimental observations, we highlight two major results and their implications.

1. Introduction

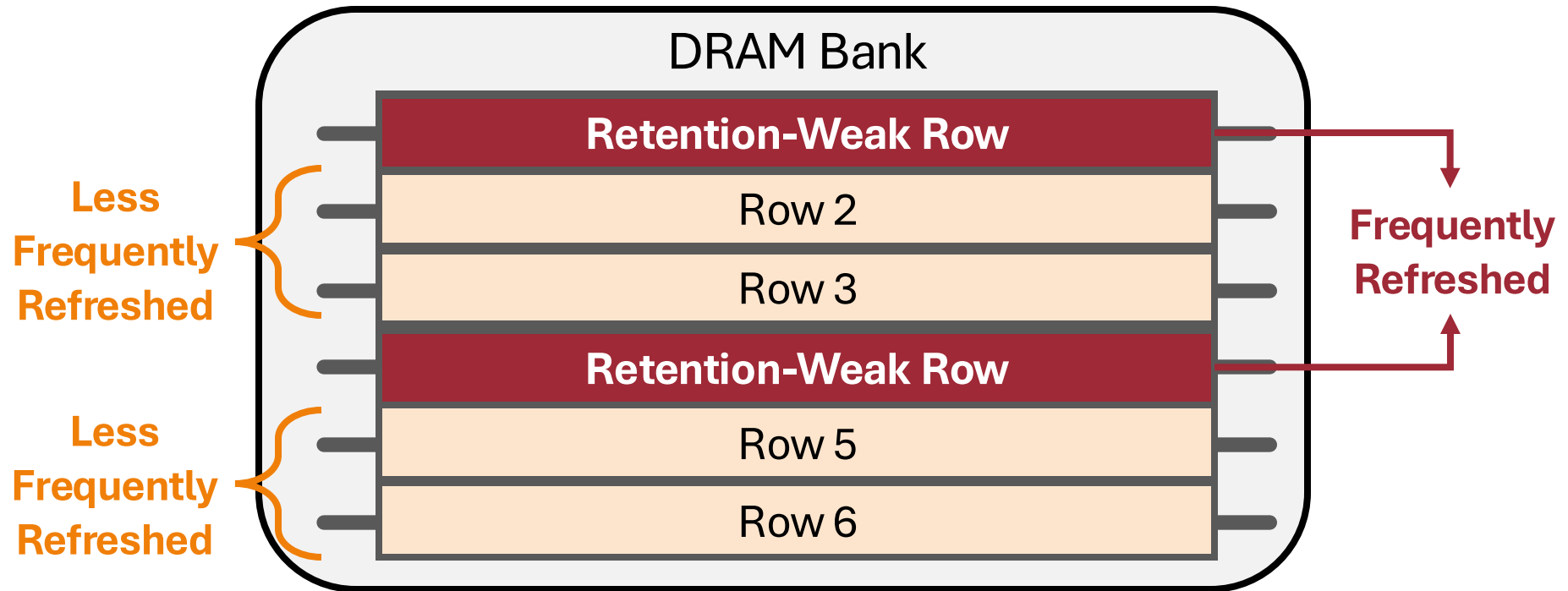
Dynamic random access memory (DRAM) [1] is the dominant main memory technology in nearly all modern computing systems due to its latency and cost characteristics. Continuing to increase DRAM capacity requires increasing the density of DRAM cells by reducing (i.e., scaling) the technology node size of DRAM, which in turn reduces DRAM cell size, cell-to-cell spacing, and cell-to-bitline spacing [2]. Unfortunately, as a result of this aggressive technology node scaling, DRAM suffers from worsening read disturbance problems [3–6].

Read disturbance in modern DRAM chips [4–9] is a widespread phenomenon and is reliably used for breaking memory isolation [5, 7, 9–70], a fundamental building block of robust (i.e., safe, secure, reliable, available) systems. RowHammer and RowPress are two prominent examples of DRAM read disturbance phenomena where a victim DRAM row experiences bitflips when a nearby aggressor DRAM row is 1) repeatedly activated (i.e., hammered) [5–7] or 2) kept open for a long period (i.e., nressed) [4, 71]. Many prior works [3–5, 7, 9, 41, 42]

<https://arxiv.org/pdf/2510.14750>

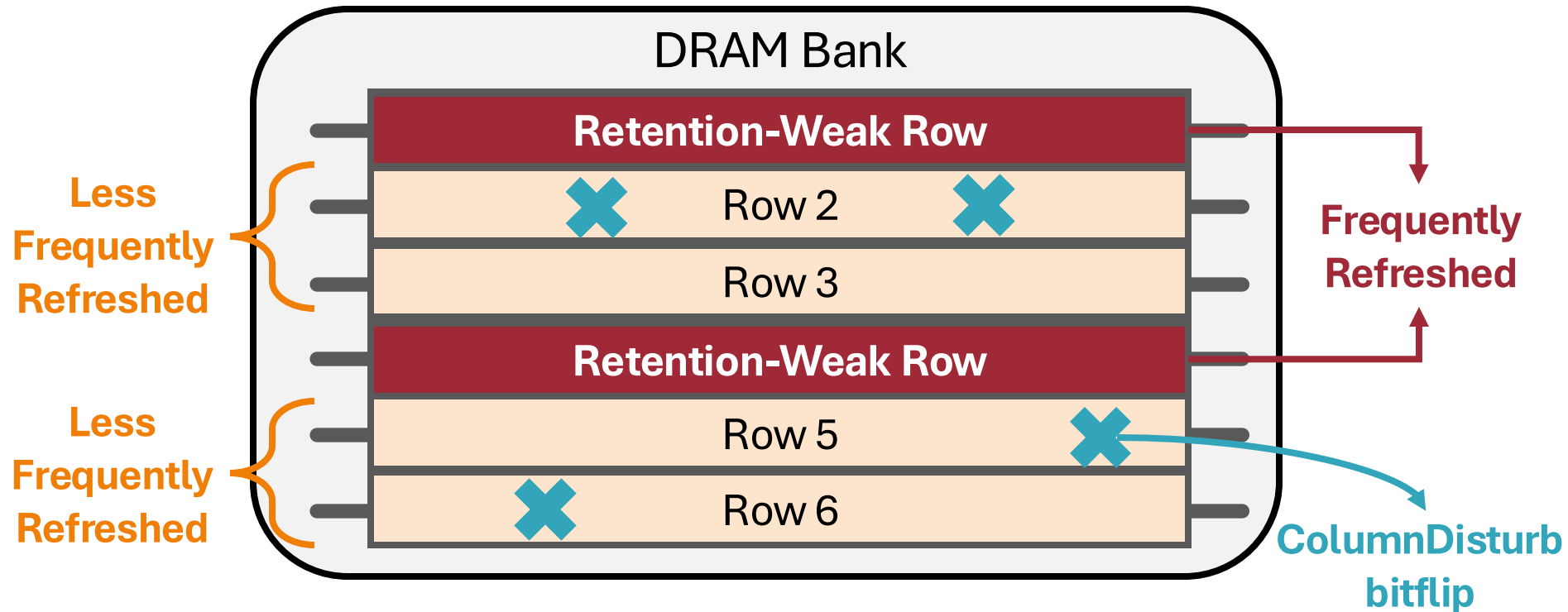
Retention-Aware Refresh Mechanisms

- Leverage the heterogeneity in DRAM cell data retention time
 - Most of the rows **do not need to be frequently refreshed**



Implications on Retention-Aware Refresh Mechanisms

- However, in the presence of **ColumnDisturb**, **less frequently refreshed rows become vulnerable** to ColumnDisturb



Benefits of retention-aware refresh mechanisms significantly reduces in the presence of ColumnDisturb

ColumnDisturb: Understanding Column-based Read Disturbance in Real DRAM Chips and Implications for Future Systems

İsmail Emir Yüksel¹

Ataberk Olgun¹

F. Nisa Bostancı¹

Haocong Luo¹

A. Giray Yağlıkçı^{1,2}

Onur Mutlu¹

¹ETH Zürich

²CISPA

We experimentally demonstrate a new widespread read disturbance phenomenon, ColumnDisturb, in real commodity DRAM chips. By repeatedly opening or keeping a DRAM row (aggressor row) open, we show that it is possible to disturb DRAM cells through a DRAM column (i.e., bitline) and induce bitflips in DRAM cells sharing the same columns as the aggressor row (across multiple DRAM subarrays). With ColumnDisturb, the activation of a single row concurrently disturbs DRAM cells across as many as three DRAM subarrays (e.g., up to 3072 DRAM rows in tested DDR4 DRAM chips) as opposed to RowHammer & RowPress, which affect only a few neighboring rows of the aggressor row in a single subarray. We rigorously and comprehensively characterize ColumnDisturb and its characteristics under various operational conditions (i.e., temperature, data pattern, DRAM timing parameters, average voltage level of the bitline, memory access pattern, and spatial variation) using 216 DDR4 and 4 HBM2 chips from three major DRAM manufacturers. Among our 27 key experimental observations, we highlight two major results and their implications.

1. Introduction

Dynamic random access memory (DRAM) [1] is the dominant main memory technology in nearly all modern computing systems due to its latency and cost characteristics. Continuing to increase DRAM capacity requires increasing the density of DRAM cells by reducing (i.e., scaling) the technology node size of DRAM, which in turn reduces DRAM cell size, cell-to-cell spacing, and cell-to-bitline spacing [2]. Unfortunately, as a result of this aggressive technology node scaling, DRAM suffers from worsening read disturbance problems [3–6].

