


DRAMScope: Uncovering DRAM Microarchitecture and Characteristics by Issuing Memory Commands

Hwayong Nam†, Seungmin Baek†, Minbok Wi†, Michael Jaemin Kim†, Jaehyun Park†, Chihun Song‡, Nam Sung Kim‡, Jung Ho Ahn†

† Seoul National University, ‡ University of Illinois Urbana Champaign

Presenter: Hwayong Nam (nhy4916@snu.ac.kr)

Overview

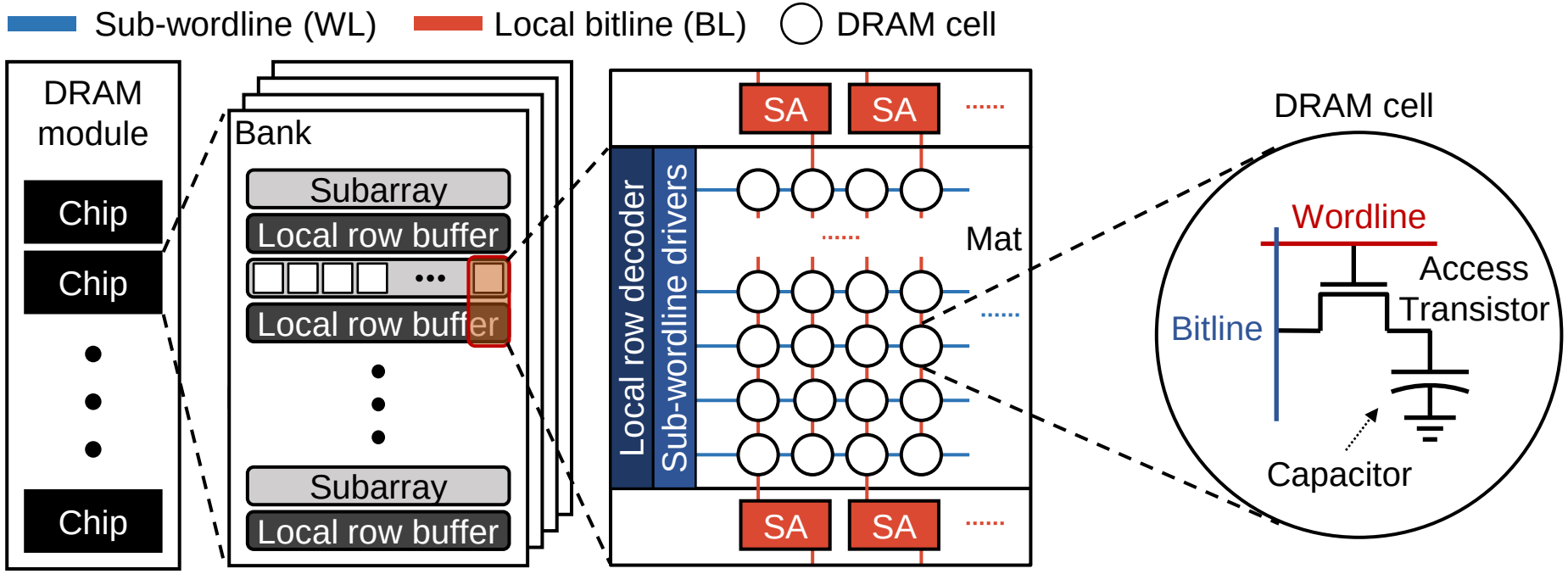
- We reverse-engineer and analyze the **DRAM microarchitecture** and **the characteristics of activate-induced bitflips** in commodity DRAM chips

- We conduct a ***macroscopic analysis*** and identify **the previously unreported structural observations** at the subarray, MAT, and row levels

- We identify **the characteristics of activate-induced bitflip** through our ***microscopic analysis***

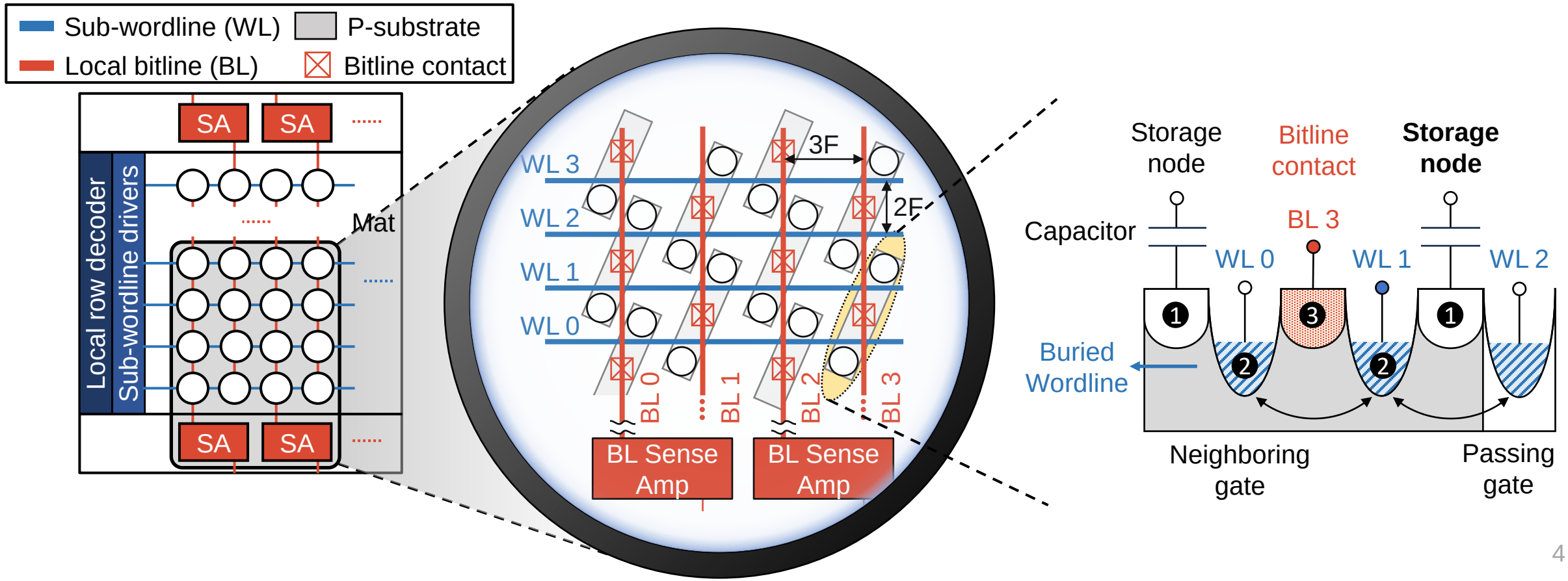
DRAM Organization

- DRAM organization from a ***macroscopic perspective***
  - DRAM has **a hierarchical structure** consisting of banks, subarrays, mats, rows, and columns
  - A DRAM cell is made up of **1 access transistor and 1 capacitor**
  - Modern DRAM chips have an **open bitline structure**



DRAM Organization

- DRAM organization from a ***microscopic perspective***.
  - Modern DRAM chips are primarily designed using a **$6F^2$ cell structure** for higher cell density
  - In $6F^2$ cell structure, the DRAM cells have **different relationships** to two adjacent WLs



Reverse-engineering Techniques

- We use **three techniques**, which are **existing techniques**, to analyze the structure and characteristics of DRAM through **non-destructive testing**

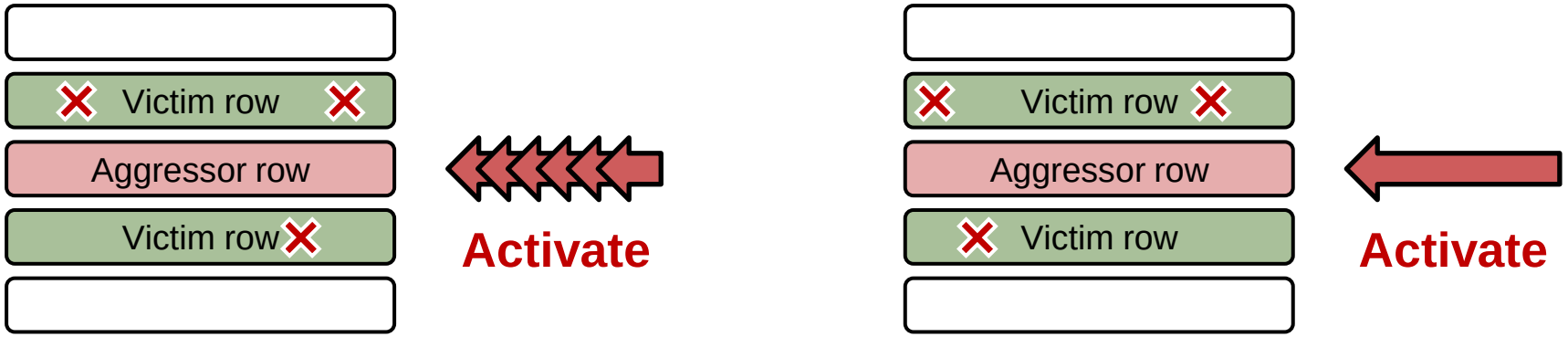
1) **Activate-induced bitflips (AIBs)**

2) **In-DRAM RowCopy**

3) **Retention time test**

Reverse-engineering Techniques – (1)

- **Activate-induced bitflips (AIBs)**
  - Activation disturbs cells in **adjacent rows** and flips the states of the cells
  - **Two attack methods**
    - = **RowHammer: Frequently** repeating *Activate* and *Precharge* on a a single row
    - = **RowPress**[1] **:** Keeping a single row activated ***for a long time***



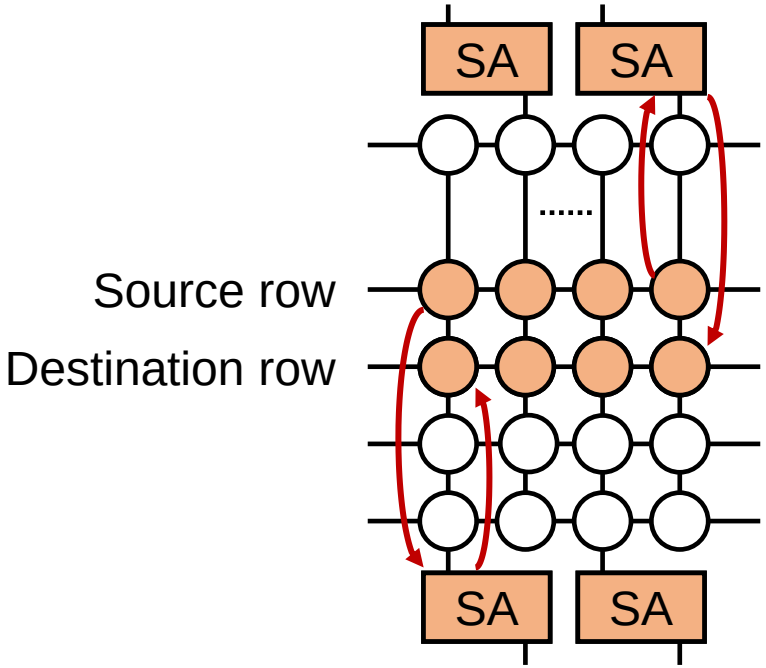
< RowHammer attack >

< RowPress attack >

[1] Luo, Haocong, et al., "Rowpress: Amplifying Read Disturbance in Modern DRAM Chips." *ISCA,* 2023.

Reverse-engineering Techniques – (2)

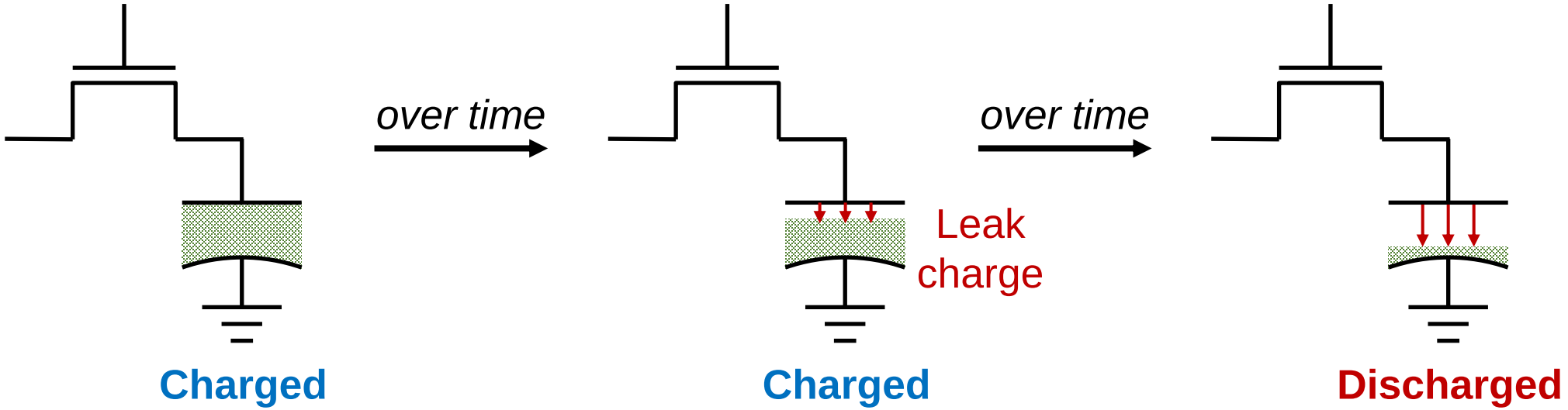
- **RowCopy**
  - An out-of-specification in-memory operation (proposed by *RowClone*[1], implemented by *ComputeDRAM*[2])
  - Copying the value of one row to another row **within the same subarray** using charge-sharing



[1] V. Seshadri et al., “RowClone: Fast and Energy-Efficient In-DRAM Bulk Data Copy and Initialization,” *MICRO*, 2013
[2] Gao, Fei et al., "ComputeDRAM: In-Memory Compute Using Off-the-Shelf DRAMs," *MICRO,* 2019.

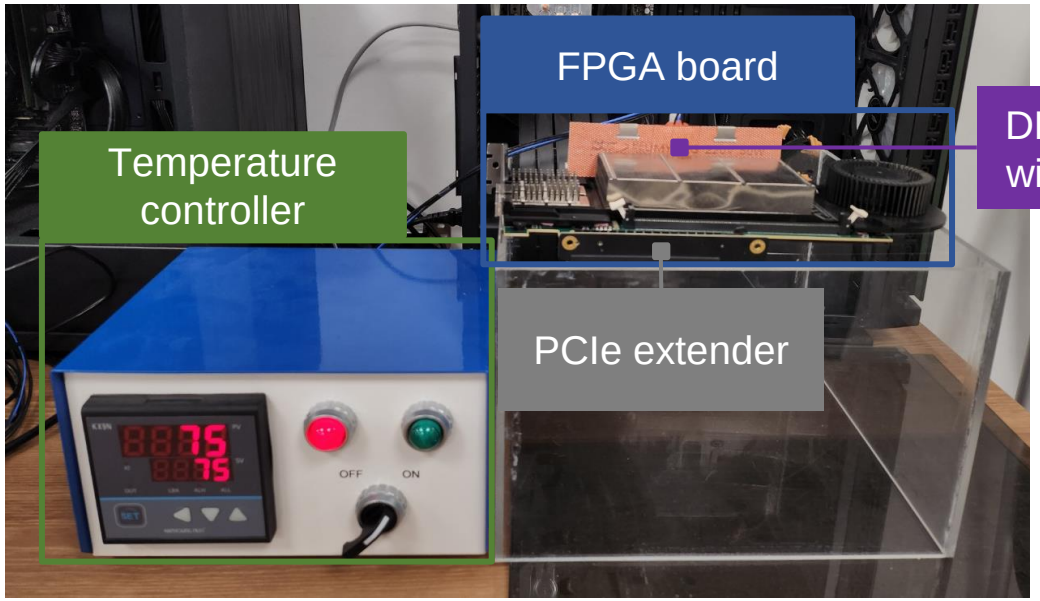
Reverse-engineering Techniques – (3)

- **Retention time test**
  - A DRAM cell **loses the charge** over time
  - Exploiting the fact that leakage occurs from a charged state to a discharged state, we perform a retention time test to distinguish between **true-cells** and **anti-cells**



Experimental setup

- **FPGA-based infrastructures for testing DDR4 and HBM2**
  - AMD Xilinx Alveo U200 and U280
  - Modified DRAM-Bender[1] for DDR4
  - Modified SoftMC[2] for HBM2
- **Temperature**
  - Temperature controller
  - Rubber heaters
  - Set the temperature to 75℃



[1] A. Olgun, H. Hassan, A. G. Yağlıkcı, Y. C. Tuğrul, L. Orosa, H. Luo, M. Patel, O. Ergin, and O. Mutlu, "DRAM Bender: An Extensible and Versatile FPGA-based Infrastructure to Easily Test State-of-the-art DRAM Chips," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2023.
[2] H. Hassan, N. Vijaykumar, S. Khan, S. Ghose, K. Chang, G. Pekhimenko, D. Lee, O. Ergin, and O. Mutlu, "SoftMC: A Flexible and Practical Open-Source Infrastructure for Enabling Experimental DRAM Studies," in HPCA, 2017.

