

Improving 3D NAND Flash Memory Lifetime by Tolerating Early Retention Loss and Process Variation

Yixin Luo Saugata Ghose Yu Cai Erich F. Haratsch Onur Mutlu

Carnegie Mellon

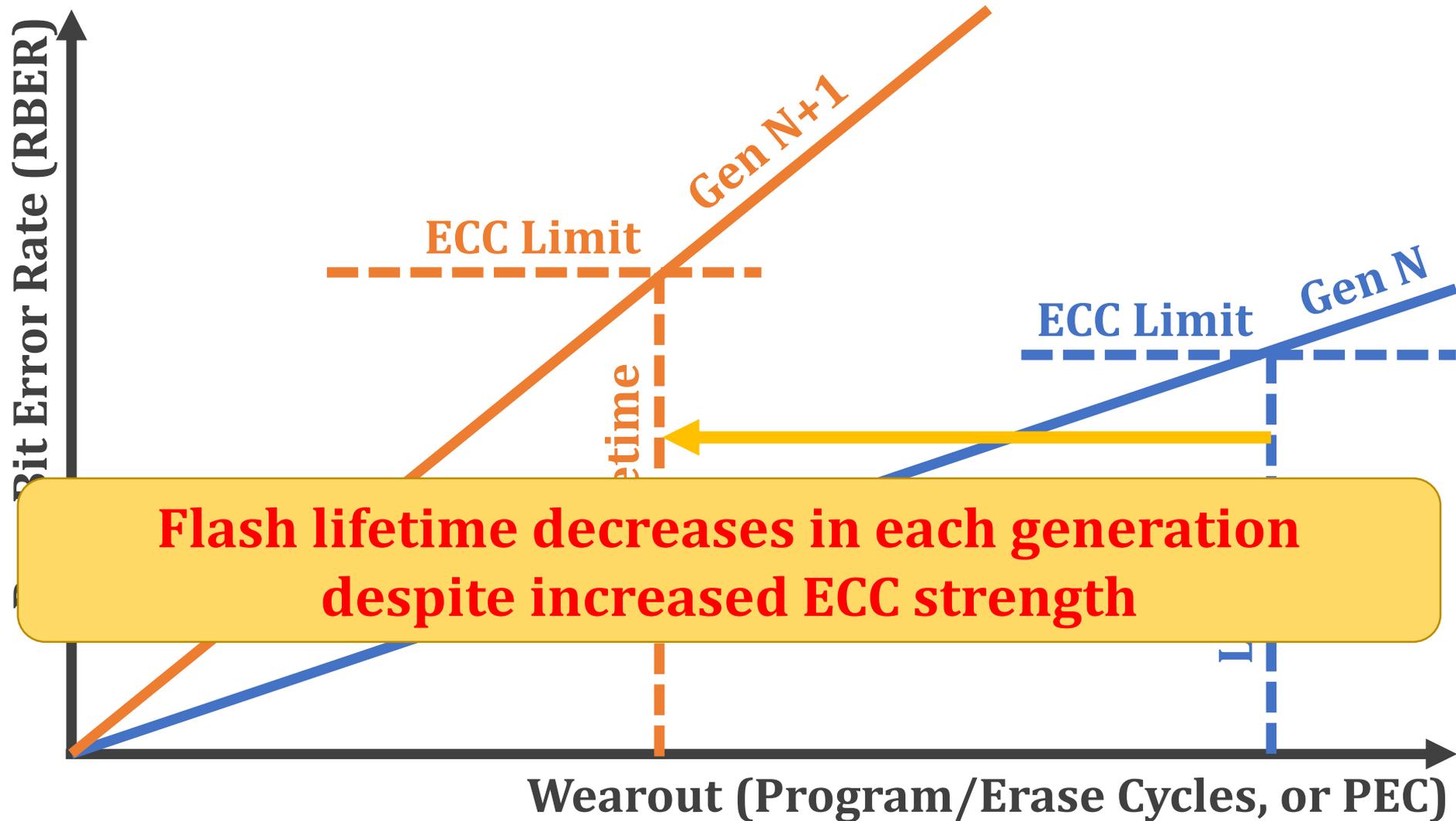


ETH zürich

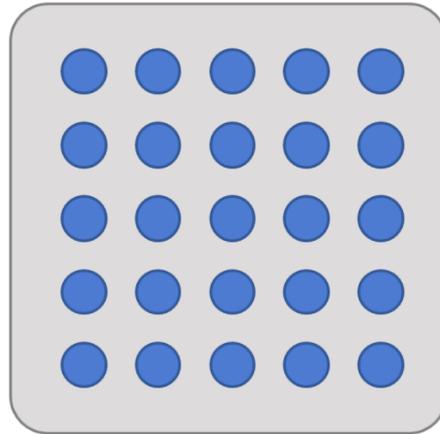
SAFARI



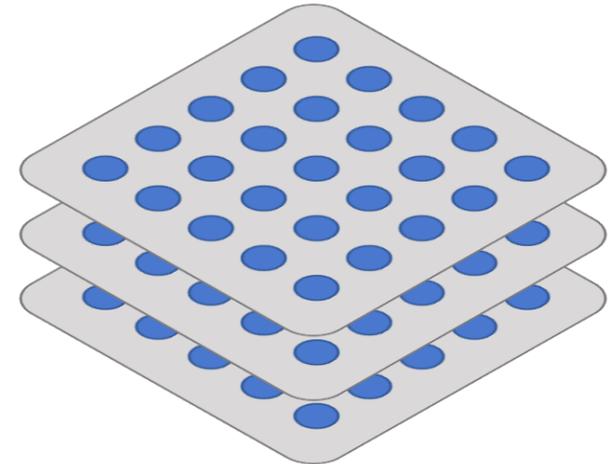
NAND Flash Memory Lifetime Problem



Planar vs. 3D NAND Flash Memory



**Planar NAND
Flash Memory**



**3D NAND
Flash Memory**

Scaling

Reduce flash cell size,
Reduce distance b/w cells

Increase # of layers

Reliability

Scaling hurts reliability

Not well studied!