Read disturbance in modern DRAM chips [4–9] is a widespread phenomenon and is reliably used for breaking memory isolation [5, 7, 9–70], a fundamental building block of robust (i.e., safe, secure, reliable, available) systems. *RowHammer* and *RowPress* are two prominent examples of DRAM read disturbance phenomena where a victim DRAM row experiences bitflips when a nearby aggressor DRAM row is 1) repeatedly activated (i.e., hammered) [5–7] or 2) kept open for a long period (i.e., nressed) [4, 71]. Many prior works [3–5, 7, 9, 41, 42]

<https://arxiv.org/pdf/2510.14750>

More in the Paper

- **Many more characterization results**
 - Average voltage level on the perturbed columns
 - Temperature
 - Spatial variation
 - Access pattern
 - Data pattern
 - Aggressor row on time
- **Details and analyses on Implications**
 - Effectiveness of System-Level and on-Die ECC
 - More details on ColumnDisturb mitigation mechanisms
 - Implications on a retention-aware refresh mechanism: RAIDR
 - How ColumnDisturb affects energy-efficiency and the speedup of such mechanisms
 - ..

ColumnDisturb: Understanding Column-based Read Disturbance in Real DRAM Chips and Implications for Future Systems

İsmail Emir Yüksel¹ Ataberk Olgun¹ F. Nisa Bostancı¹
Haocong Luo¹ A. Giray Yağlıkçı^{1,2} Onur Mutlu¹
¹ETH Zürich ²CISPA

We experimentally demonstrate a new widespread read disturbance phenomenon, ColumnDisturb, in real commodity DRAM chips. By repeatedly opening or keeping a DRAM row (aggressor row) open, we show that it is possible to disturb DRAM cells through a DRAM column (i.e., bitline) and induce bitflips in DRAM cells sharing the same columns as the aggressor row (across multiple DRAM subarrays). With ColumnDisturb, the activation of a single row concurrently disturbs DRAM cells across as many as three DRAM subarrays (e.g., up to 3072 DRAM rows in tested DDR4 DRAM chips) as opposed to RowHammer & RowPress, which affect only a few neighboring rows of the aggressor row in a single subarray. We rigorously and comprehensively characterize ColumnDisturb and its characteristics under various operational conditions (i.e., temperature, data pattern, DRAM timing parameters, average voltage level of the bitline, memory access pattern, and spatial variation) using 216 DDR4 and 4 HBM2 chips from three major DRAM manufacturers. Among our 27 key experimental observations, we highlight two major results and their implications.

1. Introduction

Dynamic random access memory (DRAM) [1] is the dominant main memory technology in nearly all modern computing systems due to its latency and cost characteristics. Continuing to increase DRAM capacity requires increasing the density of DRAM cells by reducing (i.e., scaling) the technology node size of DRAM, which in turn reduces DRAM cell size, cell-to-cell spacing, and cell-to-bitline spacing [2]. Unfortunately, as a result of this aggressive technology node scaling, DRAM suffers from worsening read disturbance problems [3–6].

Read disturbance in modern DRAM chips [4–9] is a widespread phenomenon and is reliably used for breaking memory isolation [5, 7, 9–70], a fundamental building block of robust (i.e., safe, secure, reliable, available) systems. *RowHammer* and *RowPress* are two prominent examples of DRAM read disturbance phenomena where a victim DRAM row experiences bitflips when a nearby aggressor DRAM row is 1) repeatedly activated (i.e., hammered) [5–7] or 2) kept open for a long period (i.e., pressed) [4, 71]. Many prior works [3–5, 7, 9, 41, 42]

<https://arxiv.org/pdf/2510.14750>

Outline

What is ColumnDisturb?

Testing Infrastructure

Real DRAM Chip Results

Implications for Future Systems & Mechanisms

Conclusion

Conclusion

We demonstrate and extensively study a new widespread read disturbance phenomenon in real DRAM chips: ColumnDisturb

We characterize 216 DDR4 and 4 HBM2 chips from three major manufacturers

- ColumnDisturb affects **all tested chips**
- ColumnDisturb is **getting worse** as DRAM technology node size **reduces**
- ColumnDisturb is **different** from **RowHammer, RowPress, and Retention** failures

We describe and evaluate two major implications of ColumnDisturb

- ColumnDisturb affects **robustness of future systems**
- ColumnDisturb makes **retention-aware refresh mechanisms much less effective**

We hope ColumnDisturb inspires system-level, architectural-level, and device-level solutions and understanding to enable ColumnDisturb-resilient future systems

ColumnDisturb

Understanding Column-based Read Disturbance in Real DRAM Chips and Implications for Future Systems

İsmail Emir Yüksel

Ataberk Olgun F. Nisa Bostancı Haocong Luo

A. Giray Yağlıkçı Onur Mutlu

ColumnDisturb

Understanding Column-based Read Disturbance in Real DRAM Chips and Implications for Future Systems