Experimental setup

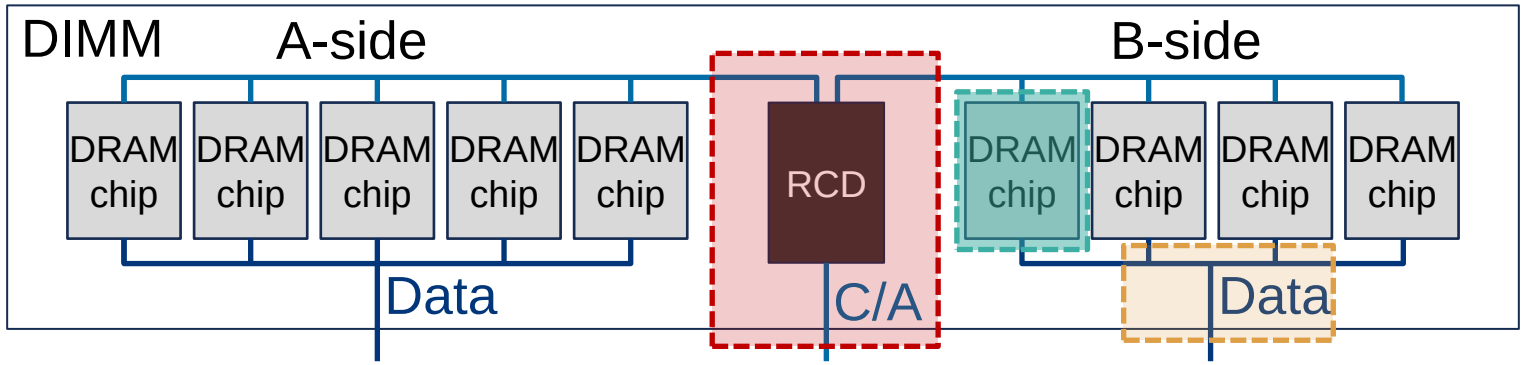
- **Tested DDR4 chips**
  - 376 DDR4 chips from three DRAM manufacturers
  - DDR4 chips on RDIMM
  - Various years and types
- **Tested HBM stacks**
  - 4 HBM2 stacks from one DRAM manufacturer

DRAM type	Vendor	Chip type	Density	Year	# chips
DDR4	Mfr. A	×4	8Gb	2016	80
		×4	8Gb	2017	16
		×4	8Gb	2018	32
		×4	8Gb	2021	32
		×8	8Gb	2017	16
		×8	8Gb	2018	32
		×8	8Gb	2019	16
DDR4	Mfr. B	×4	8Gb	2019	64
		×8	8Gb	2017	32
		×8	8Gb	2018	24
		×8	8Gb	2019	8
DDR4	Mfr. C	×4	8Gb	2018	32
		×4	8Gb	2021	32
		×8	8Gb	2016	8
		×8	8Gb	2019	16
HBM2	Mfr. A	4-Hi stack	4GB/stack	N/A	4

Common pitfalls

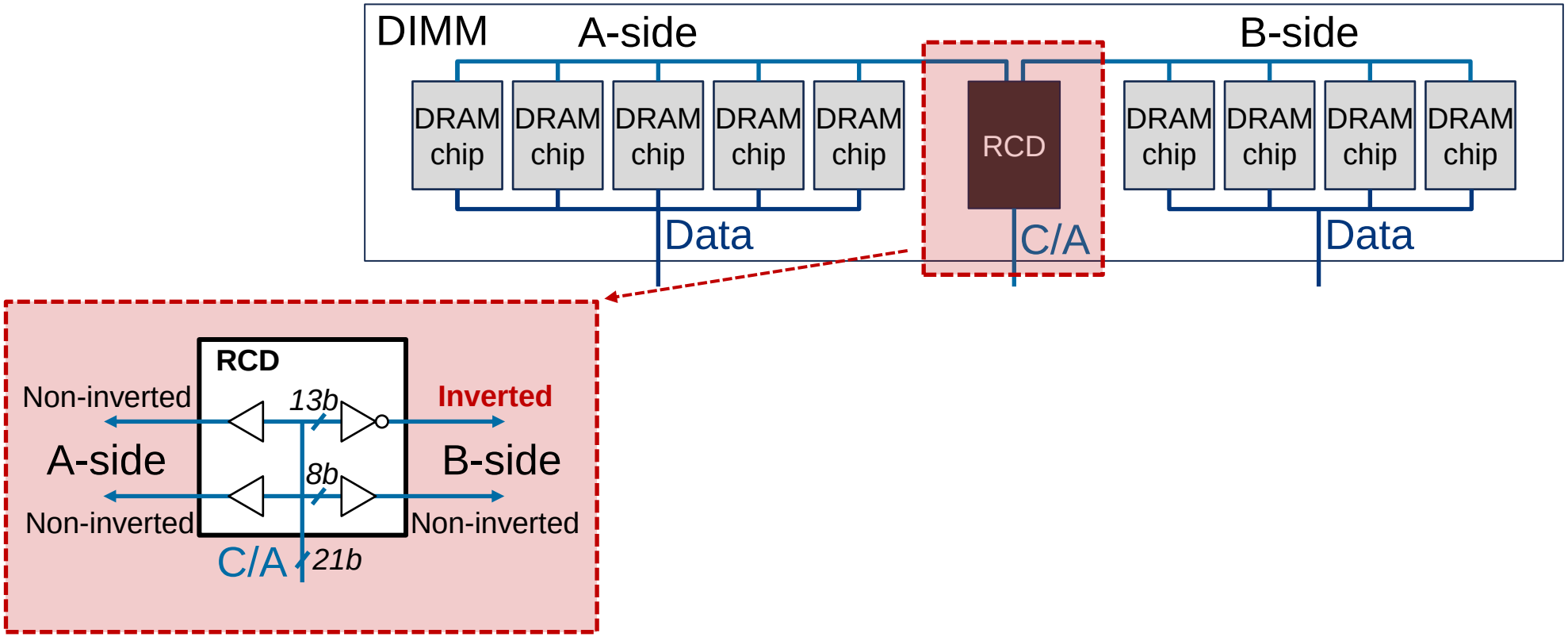
Common pitfalls

- Some information is disclosed, but they are regularly **overlooked.**



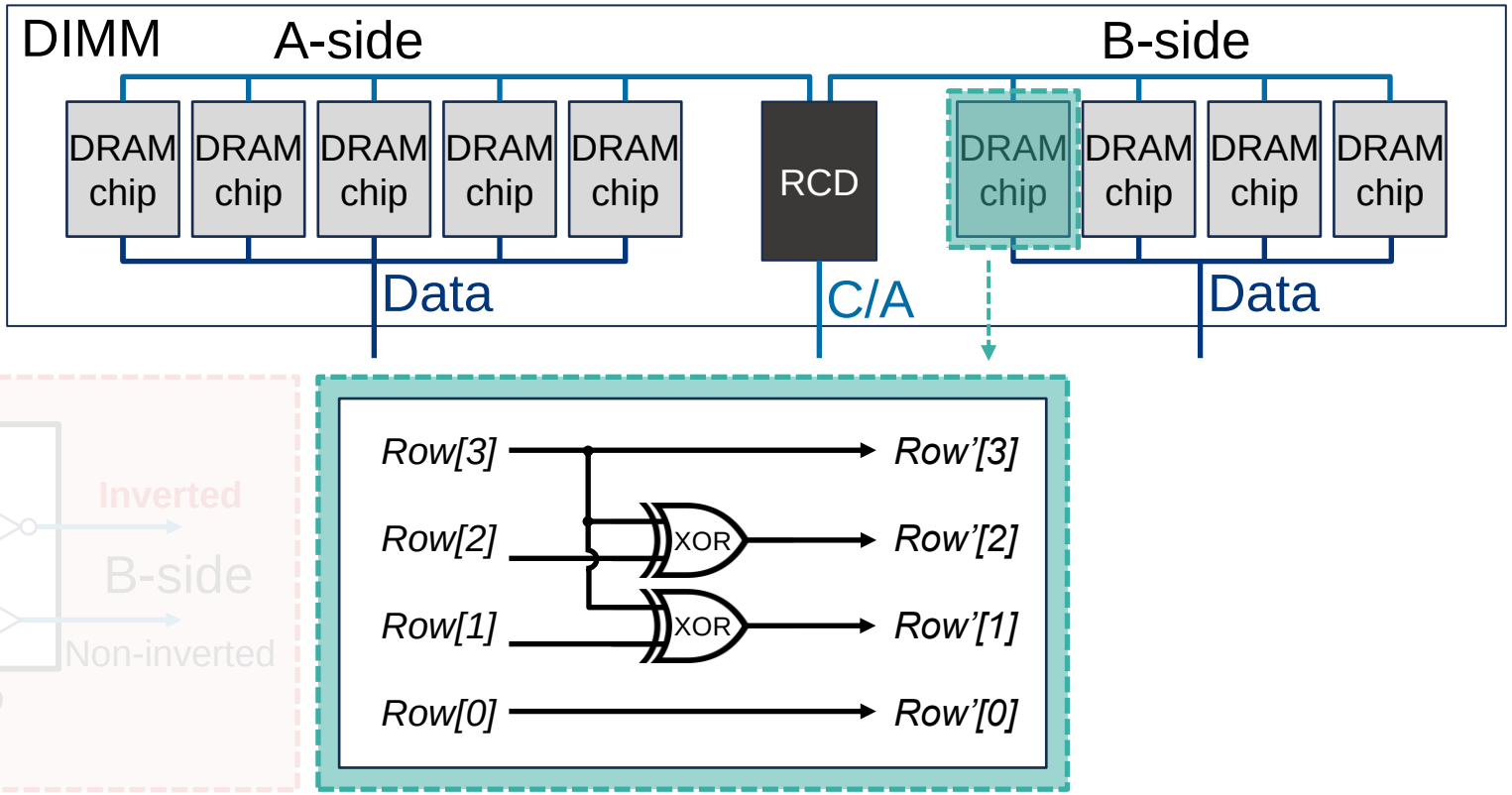
Common pitfalls – (1)

- **Address inversion in the RCD chip**
  - Some address bits is issued **invertedly** to the half of the chips on RDIMM (e.g., B-side chips).



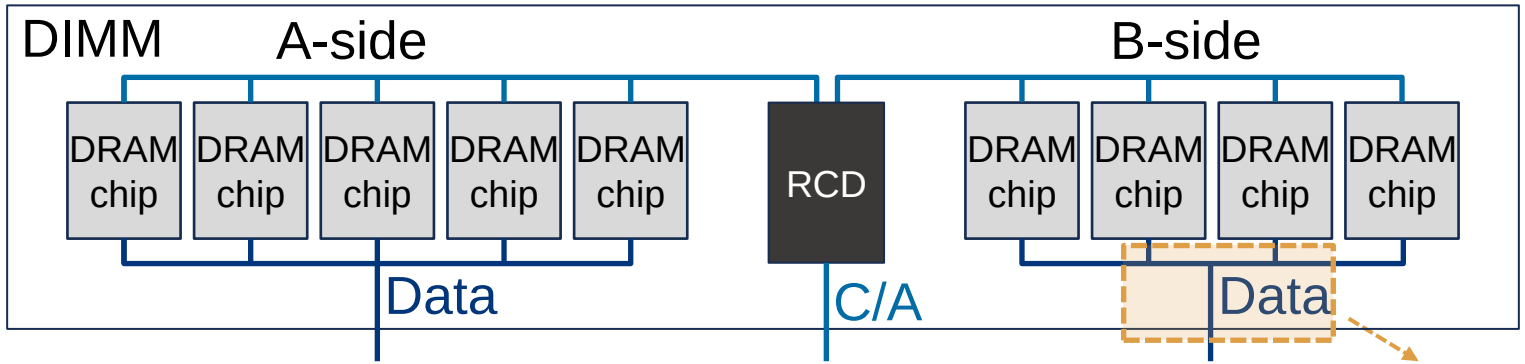
Common pitfalls – (2)

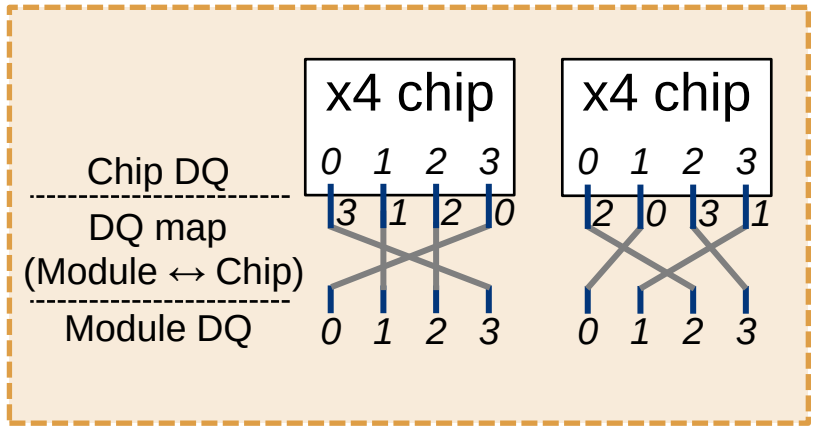
- **Internal row address remapping**
  - Many prior works have revealed this.
  - The **lower 4 bits** of the row address are **remapped**.



Common pitfalls – (3)

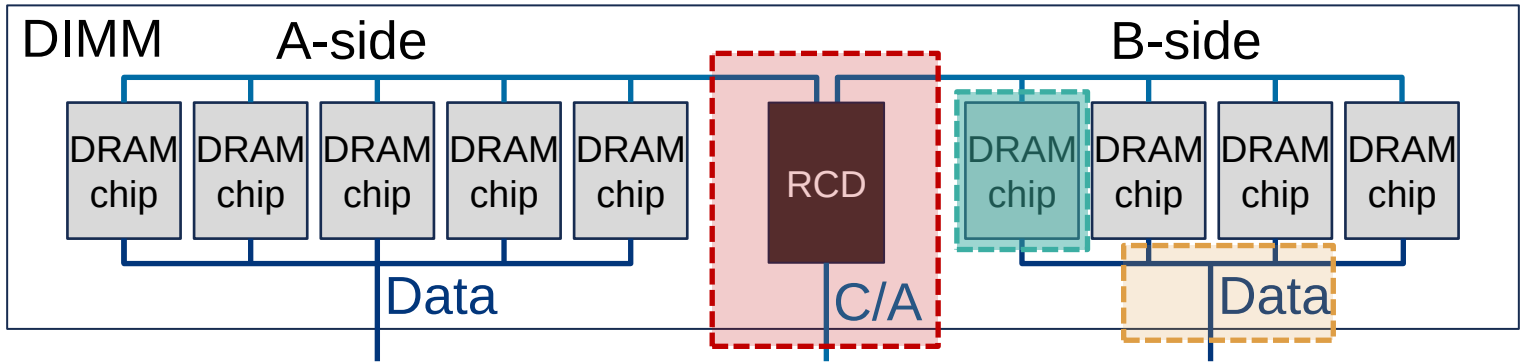
- **DQ pins of each chip are twisted**
  - This is disclosed in the standard document.
  - The DQ twisting method **varies** for each chip in the DIMM.

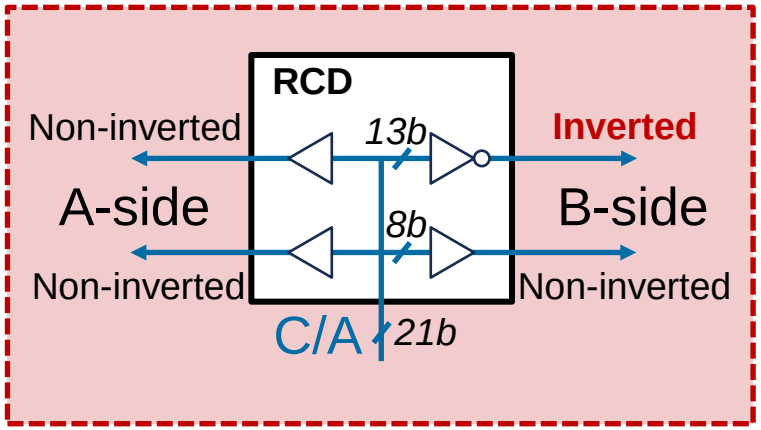


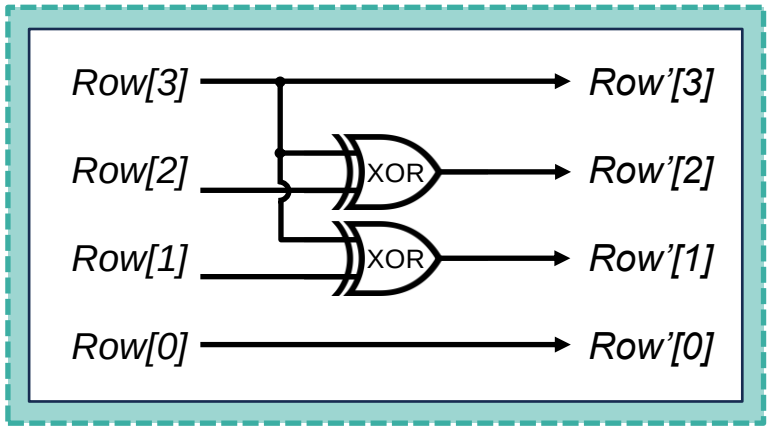


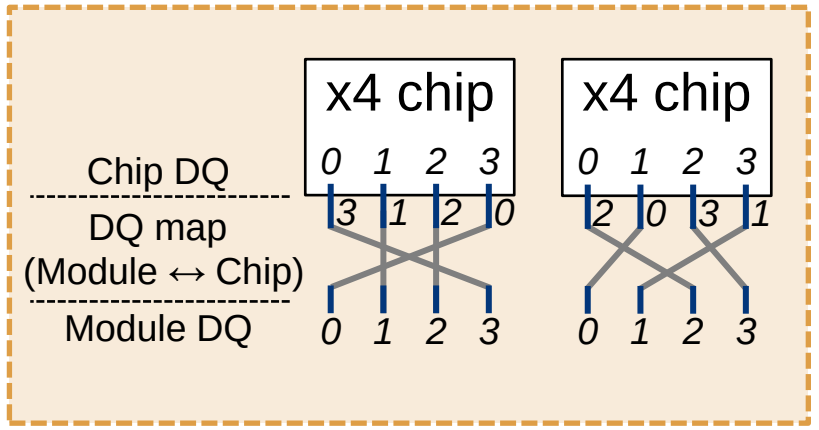
Common pitfalls

1) **The row address** can be remapped at the **RCD chip**.
2) **The row address** can be remapped by **the internal remapping scheme**.
3) **DQ pins** from a DIMM to each DRAM chip are remapped.









Macroscopic Analysis

Macroscopic Analysis

1) **Data Swizzling and MAT structure**

2) **Coupled-row activation**

3) **Subarray structure**

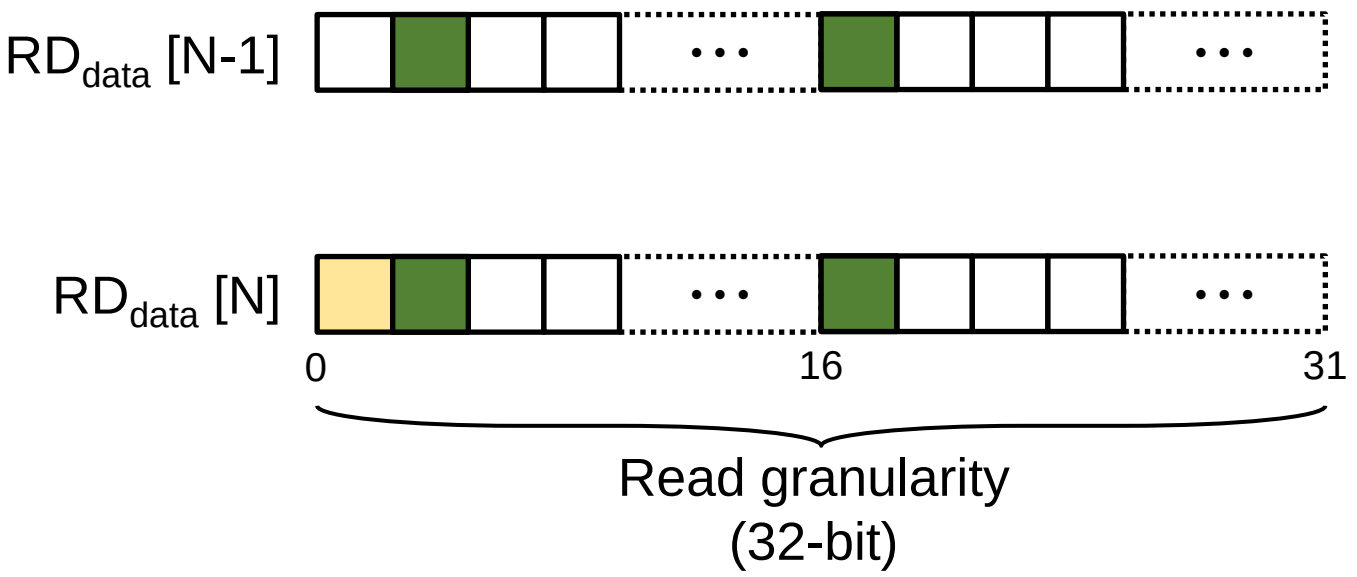
Macroscopic Analysis – (1)

- **Data Swizzling**
  - The data that enters a DRAM chip is not stored **sequentially**; instead, it is internally **swizzled and reorganized**.

Macroscopic Analysis – (1)

- **Data Swizzling**
  - The data that enters a DRAM chip is not stored **sequentially**; instead, it is internally **swizzled and reorganized**.

❶ Find the cell's adjacency based on the AIB's horizontal influence

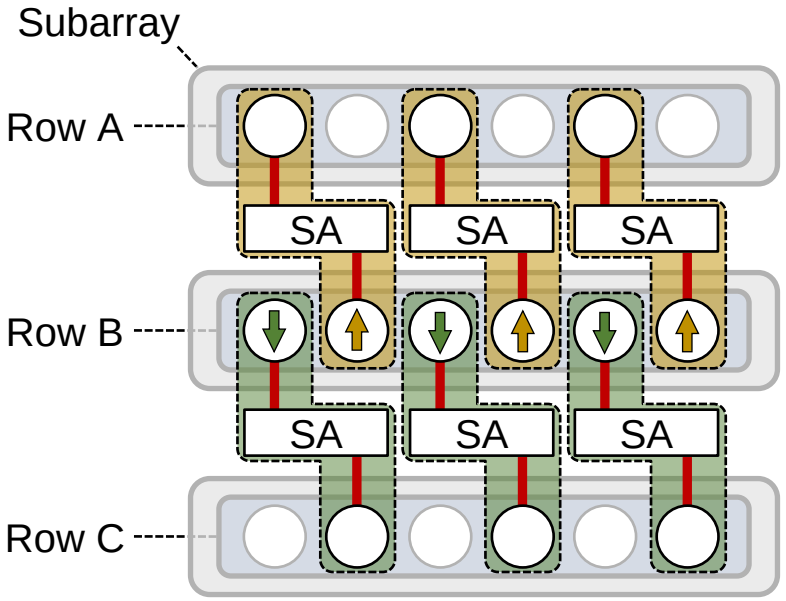


Macroscopic Analysis – (1)

- **Data Swizzling**
  - The data that enters a DRAM chip is not stored **sequentially**; instead, it is internally **swizzled and reorganized**.

❶ Find the cell's adjacency based on the AIB's horizontal influence

❷ Distinguish even BL and odd BL through RowCopy two rows in adjacent subarrays.

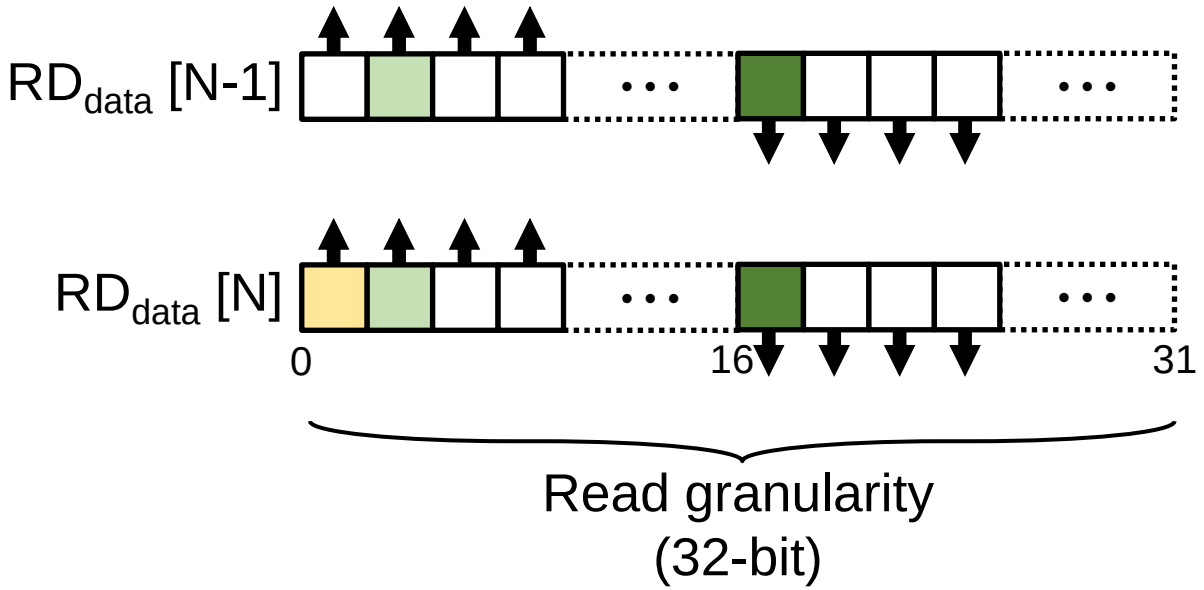


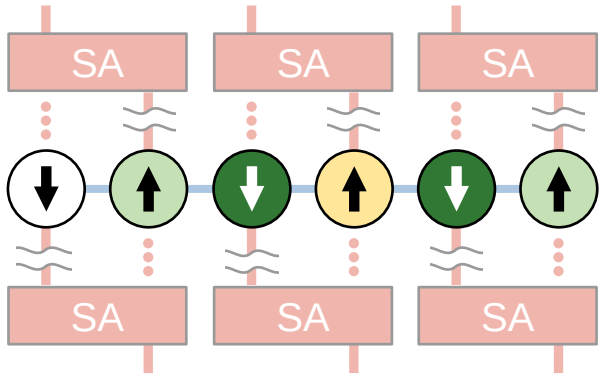
Macroscopic Analysis – (1)

- **Data Swizzling**
  - The data that enters a DRAM chip is not stored **sequentially**; instead, it is internally **swizzled and reorganized**.

❶ Find the cell's adjacency based on the AIB's horizontal influence

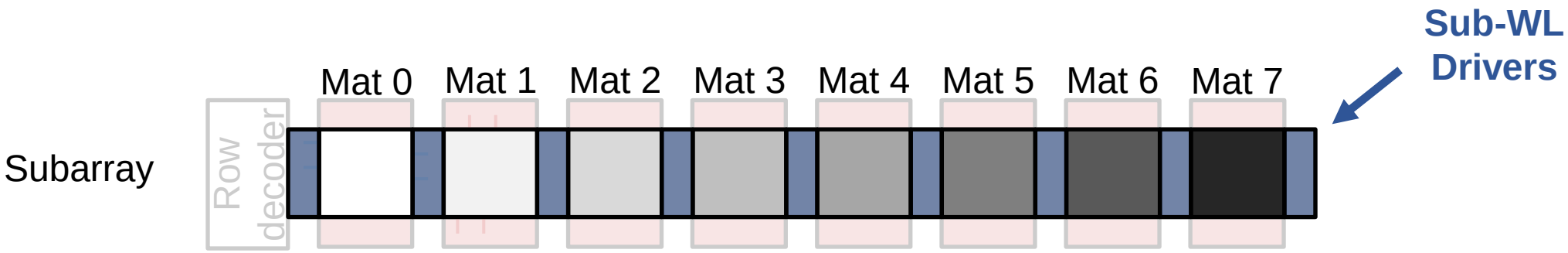
❷ Distinguish even BL and odd BL through RowCopy two rows in adjacent subarrays.





Macroscopic Analysis – (1)

- **Mat structure**
  - We speculate that a cell in one MAT is **difficult to be influenced** by a cell in another MAT due to **peripheral circuits** (e.g., sub-WL driver).
  - The measured **MAT widths** are 512-bit and 1024-bit.



Macroscopic Analysis – (1)

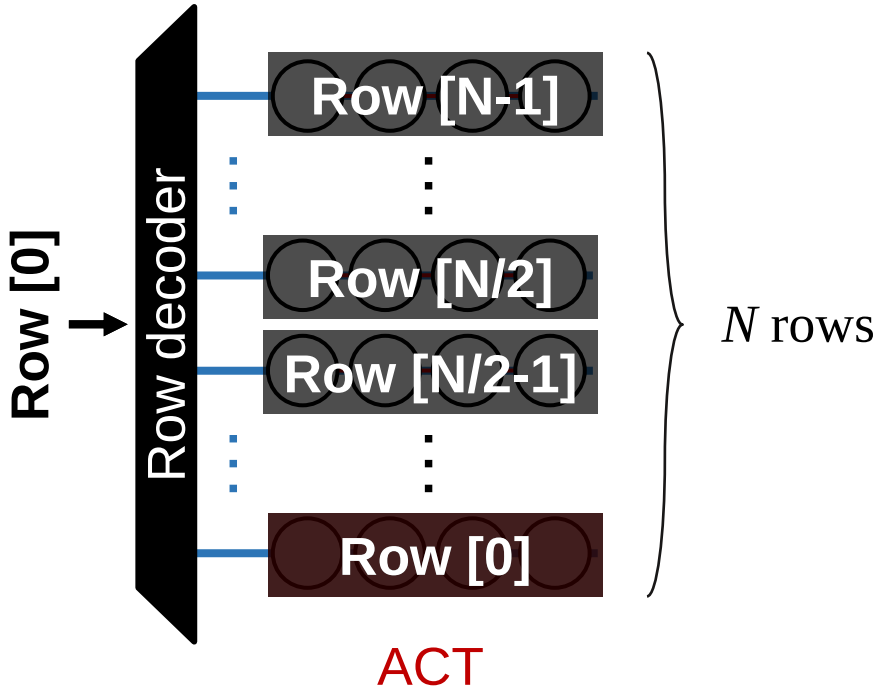
- **Mat structure**

Inside a DRAM chip, data is remapped by data swizzling

The MAT widths are 512-bit and 1024-bit

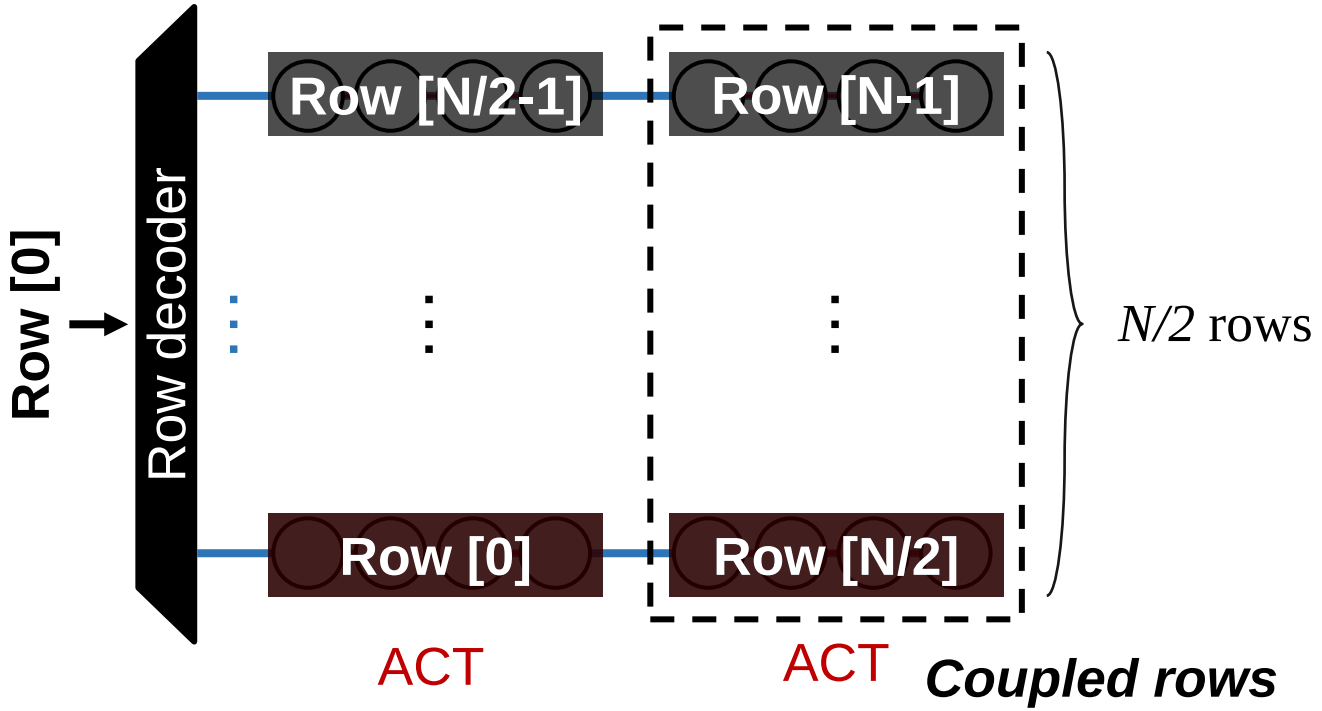
Macroscopic Analysis – (2)