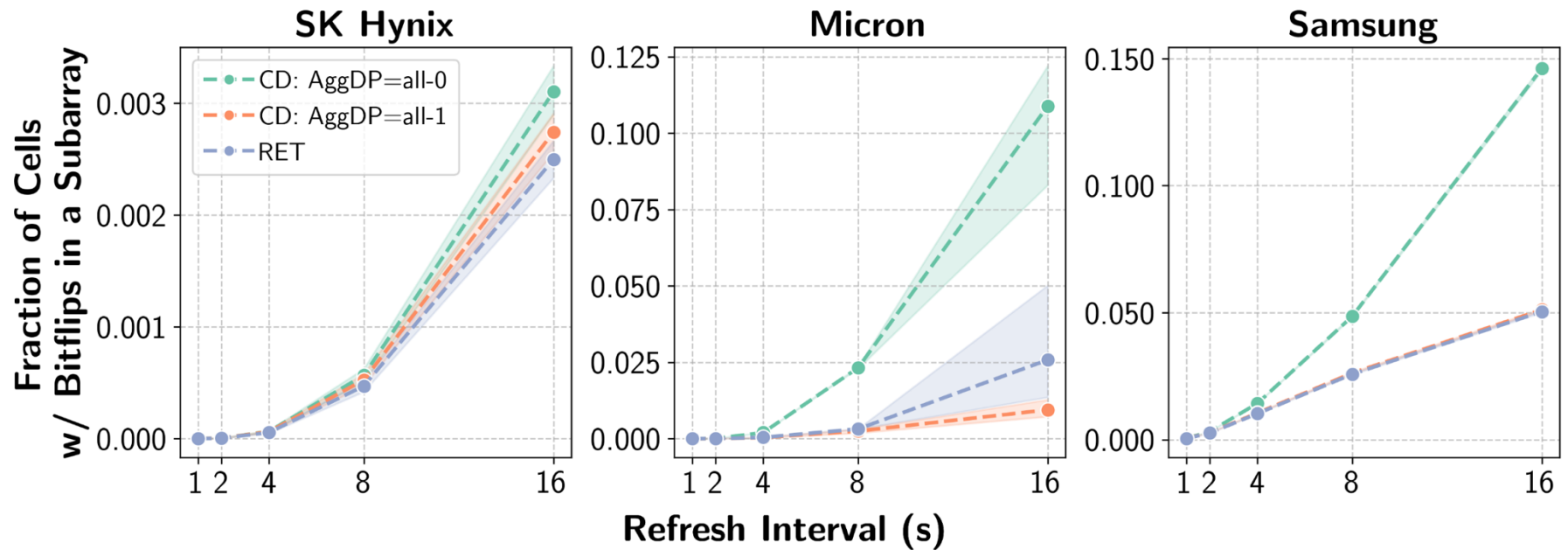
Backup Slides

İsmail Emir Yüksel

Ataberk Olgun F. Nisa Bostancı Haocong Luo

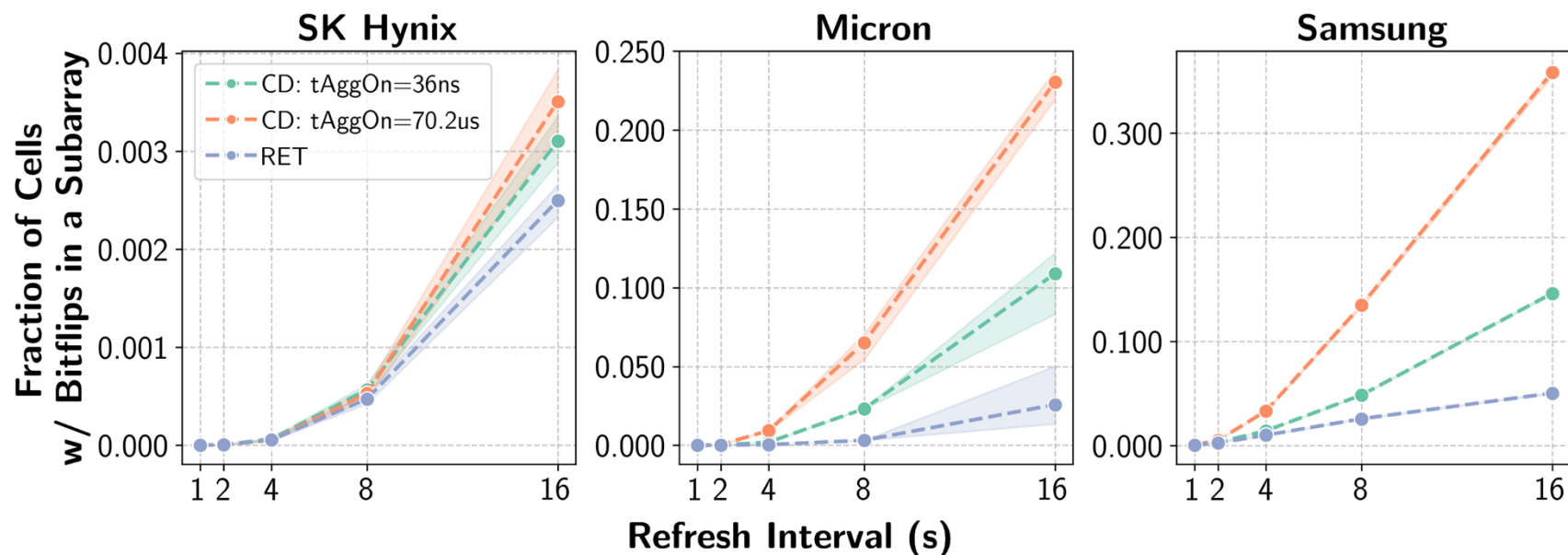
A. Giray Yağlıkçı Onur Mutlu

Foundational Results: Data Pattern



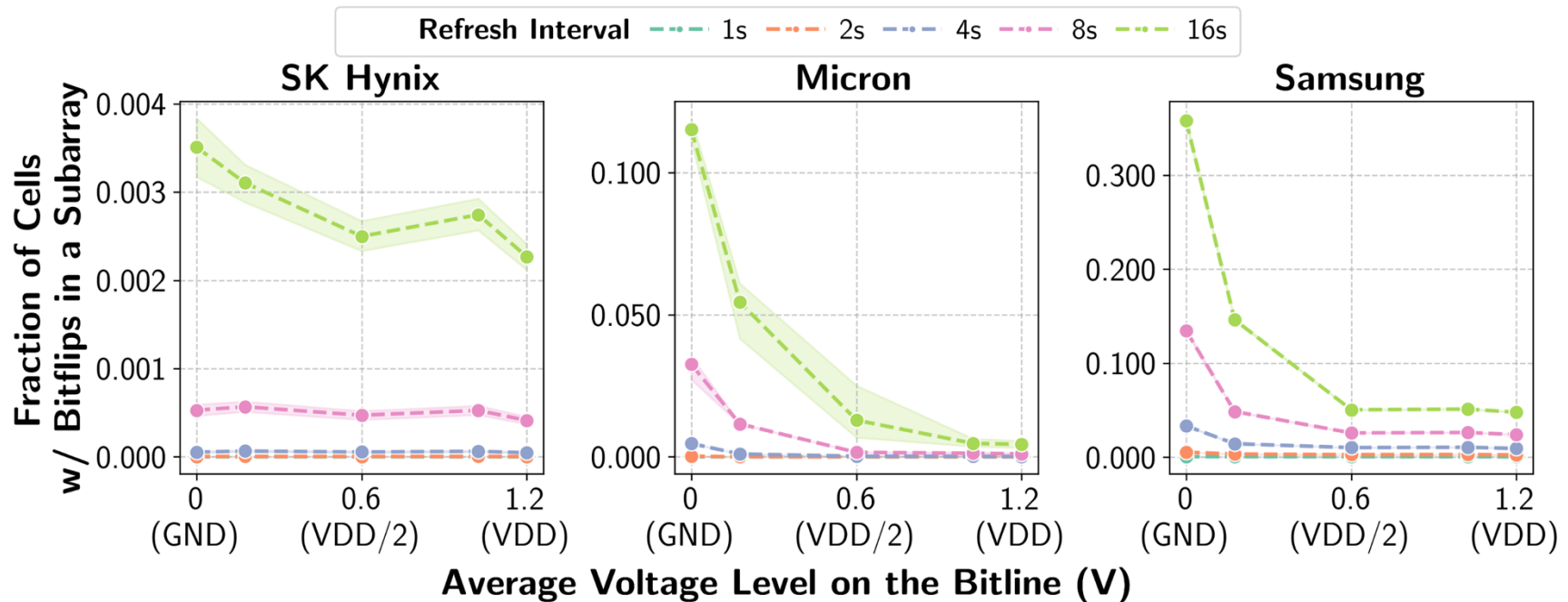
ColumnDisturb induces many more bitflips when the aggressor row data pattern is all-0 versus all-1

Foundational Results: tAggOn



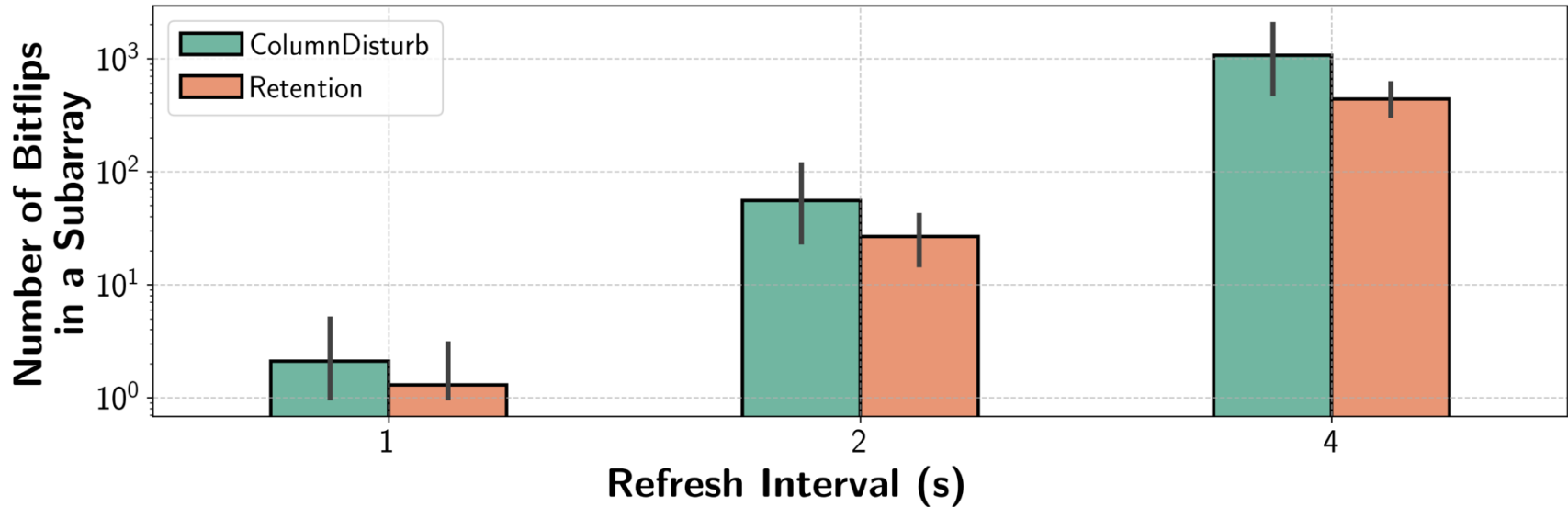
Fraction of cells with ColumnDisturb bitflips in a subarray significantly increases as t_{AggOn} increases

Foundational Results: $\text{AVG}(V_{BL})$



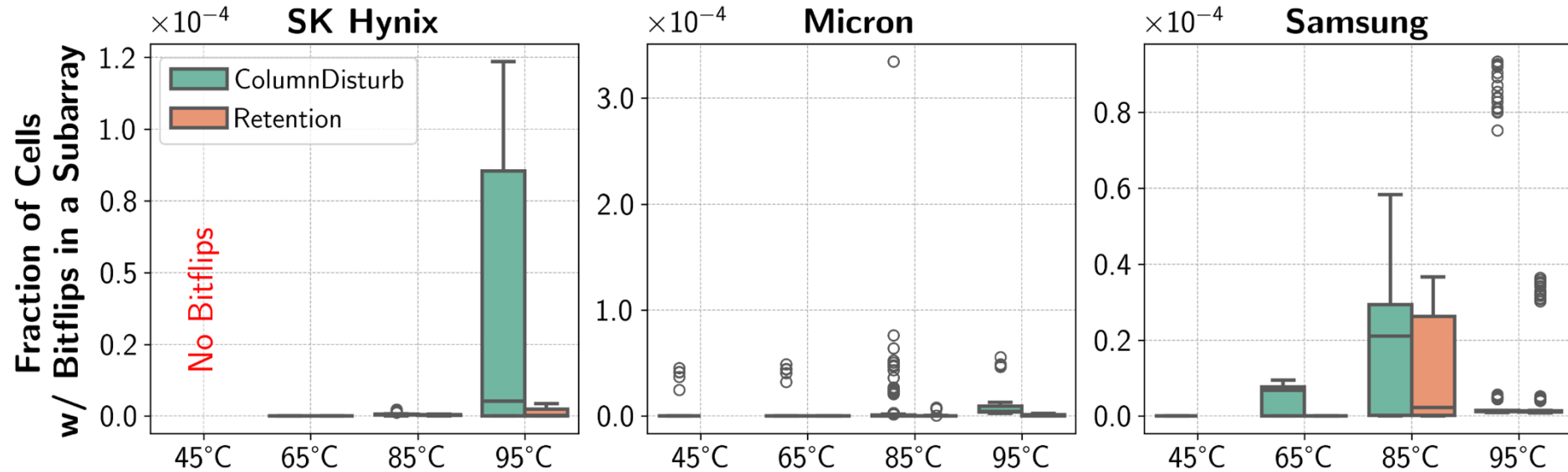
A DRAM chip becomes significantly more vulnerable to ColumnDisturb as the average voltage level on the perturbed columns decreases