- **Coupled-row Activation**
  - Two rows indexed by **different physical addresses** but mapped to the **same DRAM row** as a **coupled-row** pair
  - We discover coupled-row with AIBs and RowCopy
  - A coupled-row activation is identified for certain **×4 DRAM chips and HBM2**, and the interval between the coupled-rows is $N_{row}/2$



Macroscopic Analysis – (2)

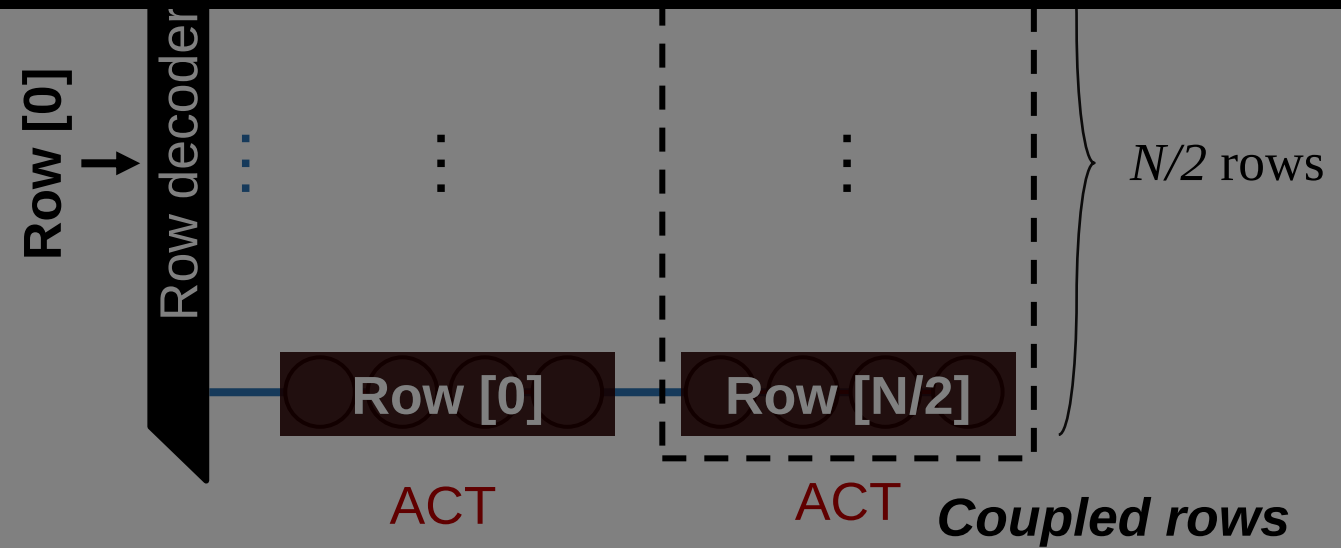
- **Coupled-row Activation**
  - Two rows indexed by **different physical addresses** but mapped to the **same DRAM row** as a **coupled-row** pair
  - We discover coupled-row with AIBs and RowCopy
  - A coupled-row activation is identified for certain **×4 DRAM chips and HBM2**, and the interval between the coupled-rows is $N_{row}/2$



Macroscopic Analysis – (2)

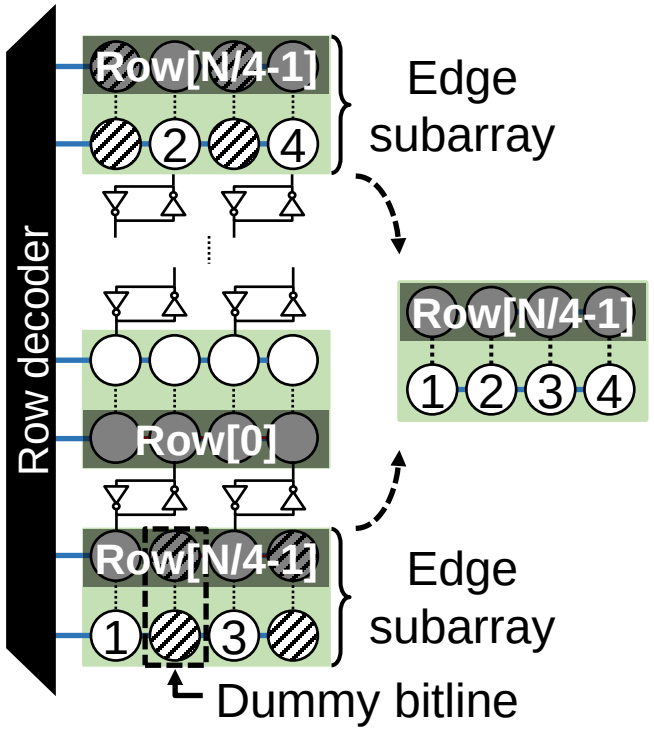
- **Coupled-row Activation**
  - Two rows indexed by **different physical addresses** but mapped to the **same DRAM row** as a **coupled-row** pair.
  - We discover coupled-row with AIBs and RowCopy

For some DRAM chips, activating a row can unintendedly activate its coupled row



Macroscopic Analysis – (3)

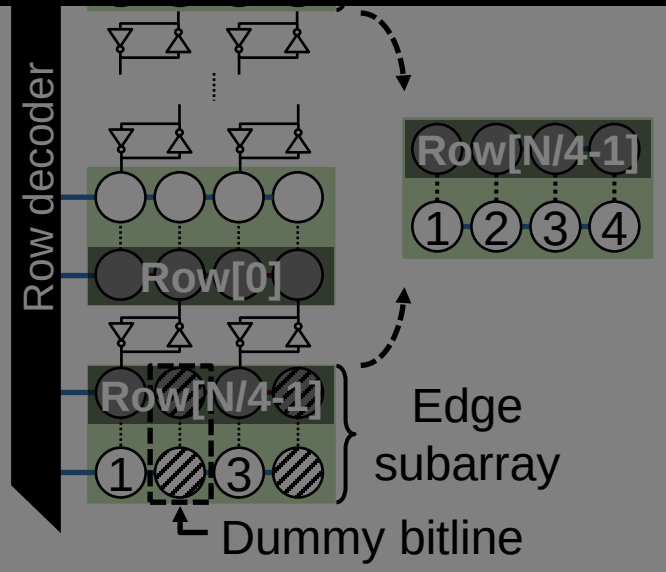
- **Edge subarray**
  - Most subarrays are **sequentially adjacent**, following the **row address order.**
  - We observed that RowCopy affects rows with a **large difference** in the row address (e.g., 0th row ⇔ (N/4-1)th row).
  - These subarrays are called **edge subarrays**, and it is aligned with prior open bitline structures proposed.



Macroscopic Analysis – (3)

- **Edge subarray**
  - Most subarrays are **sequentially adjacent**, following the **row address order.**
  - We observed that RowCopy affects rows with a **large difference** in the row address (e.g., 0th row ⇔ (N/4-1)th row).

Two edge subarrays work in tandem to form a single full subarray.



Microscopic Analysis

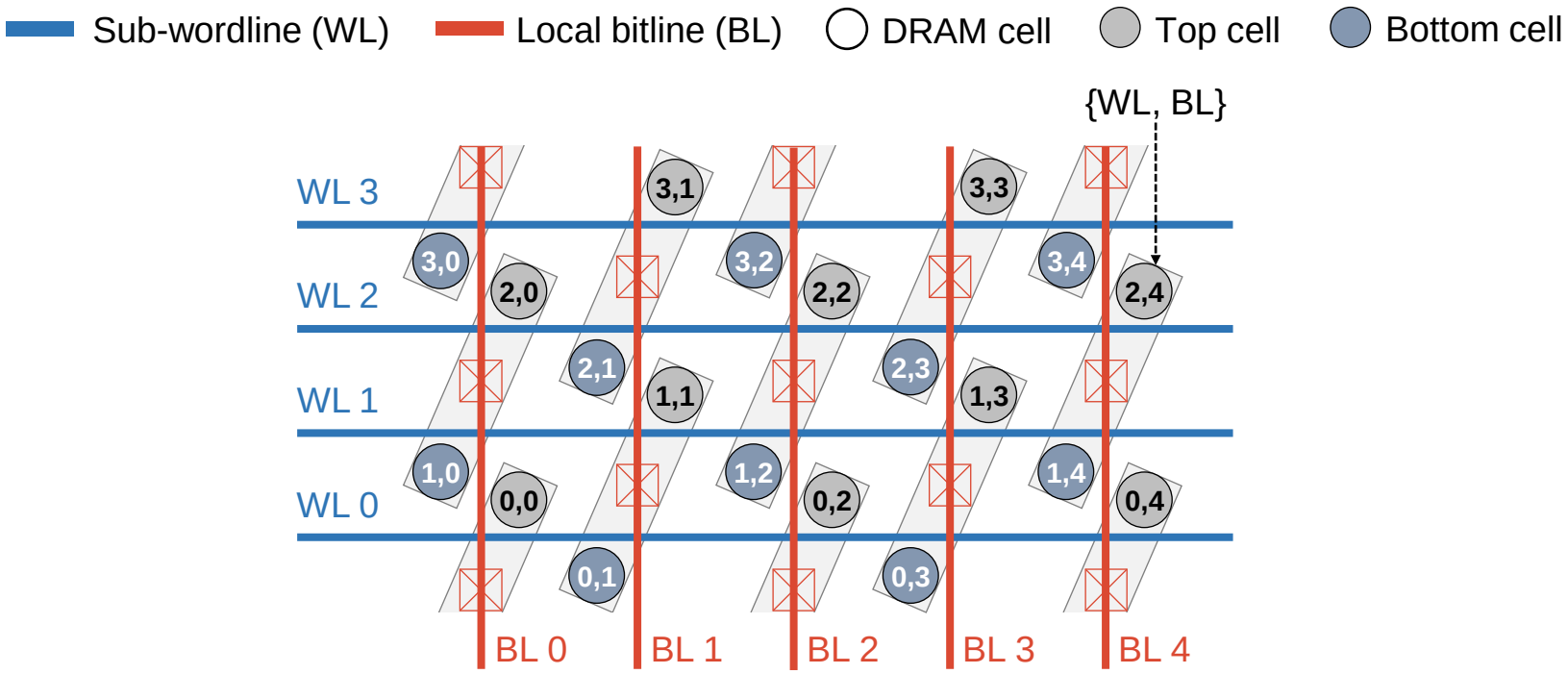
Microscopic Analysis

1) 6F^2-induced AIB Characteristics

2) Adversarial Data Pattern for H_{cnt} and BER

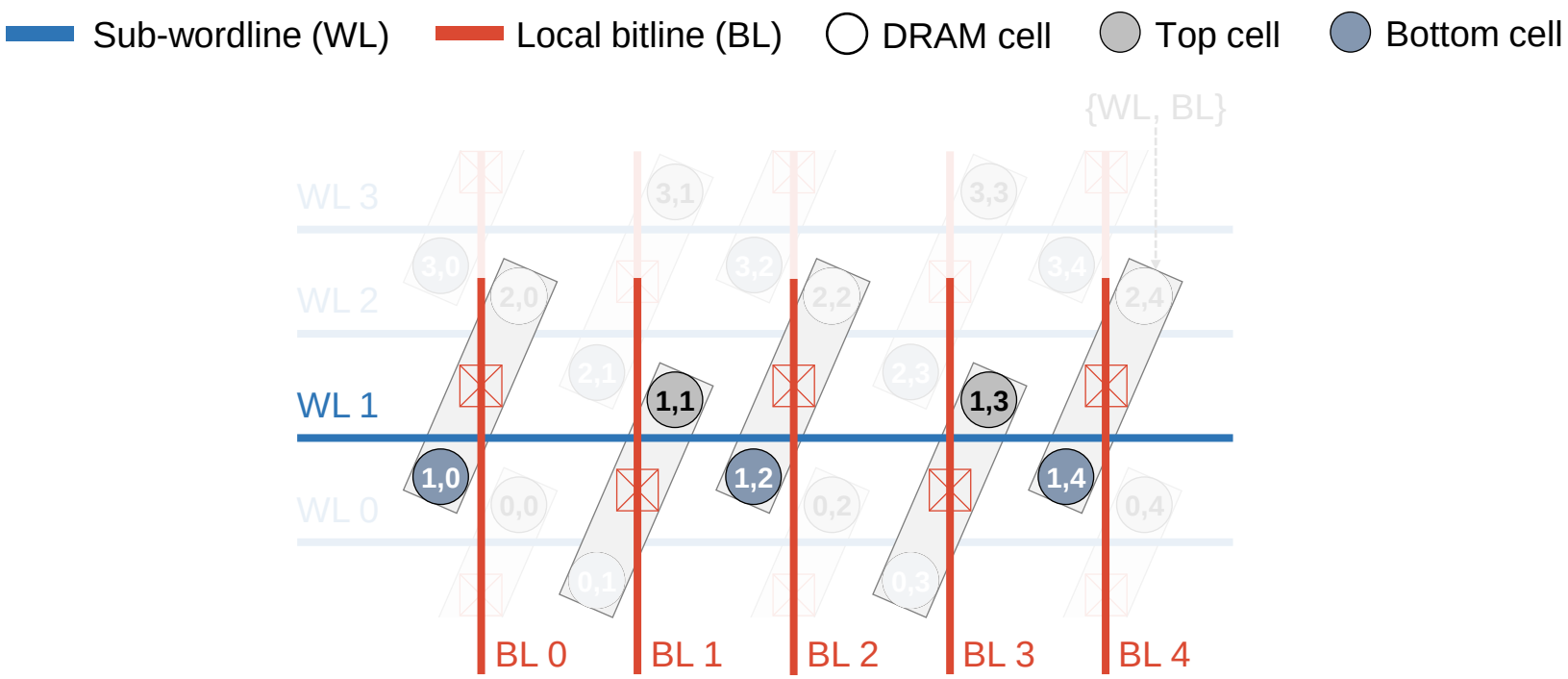
Microscopic Analysis – (1)