Foundational Results: CD on HBM



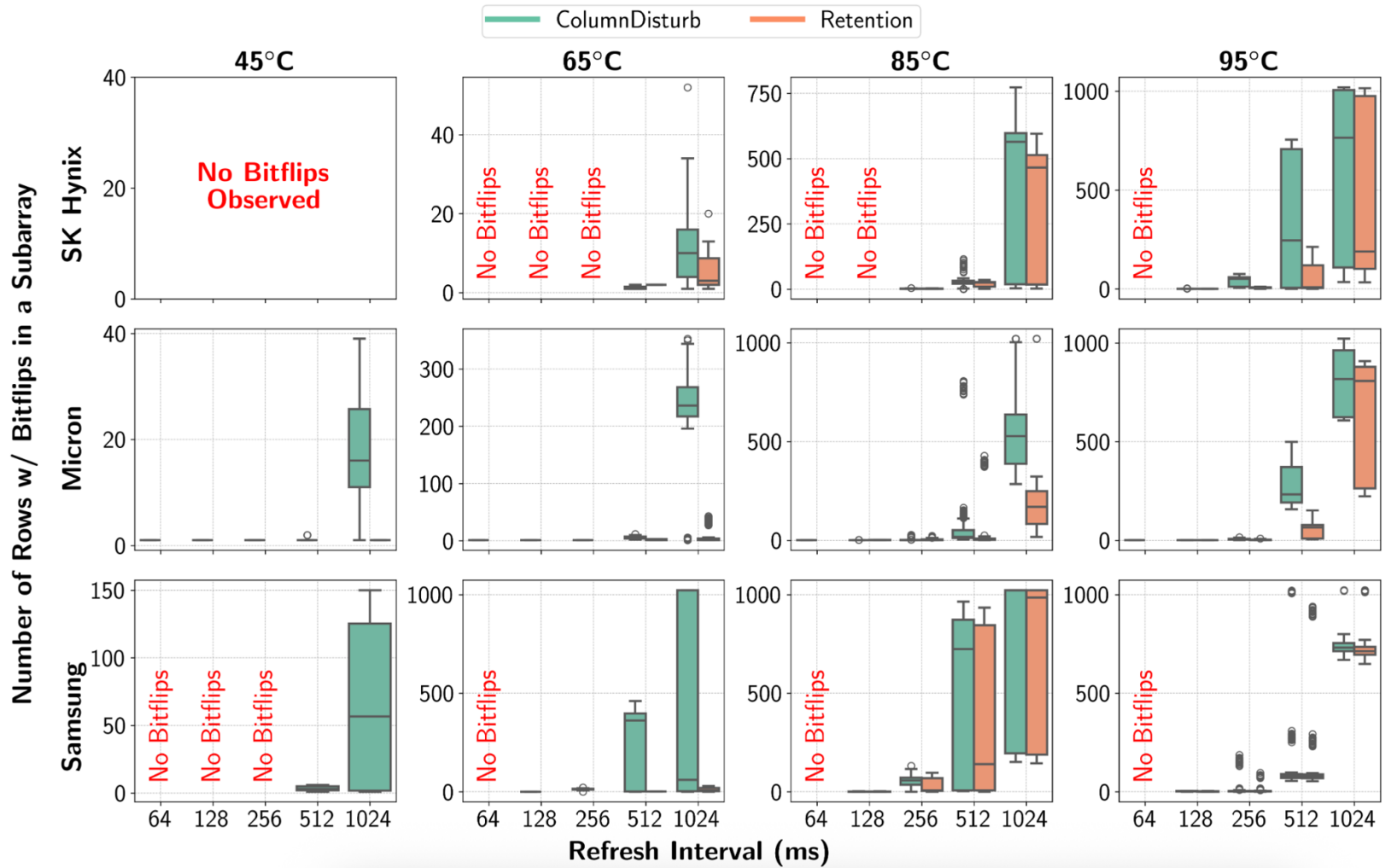
HBM2 chips are vulnerable to ColumnDisturb.
ColumnDisturb induces many more bitflips than retention failures