- **$6F^2$ DRAM layout**



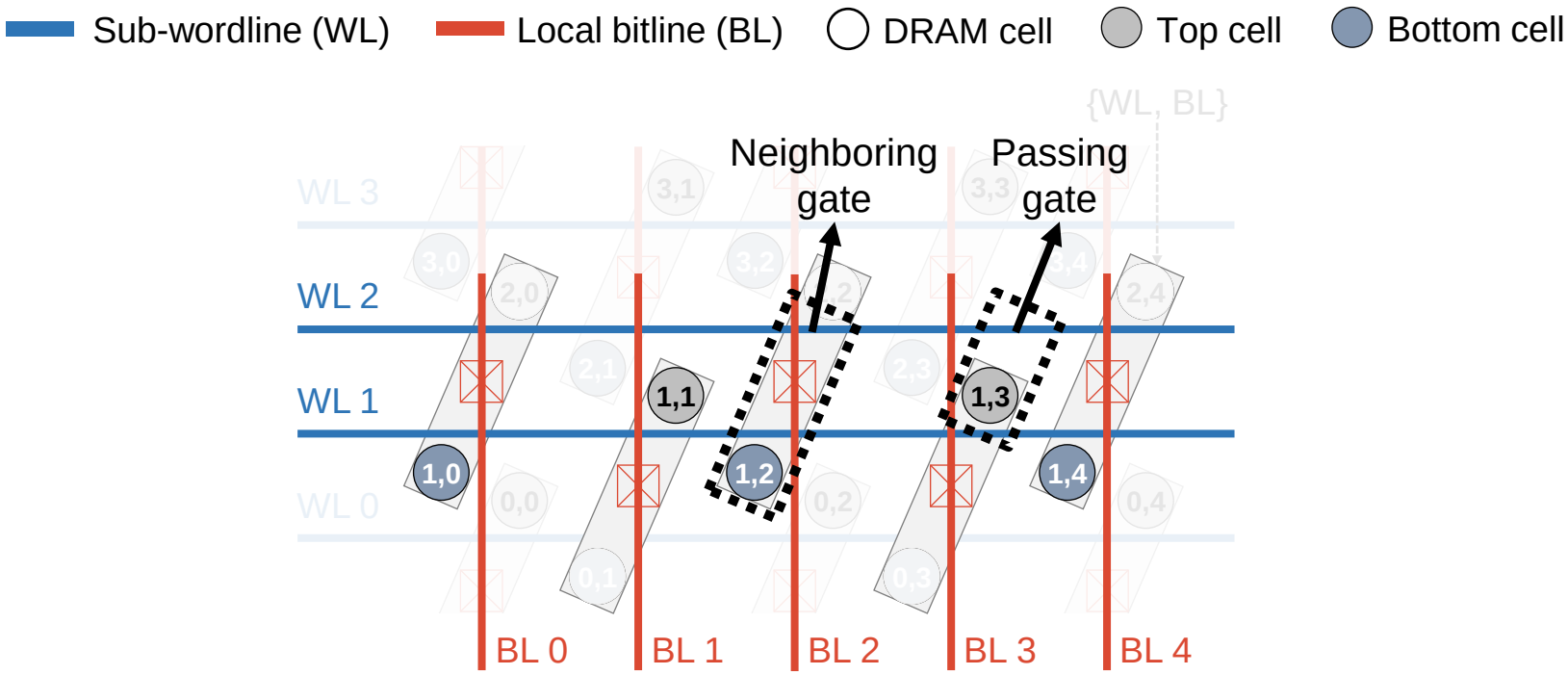
Microscopic Analysis – (1)

- **$6F^2$ DRAM layout**
  - In a single row, the cells **alternate top and bottom** of the wordline



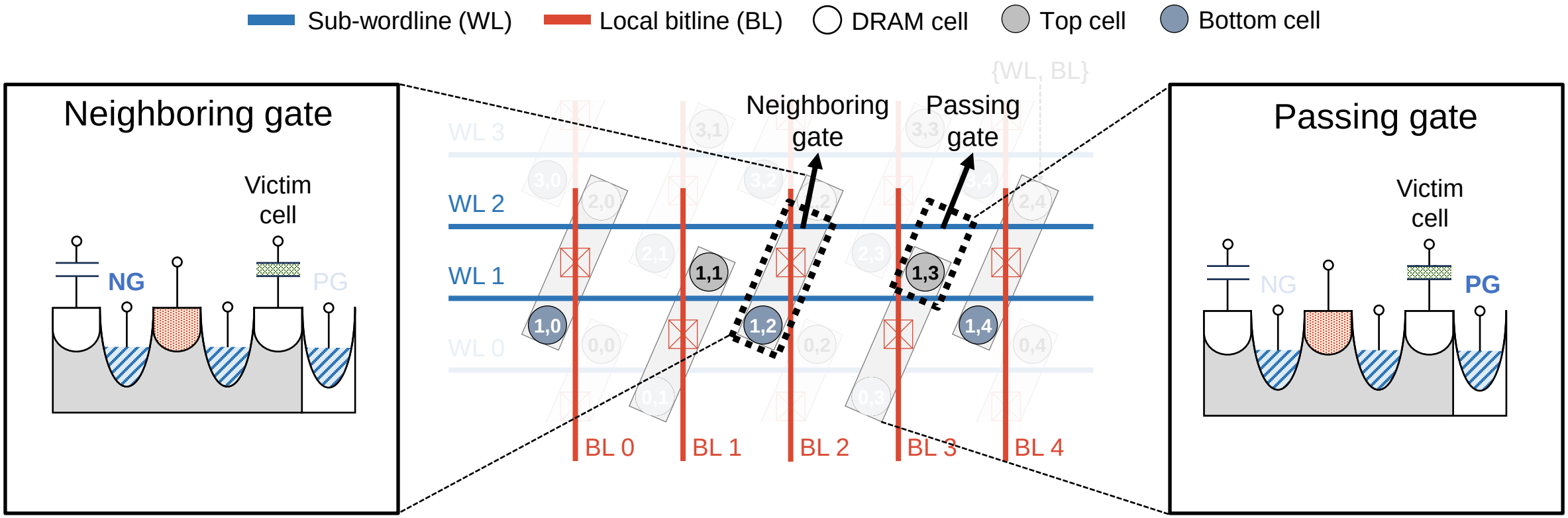
Microscopic Analysis – (1)

- **$6F^2$ DRAM layout**
  - In a single row, the cells **alternate top and bottom** of the wordline
  - The cells of a single row alternately have a passing gate and neighboring gate relationship with the adjacent wordline



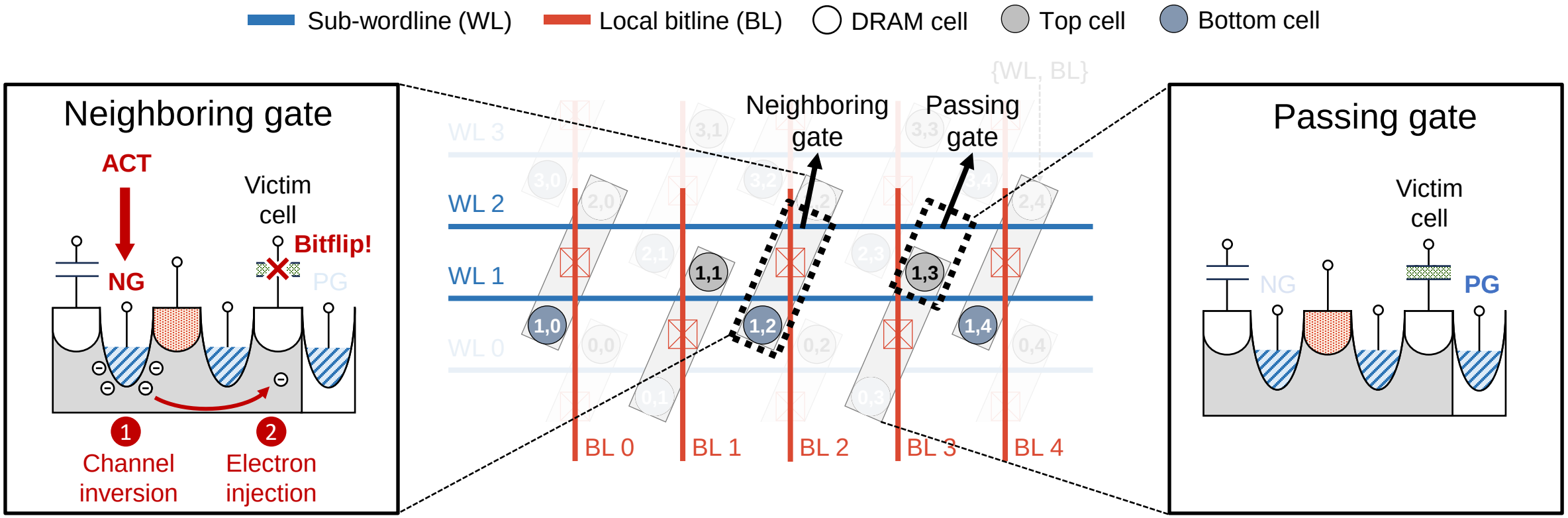
Microscopic Analysis – (1)

- **$6F^2$ DRAM layout**
  - In a single row, the cells **alternate top and bottom** of the wordline
  - The cells of a single row alternately have a passing gate and neighboring gate relationship with the adjacent wordline



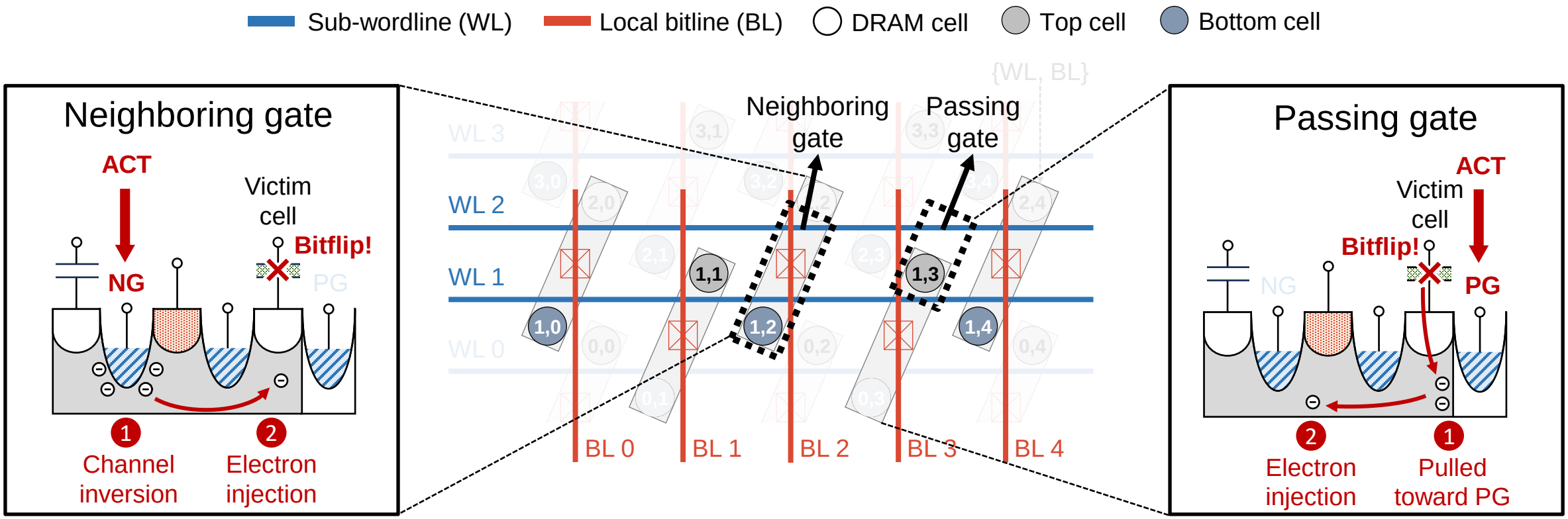
Microscopic Analysis – (1)

- **$6F^2$ DRAM layout**
  - In a single row, the cells **alternate top and bottom** of the wordline
  - The cells of a single row alternately have a passing gate and neighboring gate relationship with the adjacent wordline



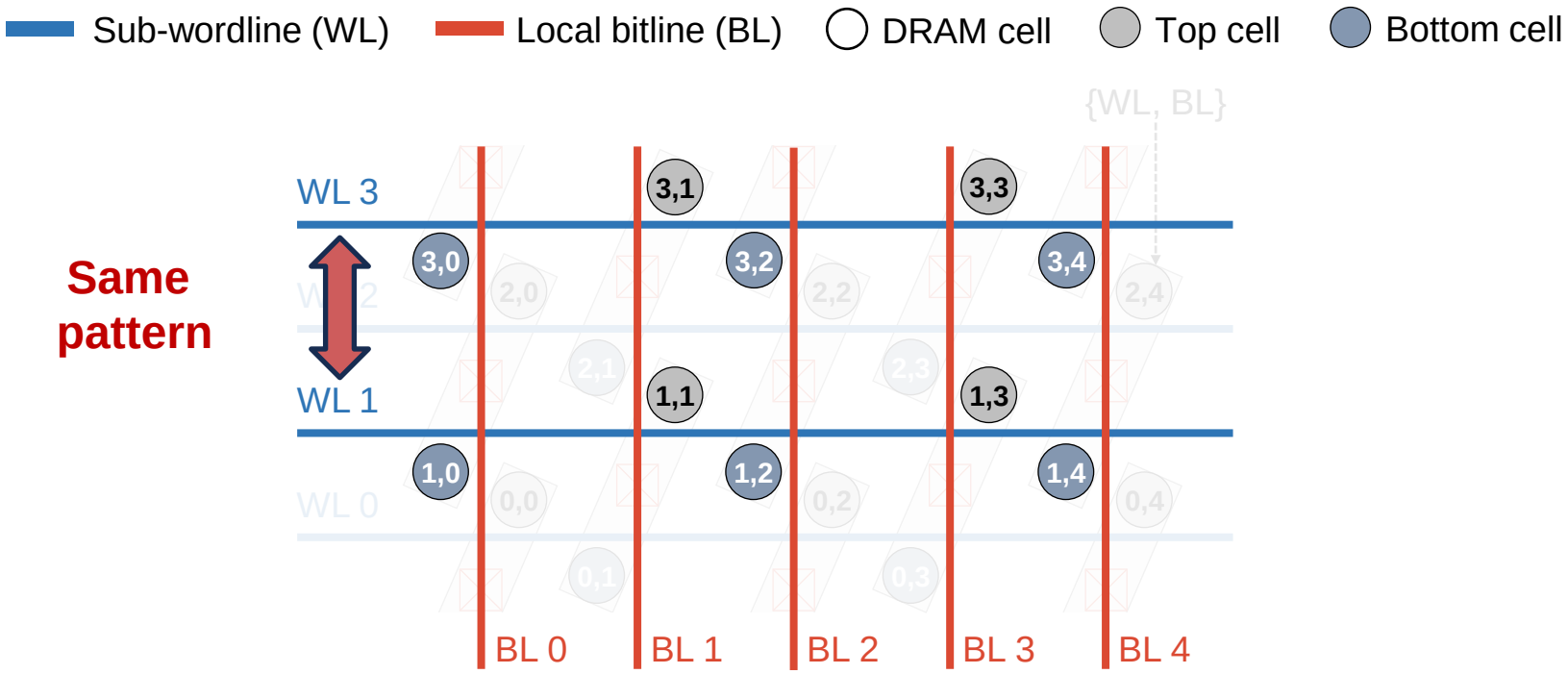
Microscopic Analysis – (1)

- **$6F^2$ DRAM layout**
  - In a single row, the cells **alternate top and bottom** of the wordline
  - The cells of a single row alternately have a passing gate and neighboring gate relationship with the adjacent wordline



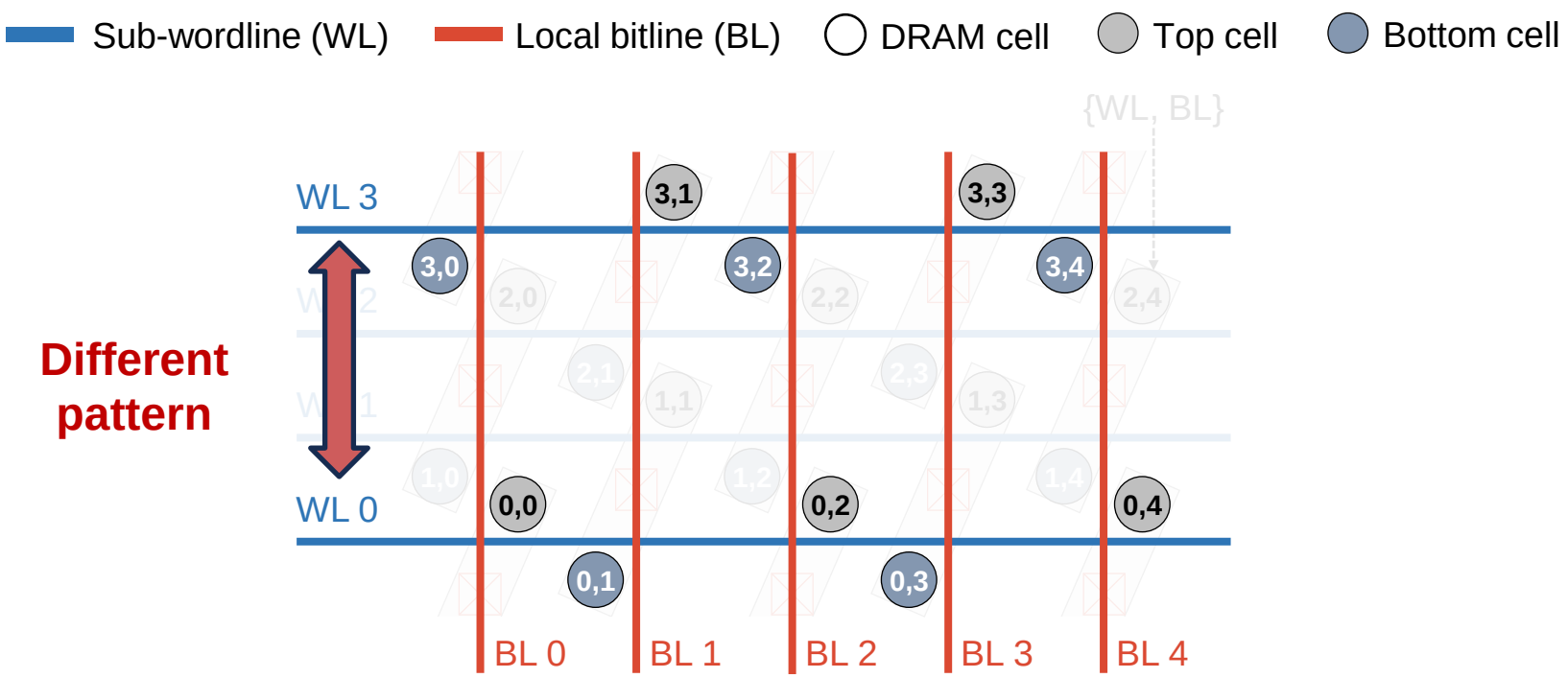
Microscopic Analysis – (1)

- **$6F^2$ DRAM layout**



Microscopic Analysis – (1)

- **$6F^2$ DRAM layout**

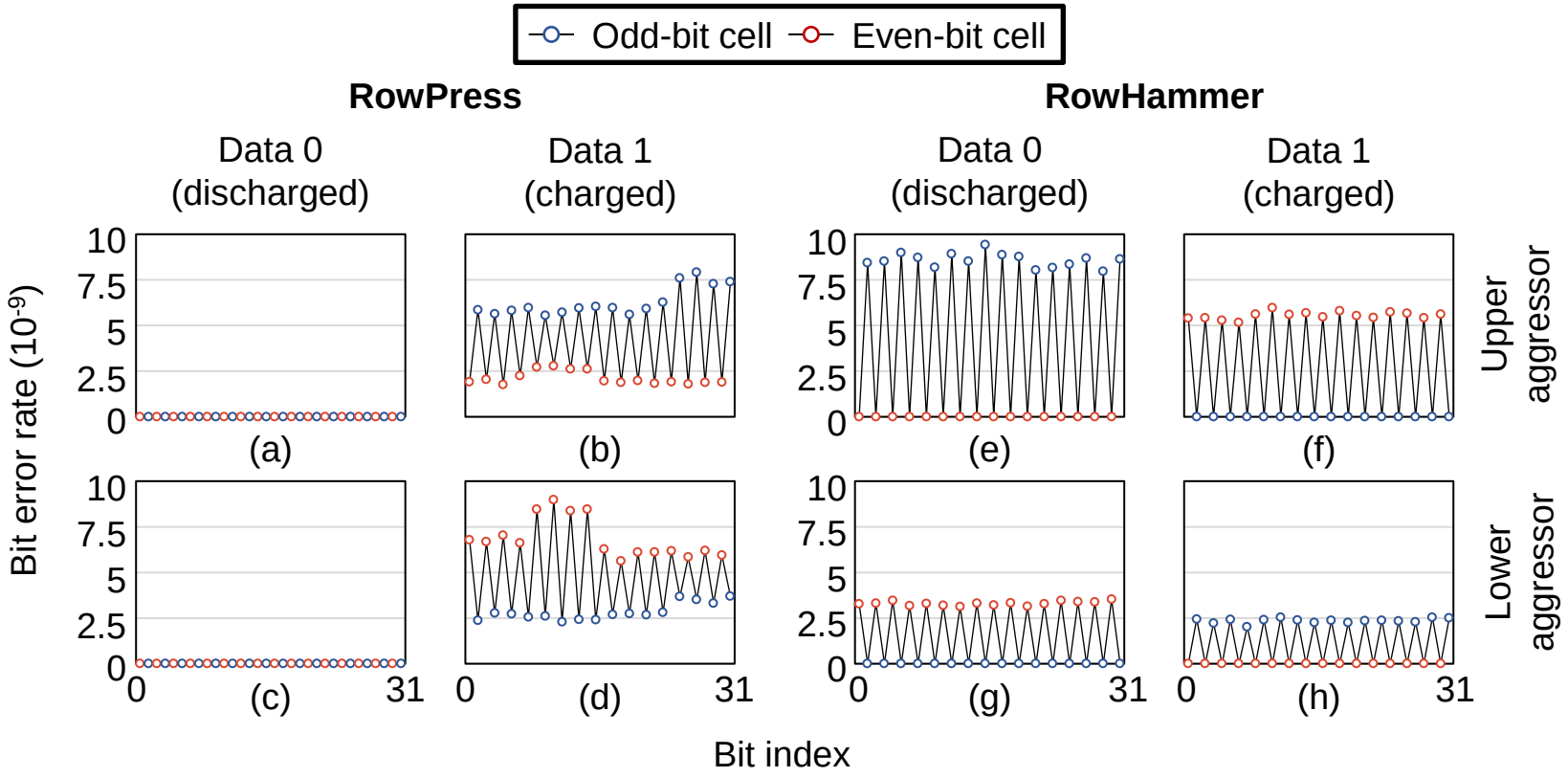


Microscopic Analysis – (1)

- **Attack patterns of AIBs**
  - **RowHammer**
    - = 1024 rows
    - = Single-sided attack
    - = ***300K*** activations
    - = ***35 ns*** for each activation
  - **RowPress**
    - = 1024 rows
    - = Single-sided attack
    - = ***8K*** activations
    - = ***7.8 us*** for each activation

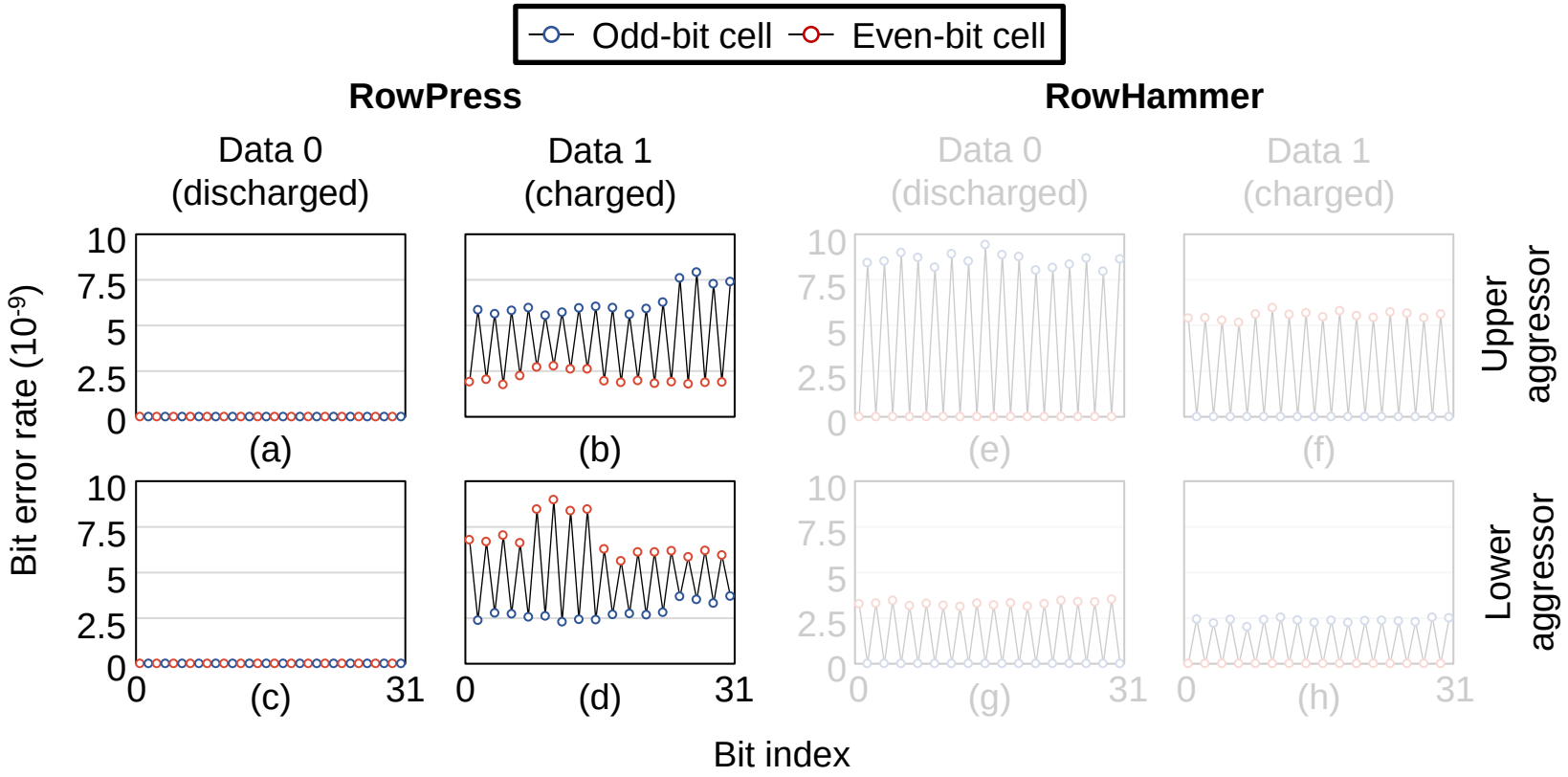
Microscopic Analysis – (1)

- **$6F^2$-induced AIB Characteristics**



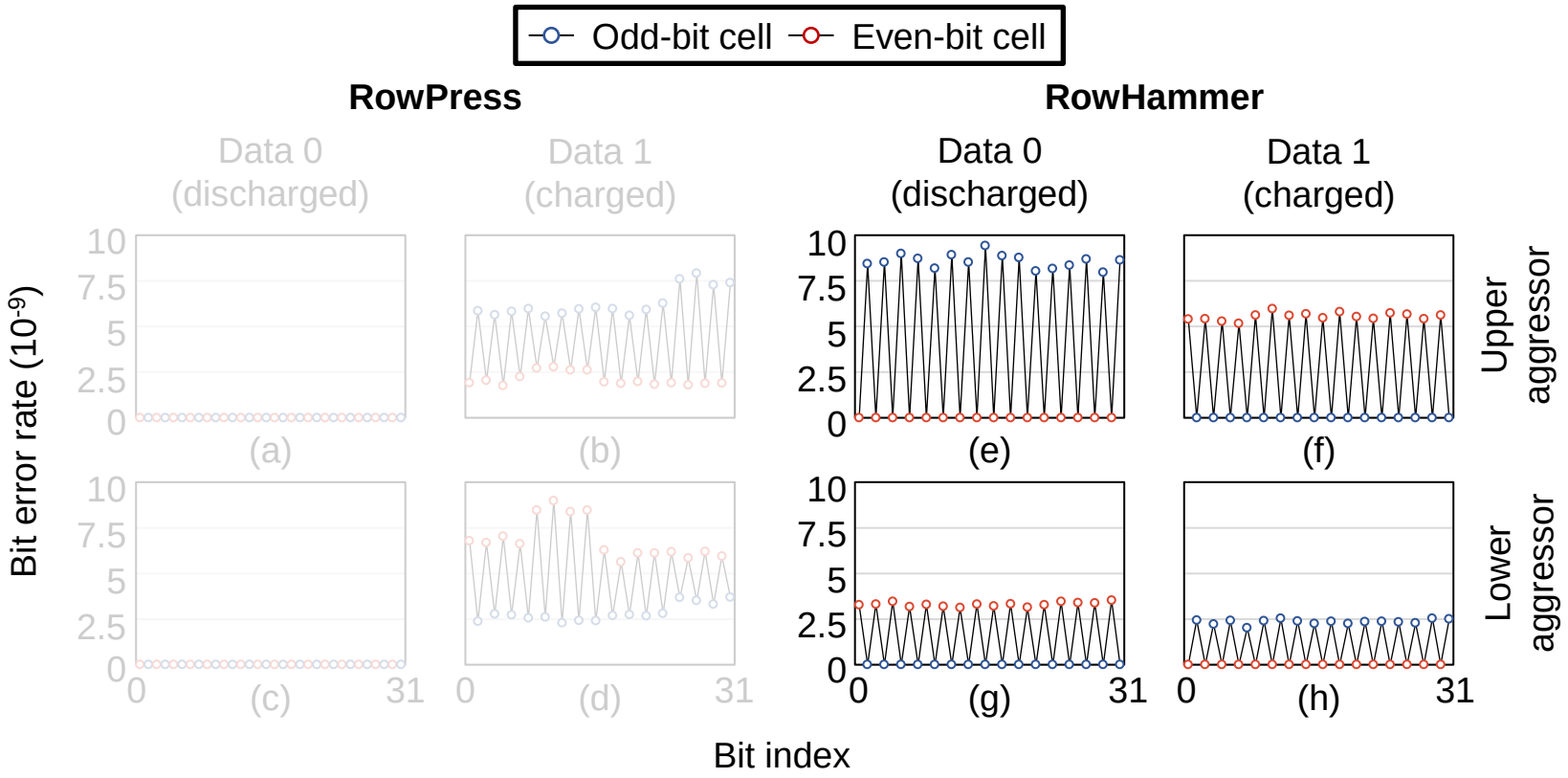
Microscopic Analysis – (1)

- **$6F^2$-induced AIB Characteristics**



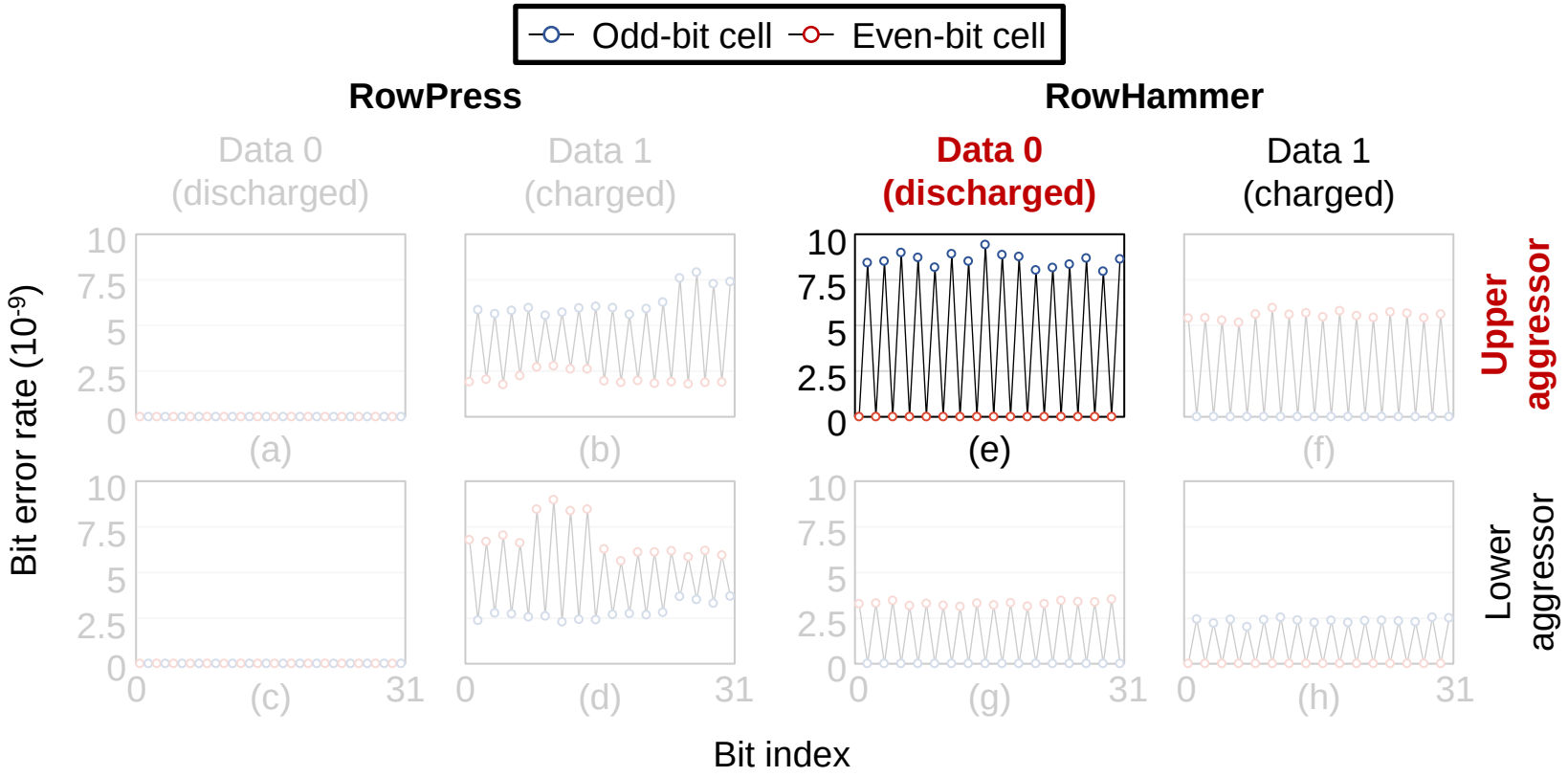
Microscopic Analysis – (1)