Temperature: Fraction of Cells w/ Bitflips

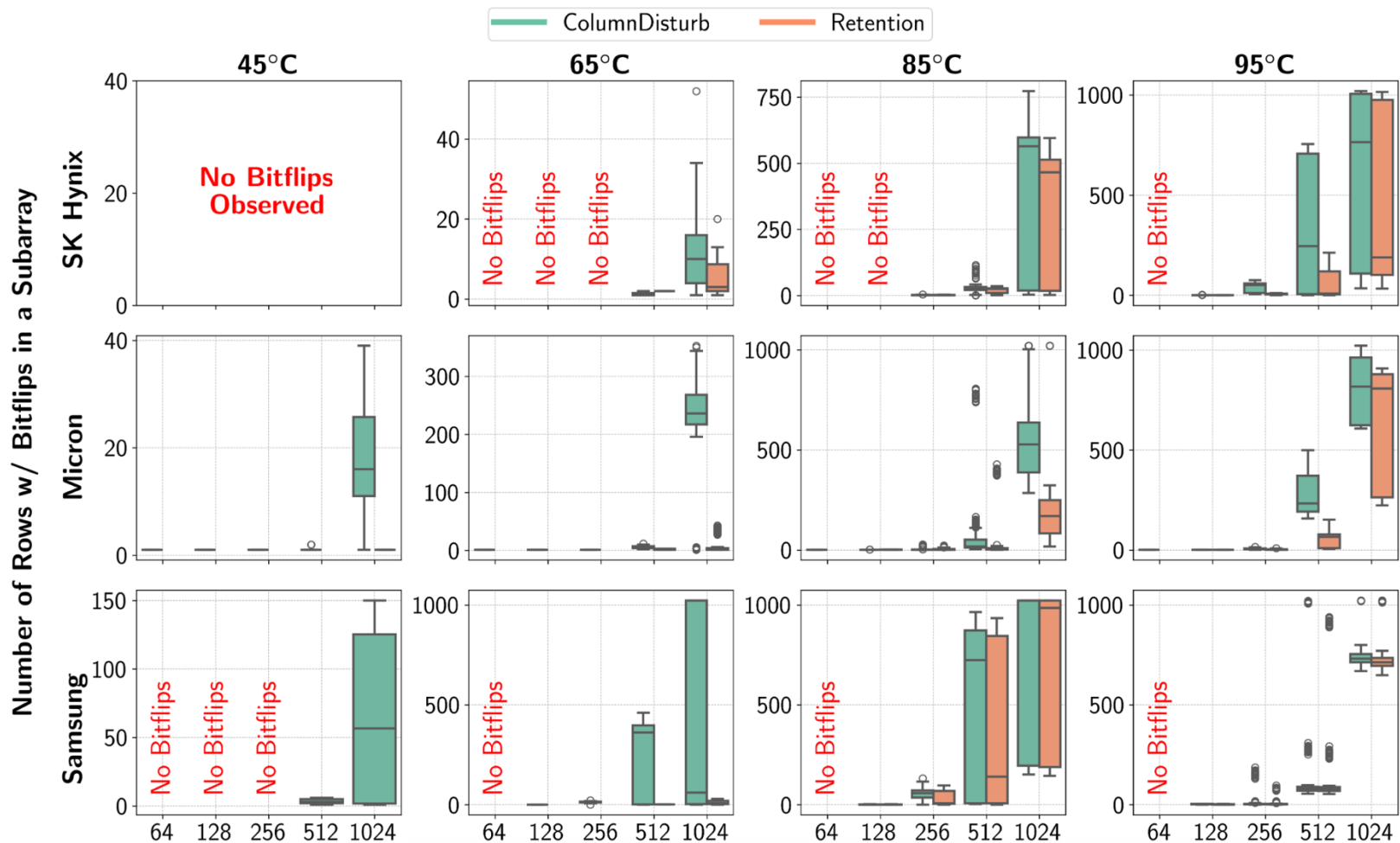


ColumnDisturb is more sensitive to temperature than retention failures

Temperature: Number of Rows w/ Bitflips

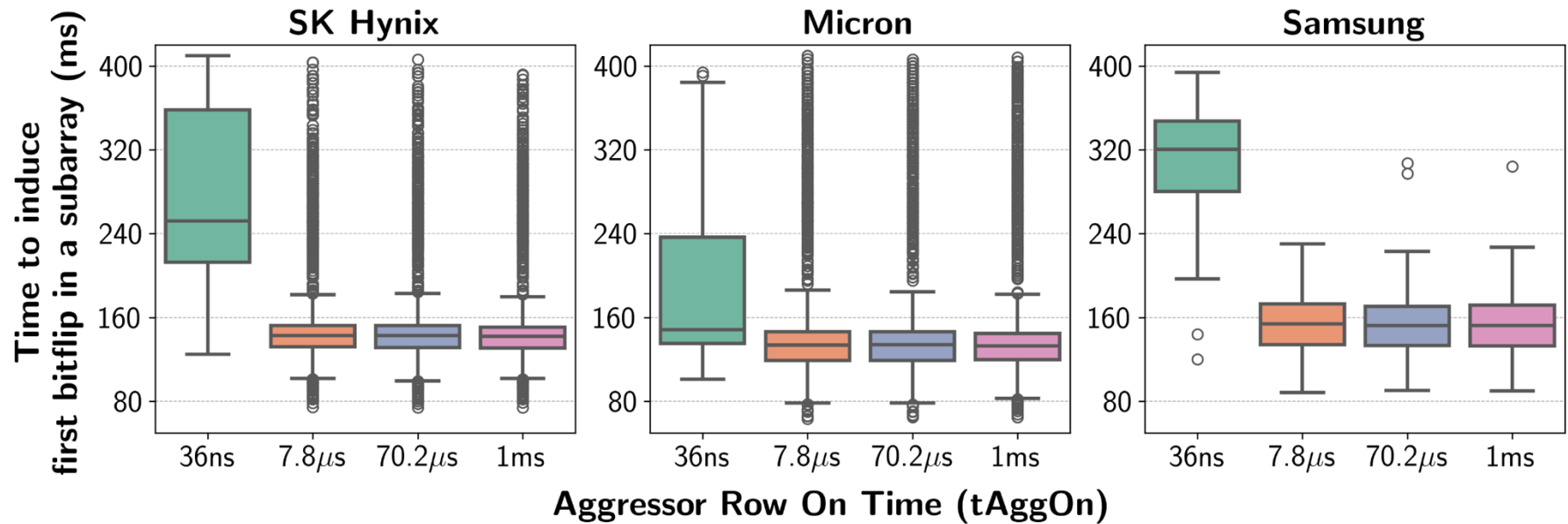


Temperature: Number of Rows w/ Bitflips



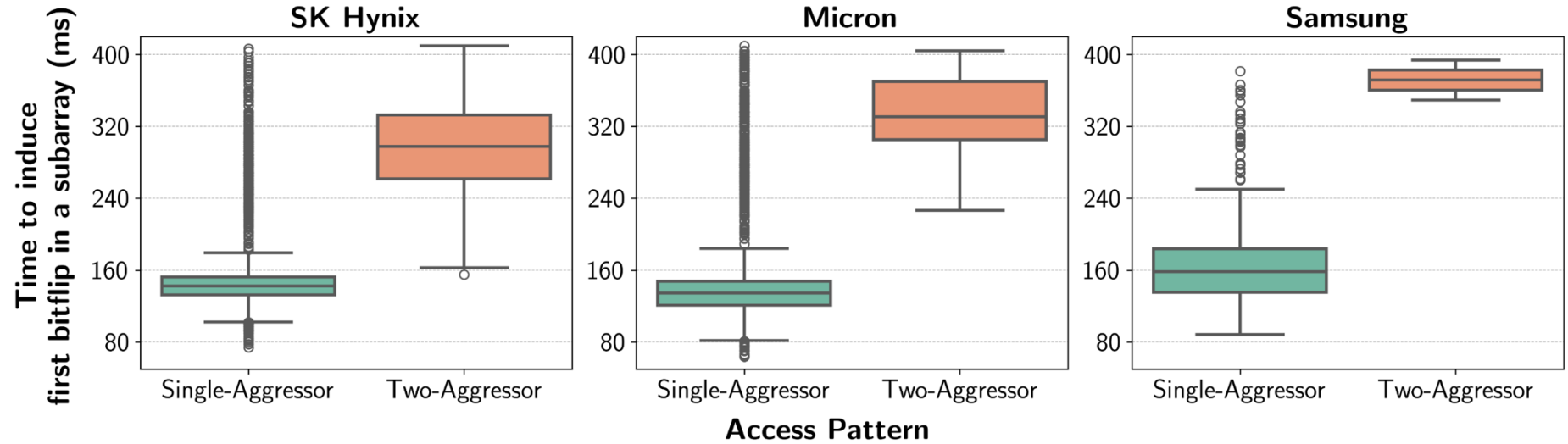
The blast radius of both ColumnDisturb and retention failures increase with higher temperature

In-Depth: Aggressor Row On Time



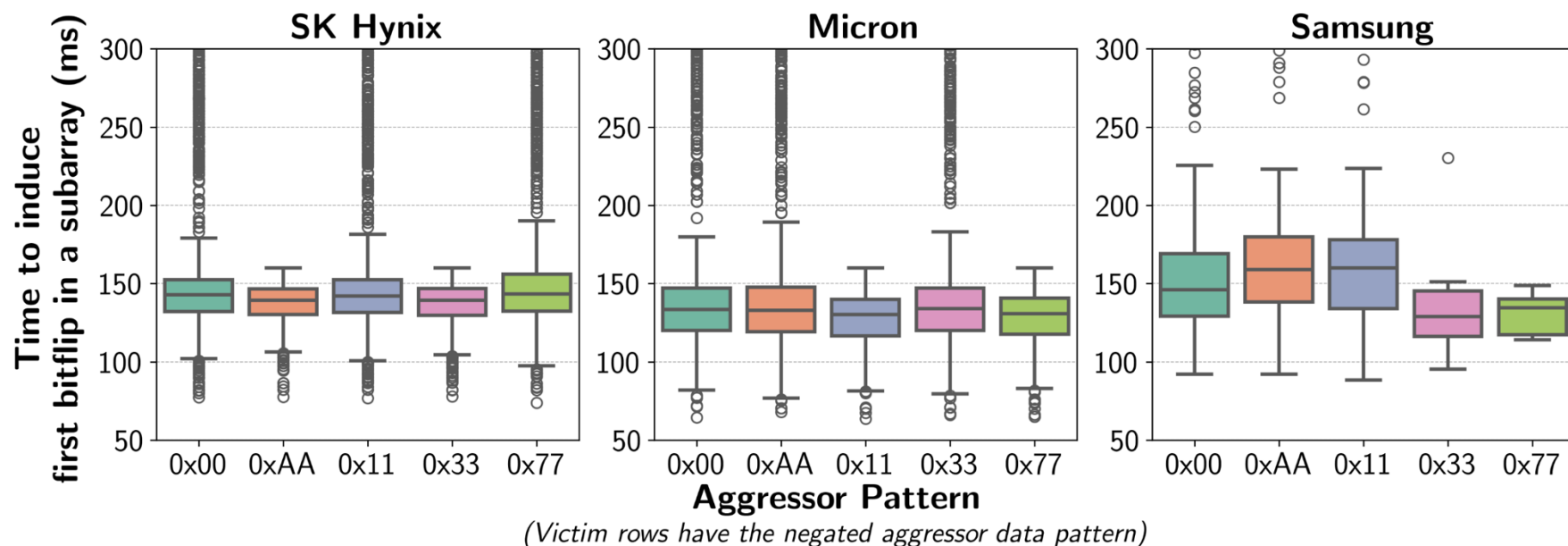
Keeping a row open ($t_{AggOn} \gg t_{RAS}$) is significantly more effective than immediately closing it ($t_{AggOn} = t_{RAS}$) in inducing the first ColumnDisturb bitflip in a subarray

In-Depth: Access Pattern



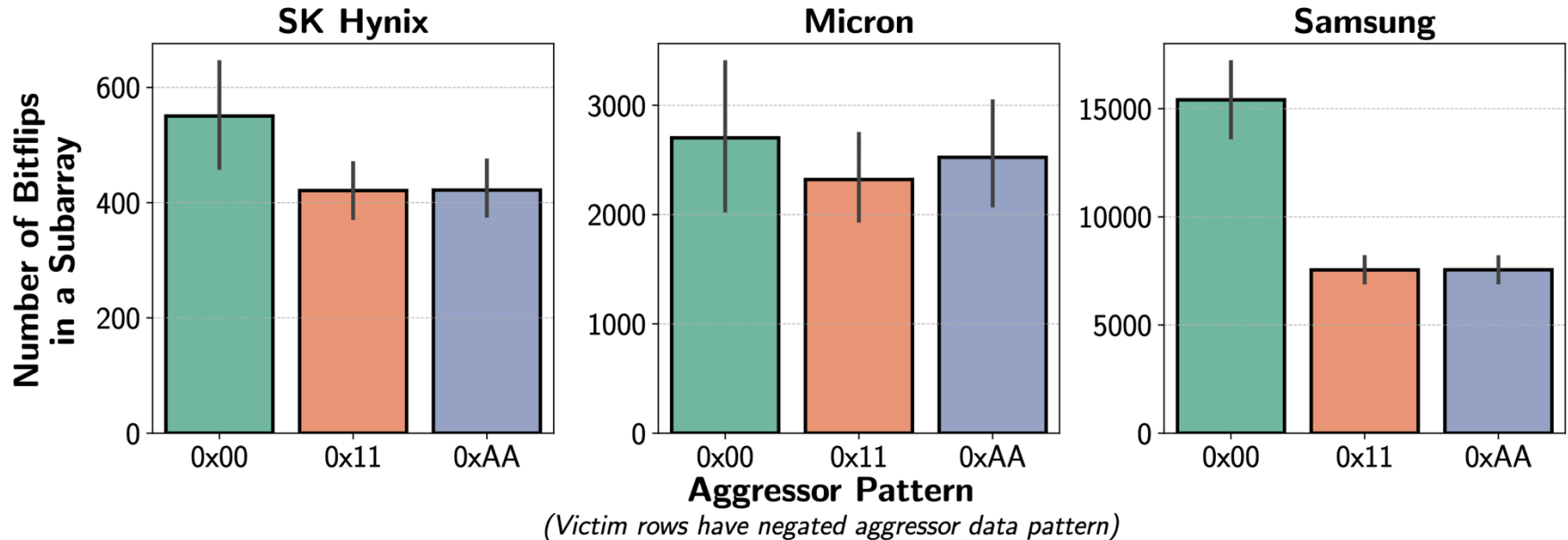
**Single-aggressor access pattern
induces the first ColumnDisturb bitflip in a subarray
significantly faster than the two-aggressor access pattern**