- **$6F^2$-induced AIB Characteristics**



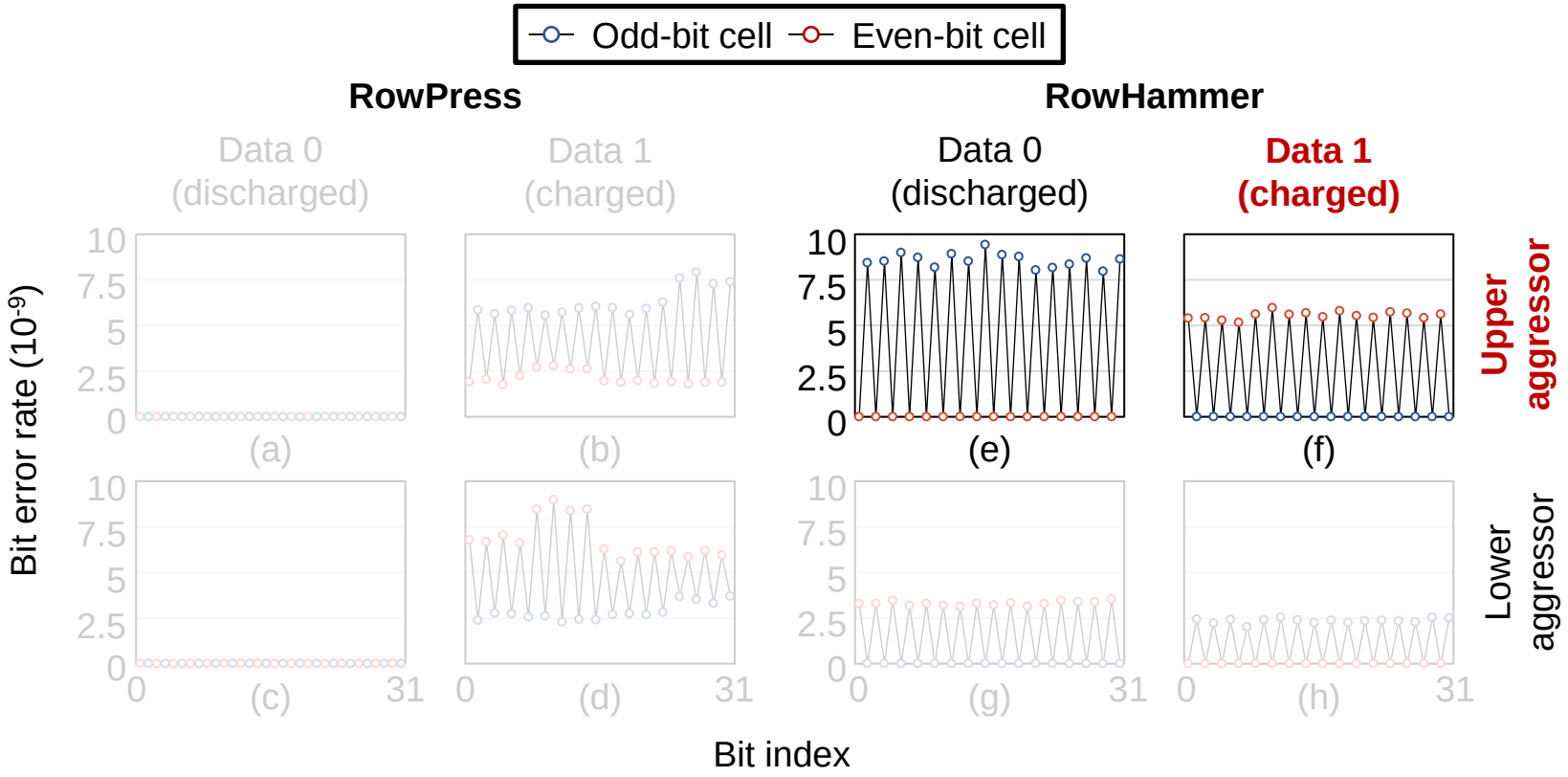
Microscopic Analysis – (1)

- **$6F^2$-induced AIB Characteristics**



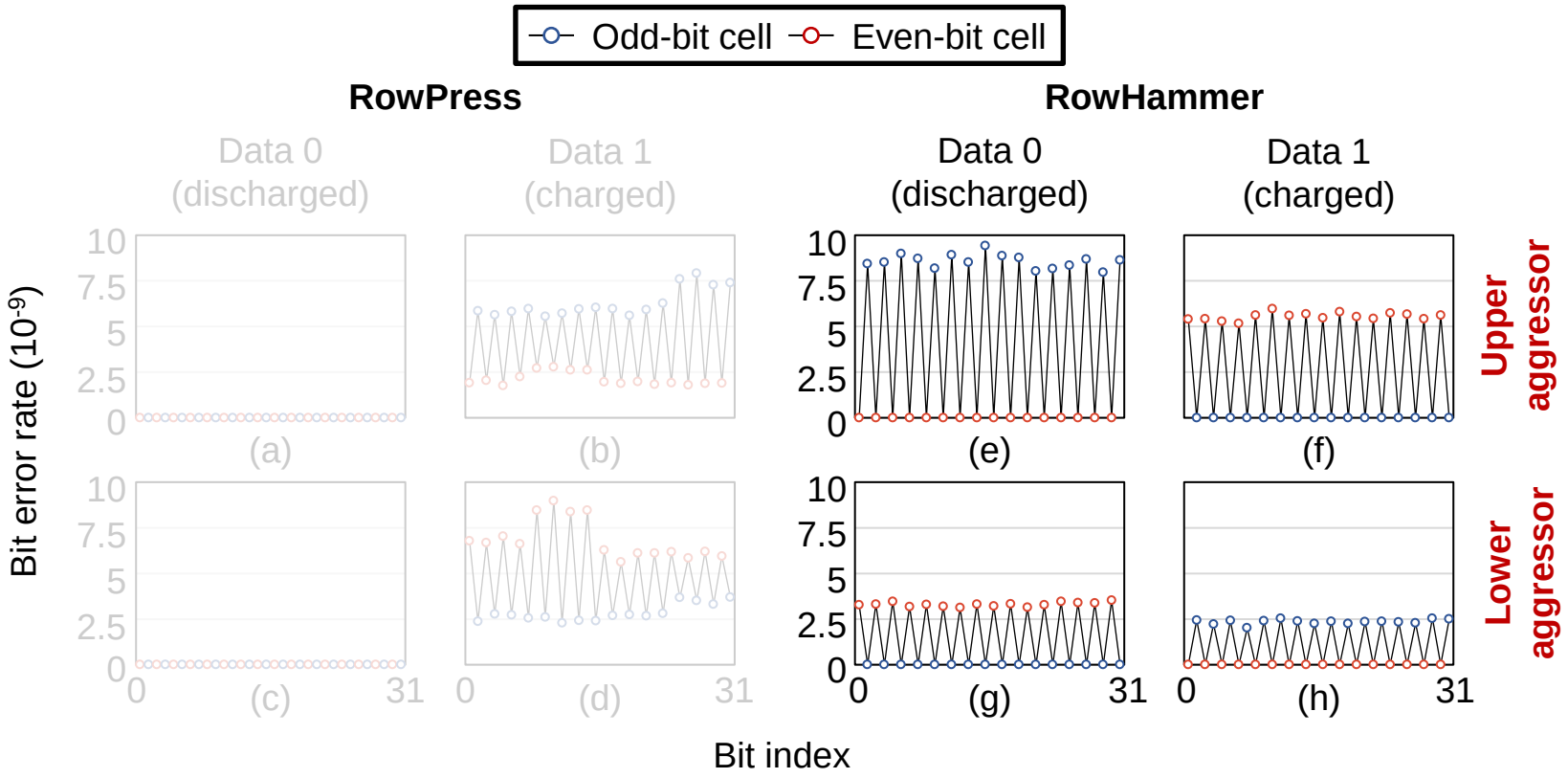
Microscopic Analysis – (1)

- **$6F^2$-induced AIB Characteristics**



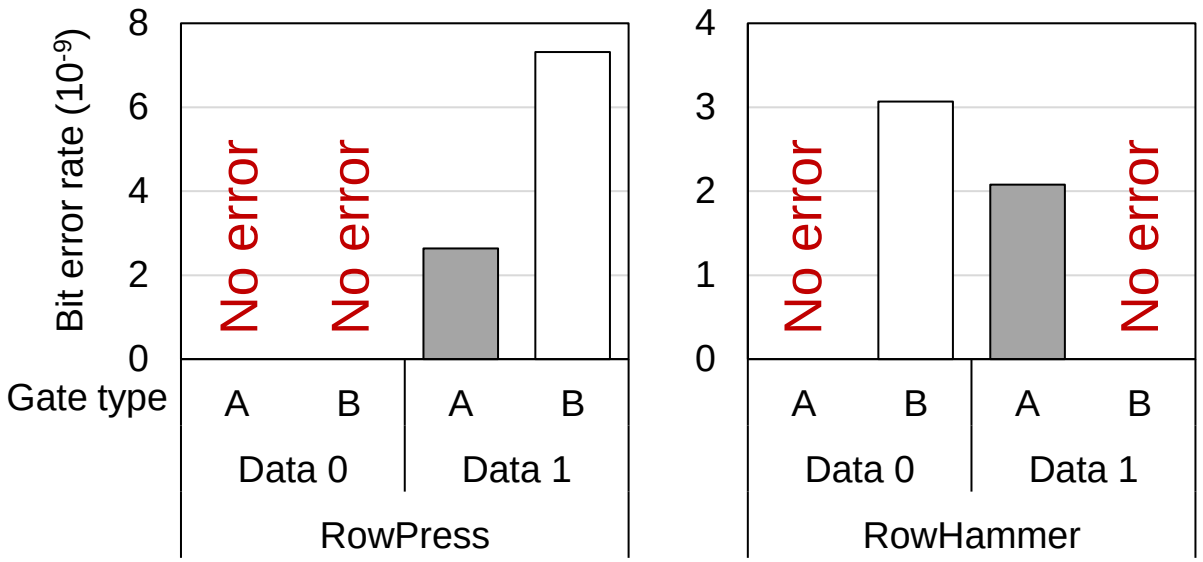
Microscopic Analysis – (1)

- **$6F^2$-induced AIB Characteristics**



Microscopic Analysis – (1)

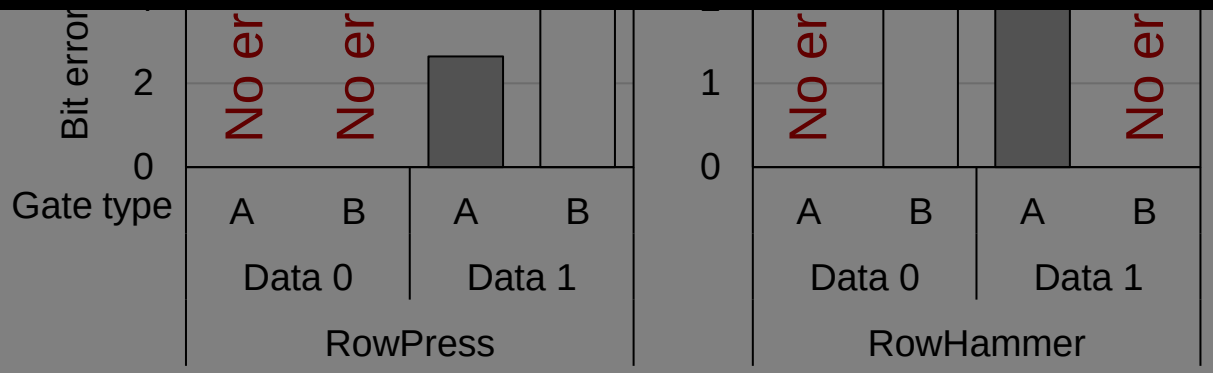
- **$6F^2$-induced AIB Characteristics**
  - Different AIBs exhibit different characteristics depending on **stored data**, **attack method** and **gate type**



Microscopic Analysis – (1)

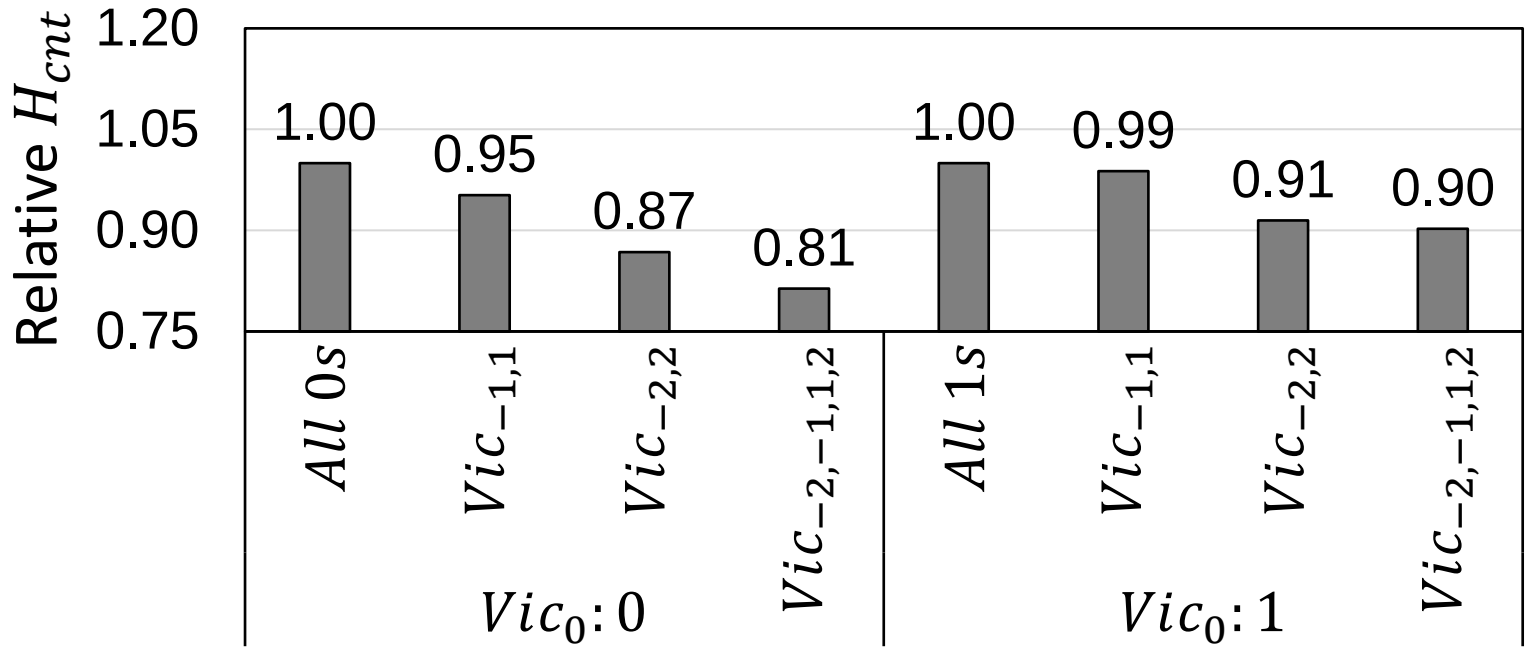
- **$6F^2$-induced AIB Characteristics**
  - Different AIBs exhibit different characteristics depending on the **attack method** and **gate type**

Activate-induced bitflips are classified based on the combination of
[Data] x [Attack method] x **[Gate type]**