In-Depth: Data Pattern: TTIFB



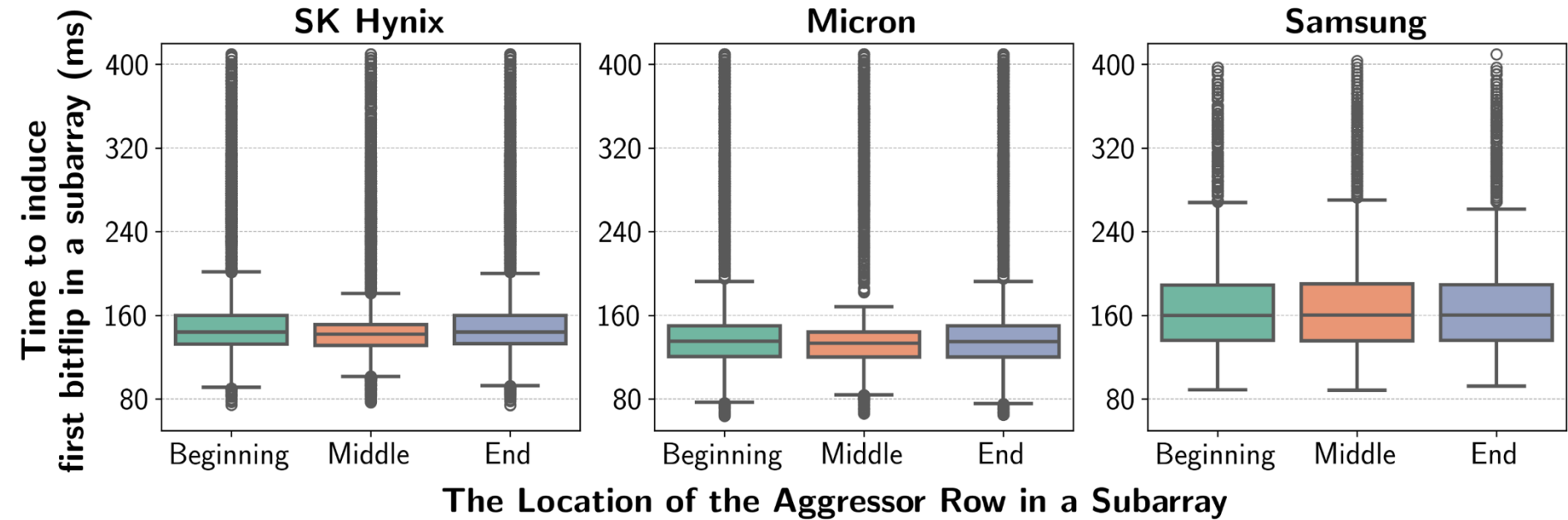
Data pattern has a small effect on the time to induce the first ColumnDisturb bitflip in a subarray

In-Depth: Data Pattern: # of Bitflips



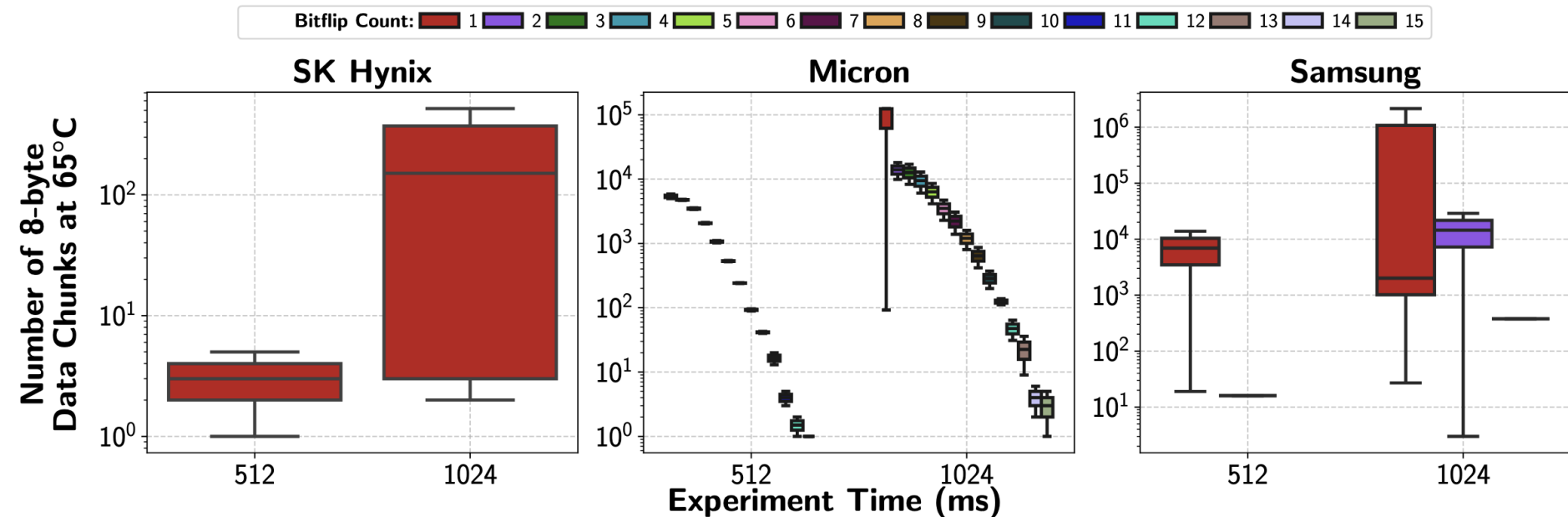
Data pattern greatly affects total ColumnDisturb:
having more logic-0 values
In the perturbed columns generates more bitflips

Aggressor Row Location in a Subarray



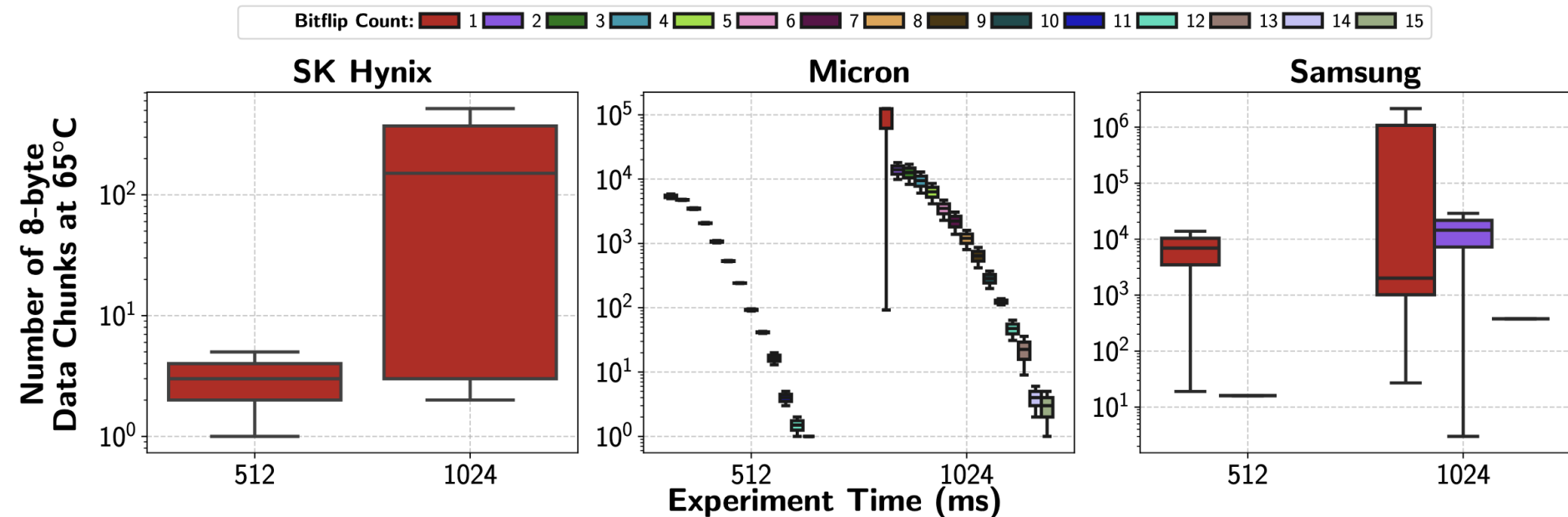
The location of the aggressor row
only slightly affects
the time to induce the first ColumnDisturb bitflip

Effectiveness of ECC



Many 8-byte data chunks experience
3 (up to 15) ColumnDisturb bitflips,
which the SECDED ECC cannot correct or detect