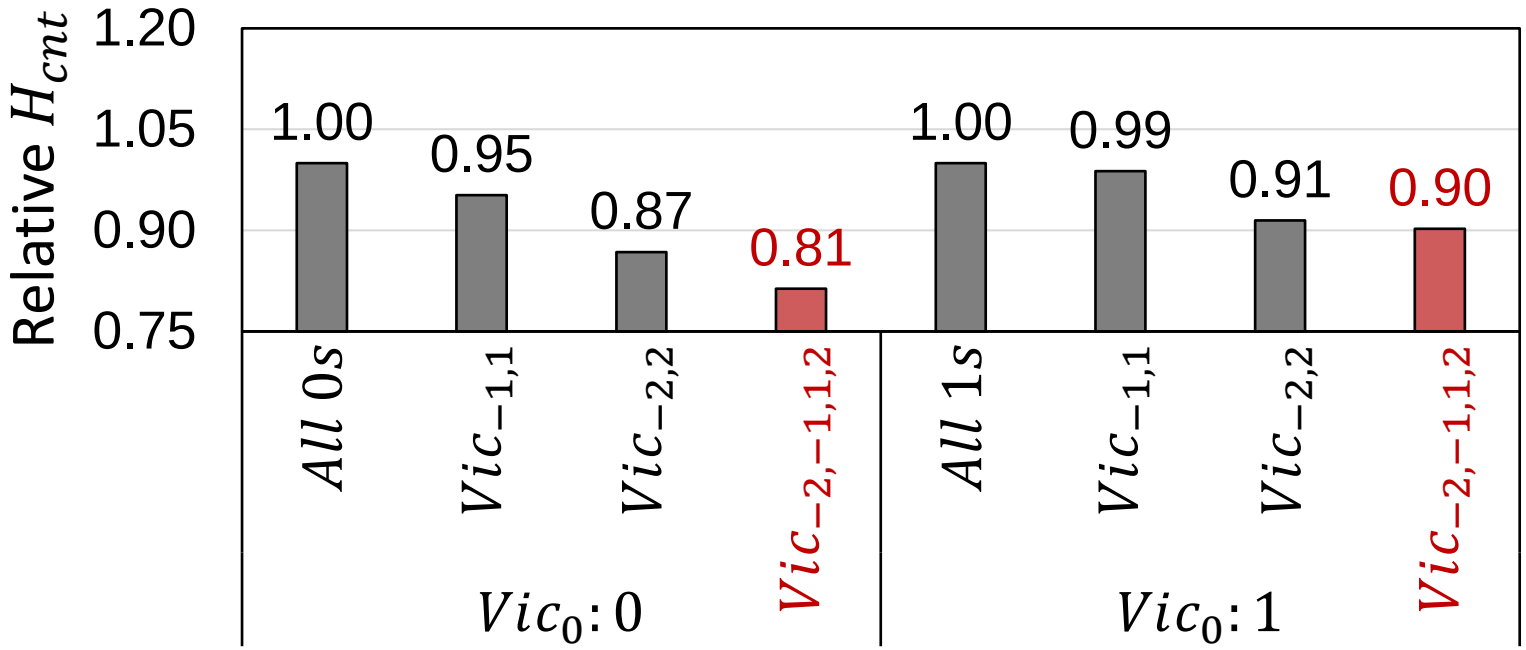
Microscopic Analysis – (2)

- **Adversarial Data Pattern for H_{cnt}**
  - $Vic_{-2,-1,1,2}$ and $Aggr_{-2,-1,0,1,2}$ hold the opposite value of Vic_0.



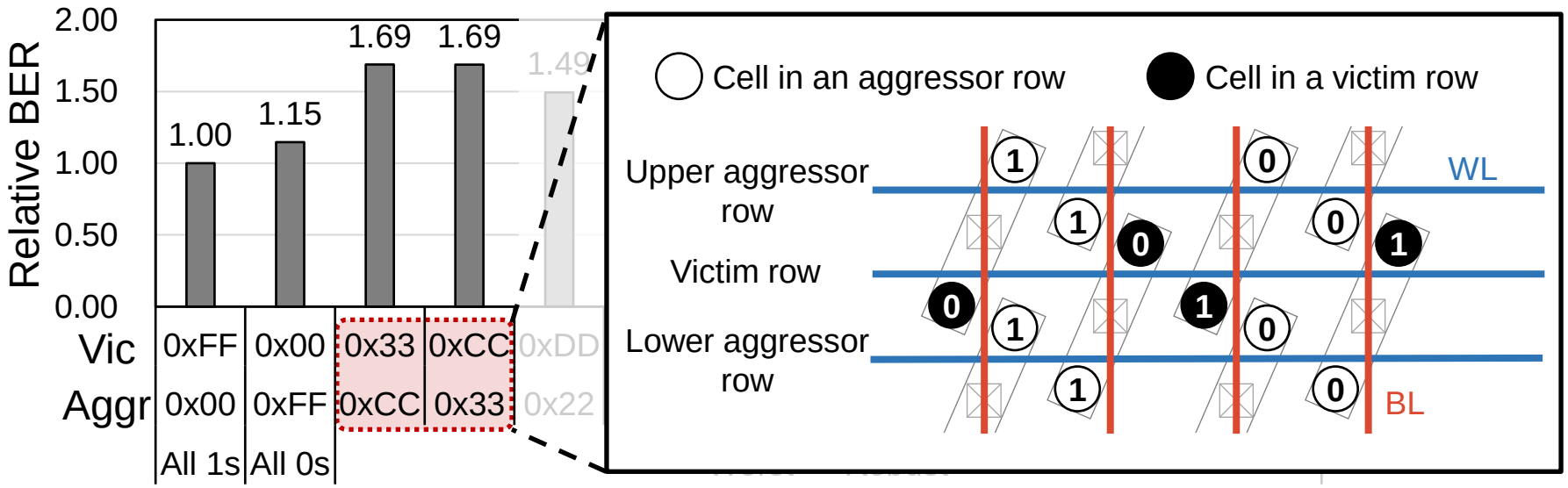
Microscopic Analysis – (2)

- **Adversarial Data Pattern for H_{cnt}**
  - $Vic_{-2,-1,1,2}$ and $Aggr_{-2,-1,0,1,2}$ hold the opposite value of Vic_0.



Microscopic Analysis – (2)

- **Adversarial Data Pattern for BER**

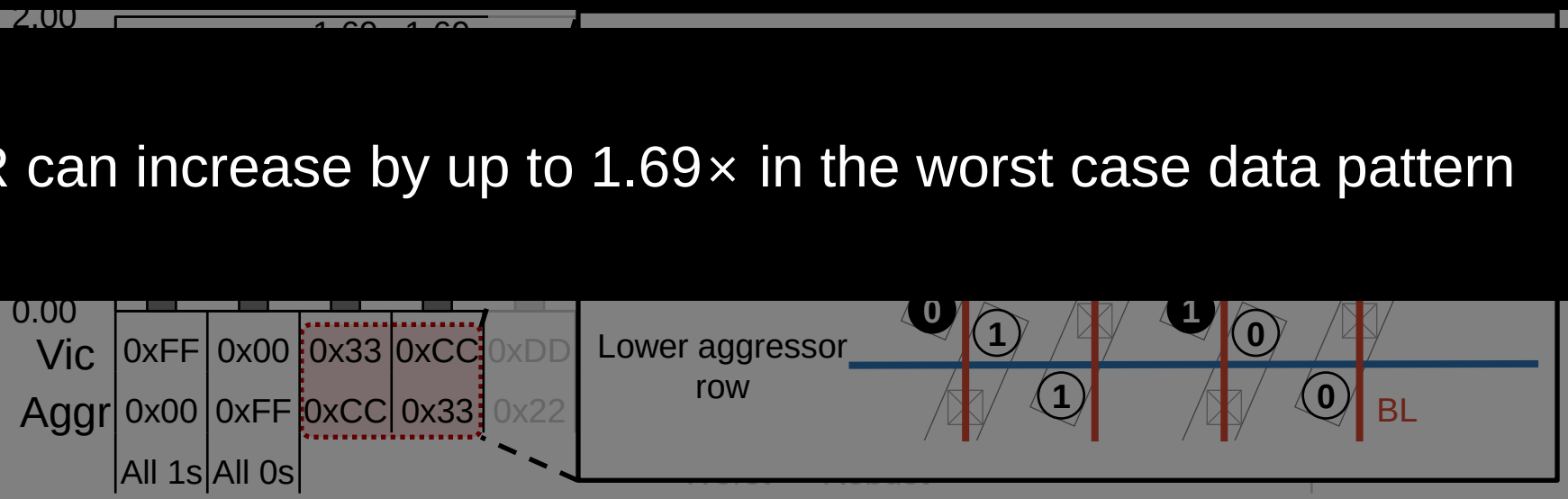


Microscopic Analysis – (2)

- **Adversarial Data Pattern for BER**

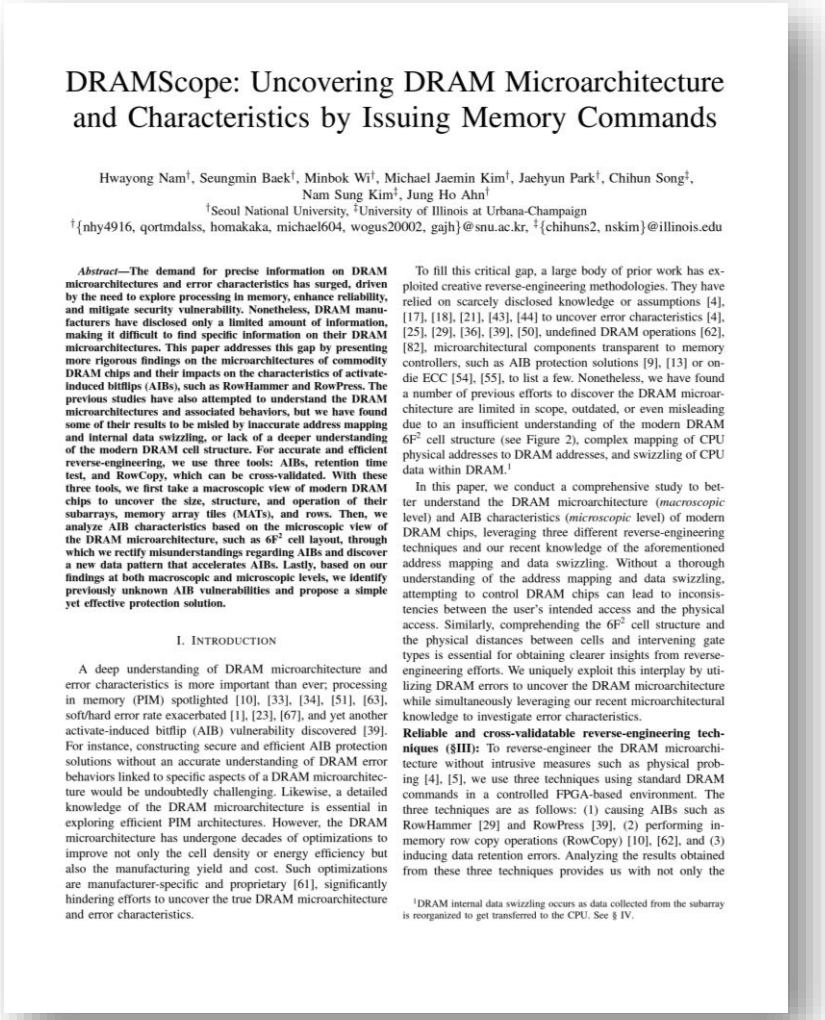
When all near cells from the victim cell has opposite value of victim cell's, victim cell is most vulnerable.

The BER can increase by up to 1.69× in the worst case data pattern



More Details

- Detailed processes of reverse-engineering
- 14 observations derived from analysis
- New vulnerabilities of AIBs and a simple yet effective protection solution



DRAMScope: Uncovering DRAM Microarchitecture and Characteristics by Issuing Memory Commands

Hwayong Nam†, Seungmin Baek†, Minbok Wi†, Michael Jaemin Kim†, Jaehyun Park†, Chihun Song‡, Nam Sung Kim‡, Jung Ho Ahn†
†Seoul National University, ‡University of Illinois at Urbana-Champaign
†{nhy4916, qortmdalss, homakaka, michael604, wogus20002, gajh}@snu.ac.kr, ‡{chihuns2, nskim}@illinois.edu

***Abstract*—The demand for precise information on DRAM microarchitectures and error characteristics has surged, driven by the need to explore processing in memory, enhance reliability, and mitigate security vulnerability. Nonetheless, DRAM manufacturers have disclosed only a limited amount of information, making it difficult to find specific information on their DRAM microarchitectures. This paper addresses this gap by presenting more rigorous findings on the microarchitectures of commodity DRAM chips and their impacts on the characteristics of activate-induced bitflips (AIBs), such as RowHammer and RowPress. The previous studies have also attempted to understand the DRAM microarchitectures and associated behaviors, but we have found some of their results to be misled by inaccurate address mapping and internal data swizzling, or lack of a deeper understanding of the modern DRAM cell structure. For accurate and efficient reverse-engineering, we use three tools: AIBs, retention time test, and RowCopy, which can be cross-validated. With these three tools, we first take a macroscopic view of modern DRAM chips to uncover the size, structure, and operation of their subarrays, memory array tiles (MATs), and rows. Then, we analyze AIB characteristics based on the microscopic view of the DRAM microarchitecture, such as $6F^2$ cell layout, through which we rectify misunderstandings regarding AIBs and discover a new data pattern that accelerates AIBs. Lastly, based on our findings at both macroscopic and microscopic levels, we identify previously unknown AIB vulnerabilities and propose a simple yet effective protection solution.**

I. Introduction

A deep understanding of DRAM microarchitecture and error characteristics is more important than ever; processing in memory (PIM) spotlighted [10], [33], [34], [51], [63], soft/hard error rate exacerbated [1], [23], [67], and yet another activate-induced bitflip (AIB) vulnerability discovered [39]. For instance, constructing secure and efficient AIB protection solutions without an accurate understanding of DRAM error behaviors linked to specific aspects of a DRAM microarchitecture would be undoubtedly challenging. Likewise, a detailed knowledge of the DRAM microarchitecture is essential in exploring efficient PIM architectures. However, the DRAM microarchitecture has undergone decades of optimizations to improve not only the cell density or energy efficiency but also the manufacturing yield and cost. Such optimizations are manufacturer-specific and proprietary [61], significantly hindering efforts to uncover the true DRAM microarchitecture and error characteristics.

To fill this critical gap, a large body of prior work has exploited creative reverse-engineering methodologies. They have relied on scarcely disclosed knowledge or assumptions [4], [17], [18], [21], [43], [44] to uncover error characteristics [4], [25], [29], [36], [39], [50], undefined DRAM operations [62], [82], microarchitectural components transparent to memory controllers, such as AIB protection solutions [9], [13] or on-die ECC [54], [55], to list a few. Nonetheless, we have found a number of previous efforts to discover the DRAM microarchitecture are limited in scope, outdated, or even misleading due to an insufficient understanding of the modern DRAM $6F^2$ cell structure (see Figure 2), complex mapping of CPU physical addresses to DRAM addresses, and swizzling of CPU data within DRAM.[1]

In this paper, we conduct a comprehensive study to better understand the DRAM microarchitecture (*macroscopic* level) and AIB characteristics (*microscopic* level) of modern DRAM chips, leveraging three different reverse-engineering techniques and our recent knowledge of the aforementioned address mapping and data swizzling. Without a thorough understanding of the address mapping and data swizzling, attempting to control DRAM chips can lead to inconsistencies between the user's intended access and the physical access. Similarly, comprehending the $6F^2$ cell structure and the physical distances between cells and intervening gate types is essential for obtaining clearer insights from reverse-engineering efforts. We uniquely exploit this interplay by utilizing DRAM errors to uncover the DRAM microarchitecture while simultaneously leveraging our recent microarchitectural knowledge to investigate error characteristics.

Reliable and cross-validatable reverse-engineering techniques (§III): To reverse-engineer the DRAM microarchitecture without intrusive measures such as physical probing [4], [5], we use three techniques using standard DRAM commands in a controlled FPGA-based environment. The three techniques are as follows: (1) causing AIBs such as RowHammer [29] and RowPress [39], (2) performing in-memory row copy operations (RowCopy) [10], [62], and (3) inducing data retention errors. Analyzing the results obtained from these three techniques provides us with not only the

[1]DRAM internal data swizzling occurs as data collected from the subarray is reorganized to get transferred to the CPU. See § IV.

Summary

Summary

1) We have reliably revealed **the DRAM microarchitectures** and **activate-induced bitflip (AIBs) characteristics** using commercial DRAM chips.
2) We discovered undisclosed **DRAM microarchitectures** and associated behaviors in **subarray, row, and MAT levels**.
3) We showed that **precise mapping information** of DRAM modules and chips is necessary to accurately analyze the **AIB characteristics**.
4) By considering the DRAM's microscopic aspect, such as **DRAM $6F^2$ cell structure**, we also identified the **data pattern dependency** on the **AIB phenomenon**.

Thank you!

Question?