Effectiveness of ECC



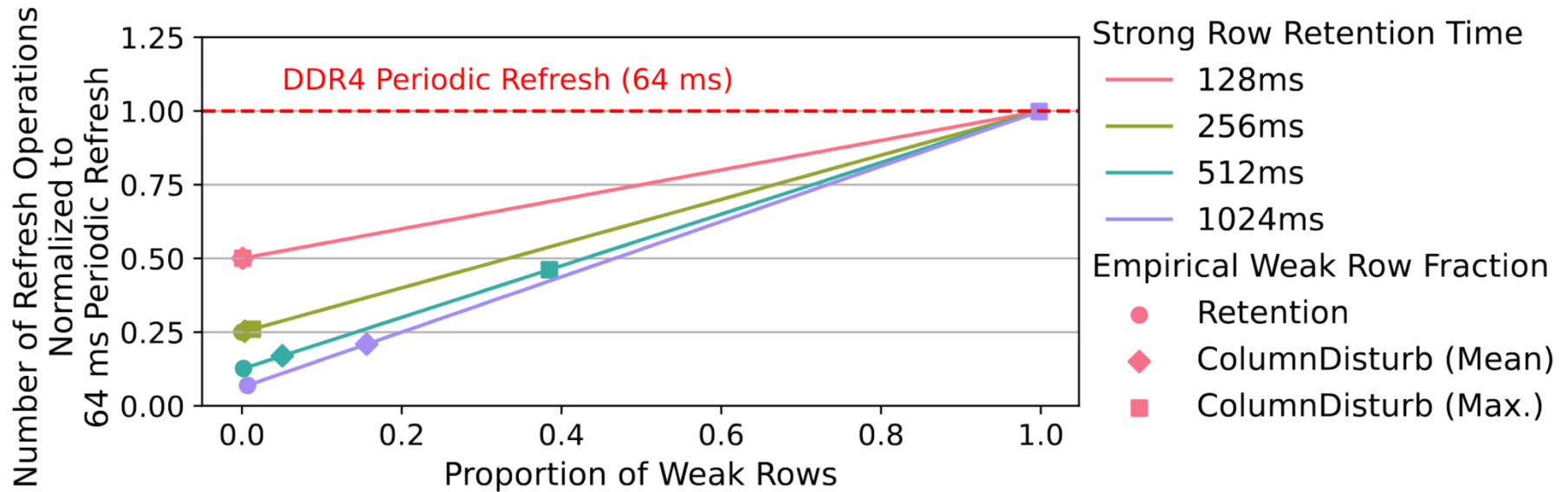
Since ColumnDisturb can induce 15 bitflips in multiple 8-byte data chunks, a (7,4) Hamming code could correct such bitflips with very large DRAM storage overheads (75%, i.e., three parity bits for every four data bits)

Implications for System Robustness

Increasing DRAM Refresh Rate

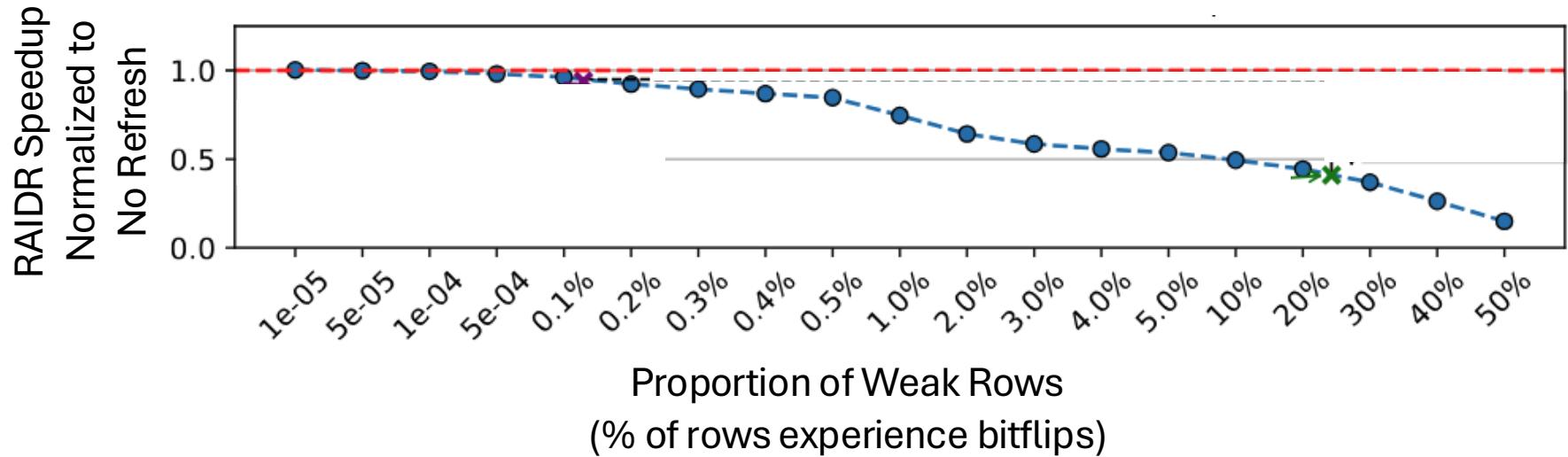
- **Increasing DRAM refresh rate** in future DRAM chips to prevent ColumnDisturb-induced bitflips
- DRAM chips perform periodic refreshes to avoid retention failures
 - E.g., Each row in DDR5 chips is refreshed **in every 32ms**
- The benefit
 - It can be applied **immediately in today's systems**
- Downside
 - It **degrades** system performance and energy efficiency **drastically**
- For a 32Gb DDR5 chip, 4x higher refresh rate
 - **Reduces system throughput** by 42.1%
 - **Increases energy consumption** by 67.5%

Number of Refresh Operations Needed



**In the presence of ColumnDisturb,
for a strong row retention time of 1024 ms,
the number of refresh operations increases by 3.02x on average**

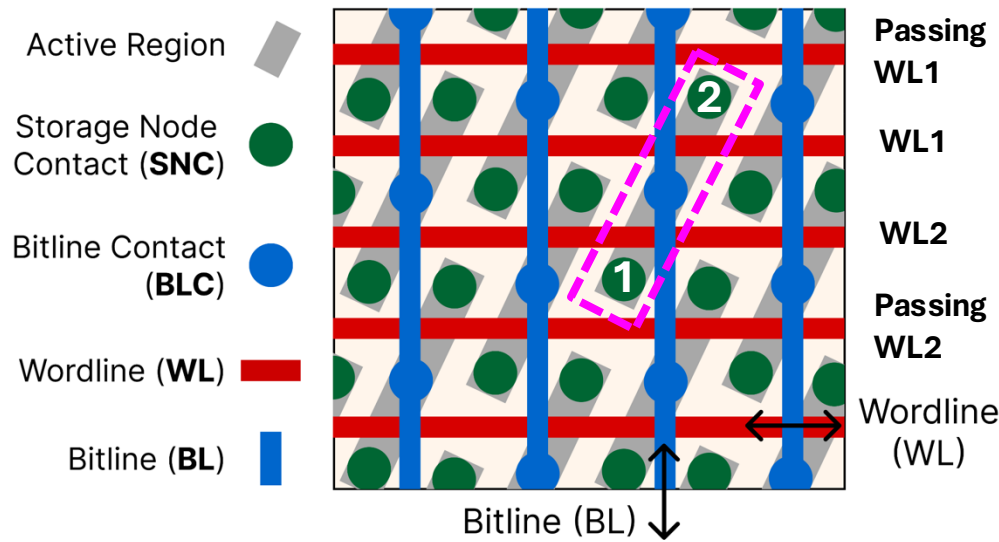
Implications on Retention-Aware Refresh



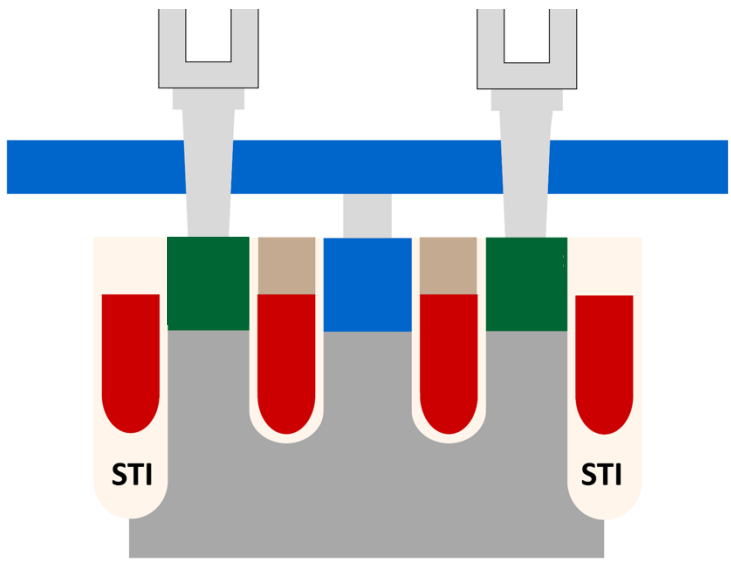
The speedup benefits of RAIDR reduce significantly in the presence of ColumnDisturb
compared to a baseline where there is no ColumnDisturb bitflips

Device-Level Mechanisms – Physical Layout

■ Modern 6F² DRAM cell array layout



a) Physical Layout of 6F² DRAM (Top View)

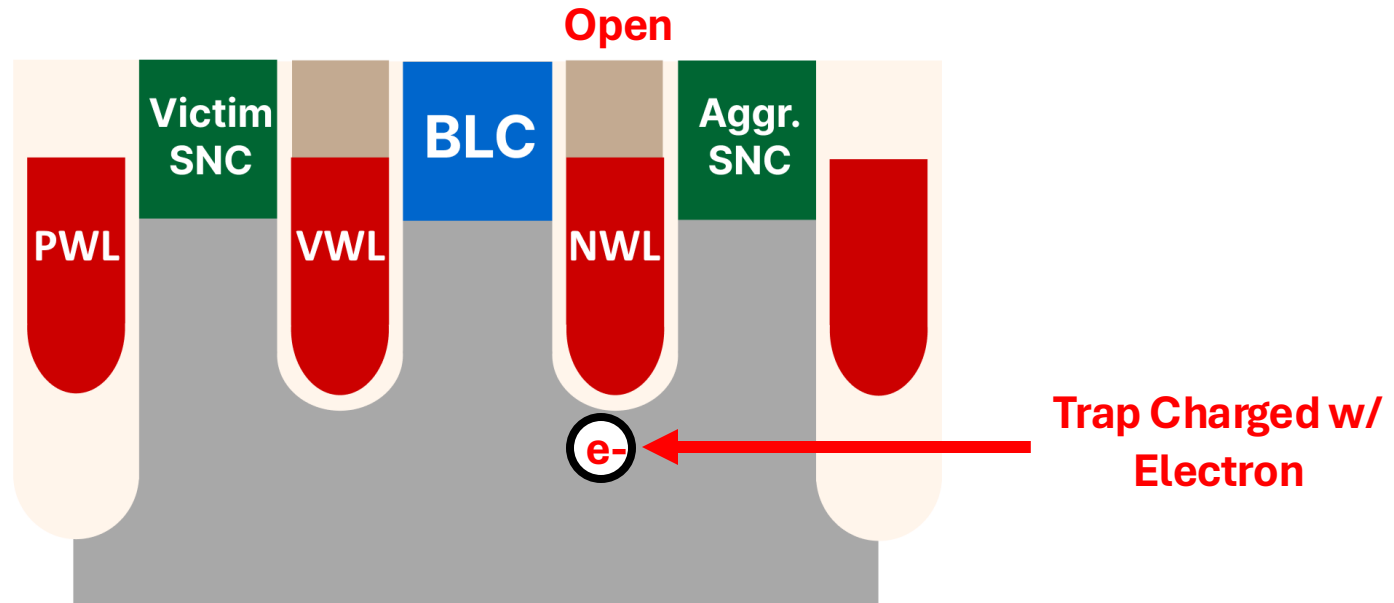


b) Cross-section of an Active Region (Side View, 2 Cells)

Device-Level Mechanism – RowHammer

■ Key Error Mechanisms of RowHammer

- Trap-assisted Electron Migration [Yang+, EDL'19] [Walker+, TED'21] [Zhou+, IRPS'23]
 1. When NWL (aggressor) is open, acceptor-like traps are charged with electrons

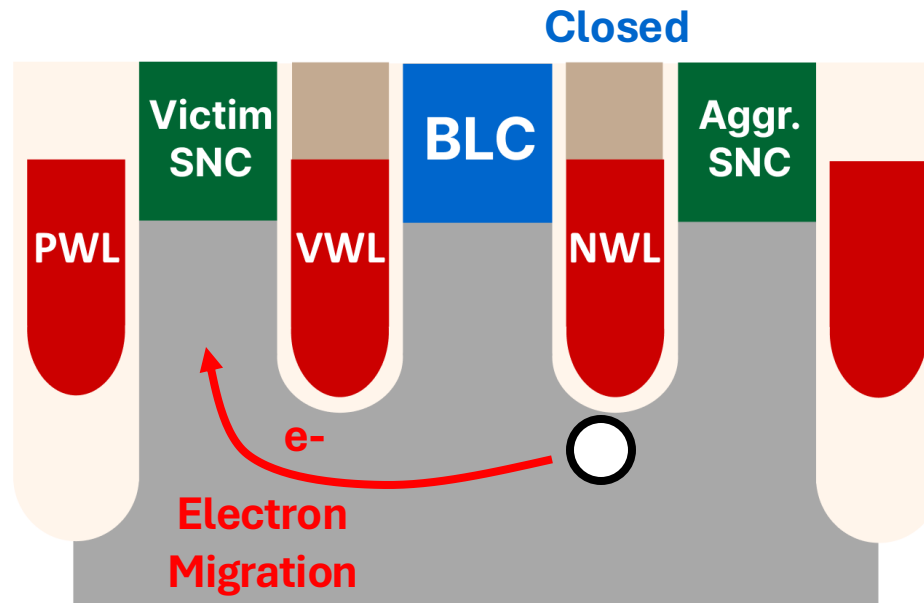


PWL: Passing Wordline **VWL:** Victim Wordline
NWL: Neighboring Wordline (Aggressor)

Device-Level Mechanism – RowHammer

■ Key Error Mechanisms of RowHammer

- ❑ Trap-assisted Electron Migration [Yang+, EDL'19] [Walker+, TED'21] [Zhou+, IRPS'23]
 1. When NWL (aggressor) is open, acceptor-like traps are charged with electrons
 2. When NWL (aggressor) is closed, electrons are emitted from traps and migrate towards the victim cell

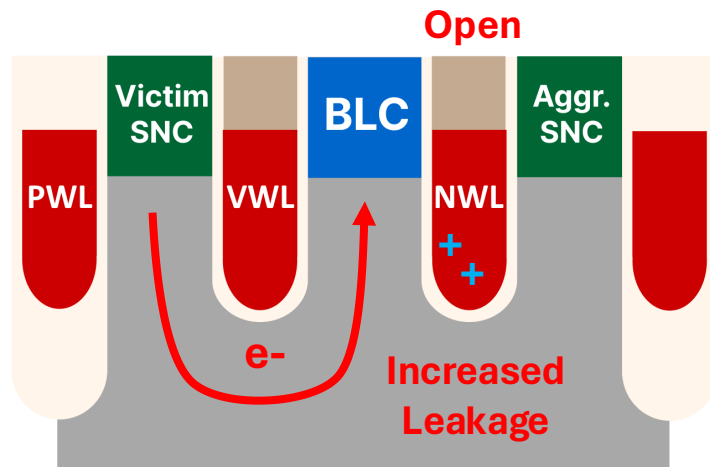


PWL: Passing Wordline **VWL:** Victim Wordline
NWL: Neighboring Wordline (Aggressor)

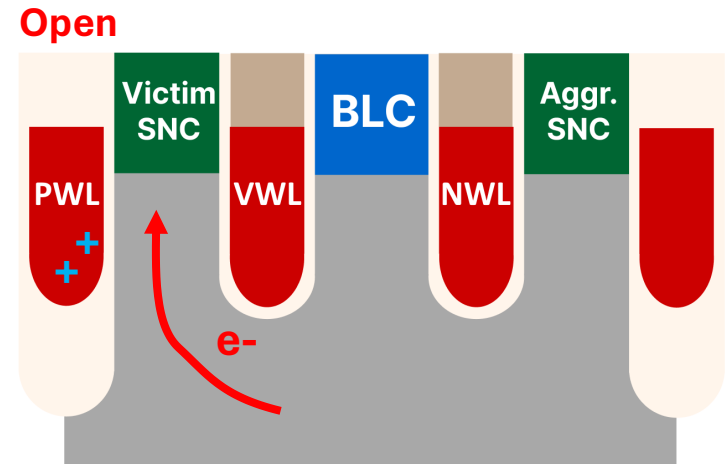
Device-Level Mechanism – RowPress

■ Key Error Mechanisms of RowPress

- ❑ **NWL RowPress:** When NWL is kept open for a long period, its strong electric field increases the leakage from the victim to the BLC, **causing 0→1 bitflips** [Zhou+, TED'24] [Zhou+, IRPS'24]
- ❑ **PWL RowPress:** When PWL is kept open for a long period, its strong electric field draws electrons towards the victim, **causing 1→0 bitflips** [Zhou+, TED'24] [Zhou+, IRPS'24]



NWL RowPress



PWL RowPress

Device-Level ColumnDisturb Mechanism

- ColumnDisturb happens due to columns are perturbed and then driven. There is a direct contact between the cell and the column
- This contact spans and affects thousands of rows
- We have preliminary hypotheses on what might be happening
 - *“ColumnDisturb exacerbates subthreshold leakage of the access transistor [112] and/or the dielectric leakage between the capacitor and the bitline [2].”*

**We need device-level studies
to fundamentally understand ColumnDisturb, and
our study can inspire such future works and be a good starting point**

Access Pattern Difference

- For **RowHammer** and **RowPress**,
 - one must need to hammer/press **the same aggressor row(s)** to observe bitflips.
- However, **for ColumnDisturb**,
 - one can **distribute the number of activations** across many rows
 - As long as **the activated rows perturbs the same set of columns**, **it does not matter** whether **one aggressor row or thousands of aggressor rows are used**
- For example (assume one cell requires 3K activations for a bitflip)
 - **Activating a single row 3K times = activating 1K rows 3 times**
 - It is not easy to *detect and refresh cells*