Executive Summary

- Problem: 3D NAND error characteristics are **not well studied**
- Goal: *Understand & mitigate* 3D NAND errors to improve lifetime
- **Contribution 1: Characterize** real 3D NAND flash chips
 - **Process variation:** *21× error rate difference across layers*
 - **Early retention loss:** *Error rate increases by 10× after 3 hours*
 - **Retention interference:** *Not observed before in planar NAND*
- **Contribution 2: Model** RBER and threshold voltage
 - *RBER (raw bit error rate) variation model*
 - *Retention loss model*
- **Contribution 3: Mitigate** 3D NAND flash errors
 - **LaVAR:** *Layer Variation Aware Reading*
 - **LI-RAID:** *Layer-Interleaved RAID*
 - **ReMAR:** *Retention Model Aware Reading*
 - *Improve flash lifetime by 1.85× or reduce ECC overhead by 78.9%*

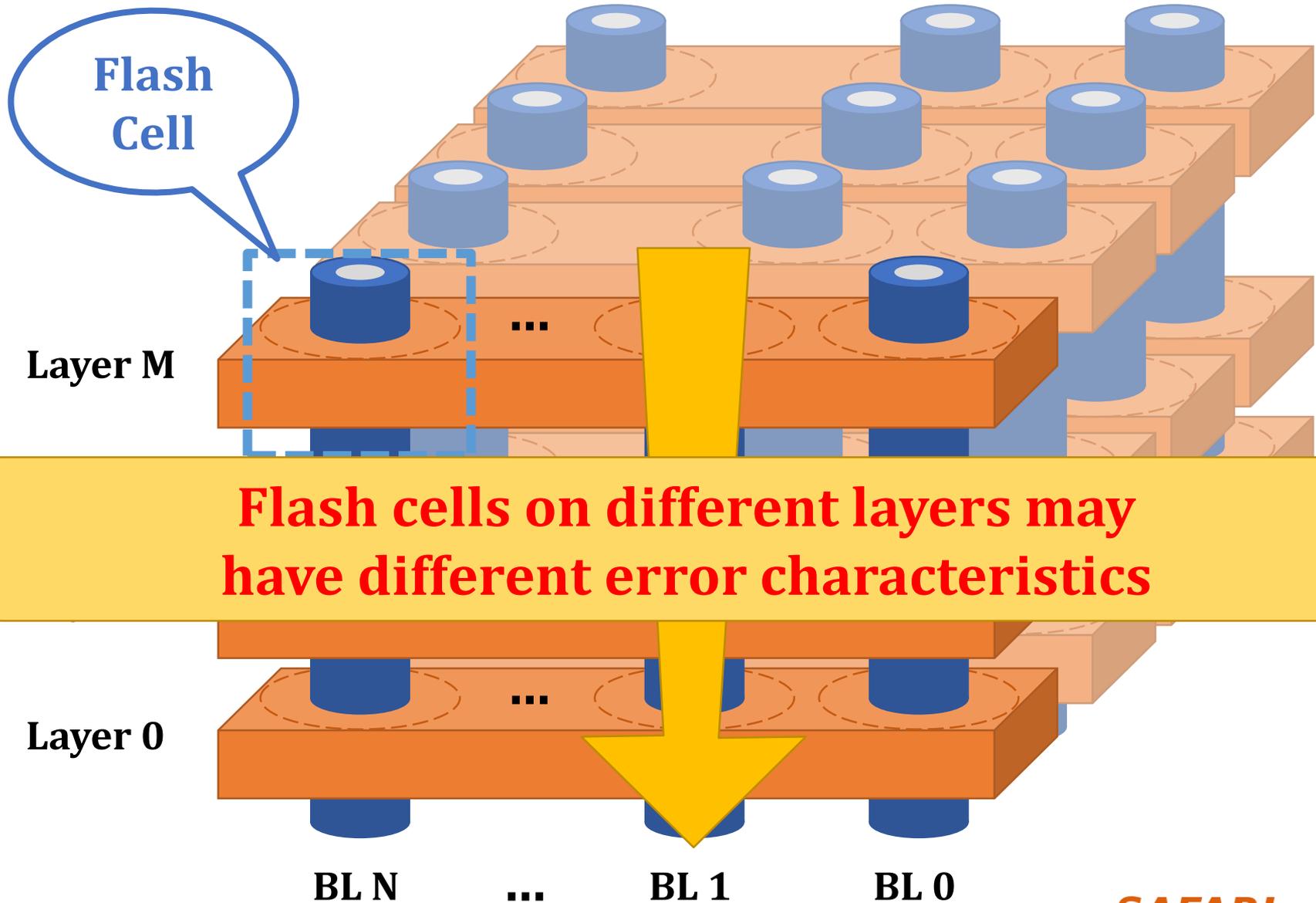
Agenda

- Background & Introduction
- Contribution 1: Characterize real 3D NAND flash chips
- Contribution 2: Model RBER and threshold voltage
- Contribution 3: Mitigate 3D NAND flash errors
- Conclusion

Agenda

- Background & Introduction
- **Contribution 1: Characterize real 3D NAND flash chips**
 - Process variation
 - Early retention loss
 - Retention interference
- Contribution 2: Model RBER and threshold voltage
- Contribution 3: Mitigate 3D NAND flash errors
- Conclusion

Process Variation Across Layers



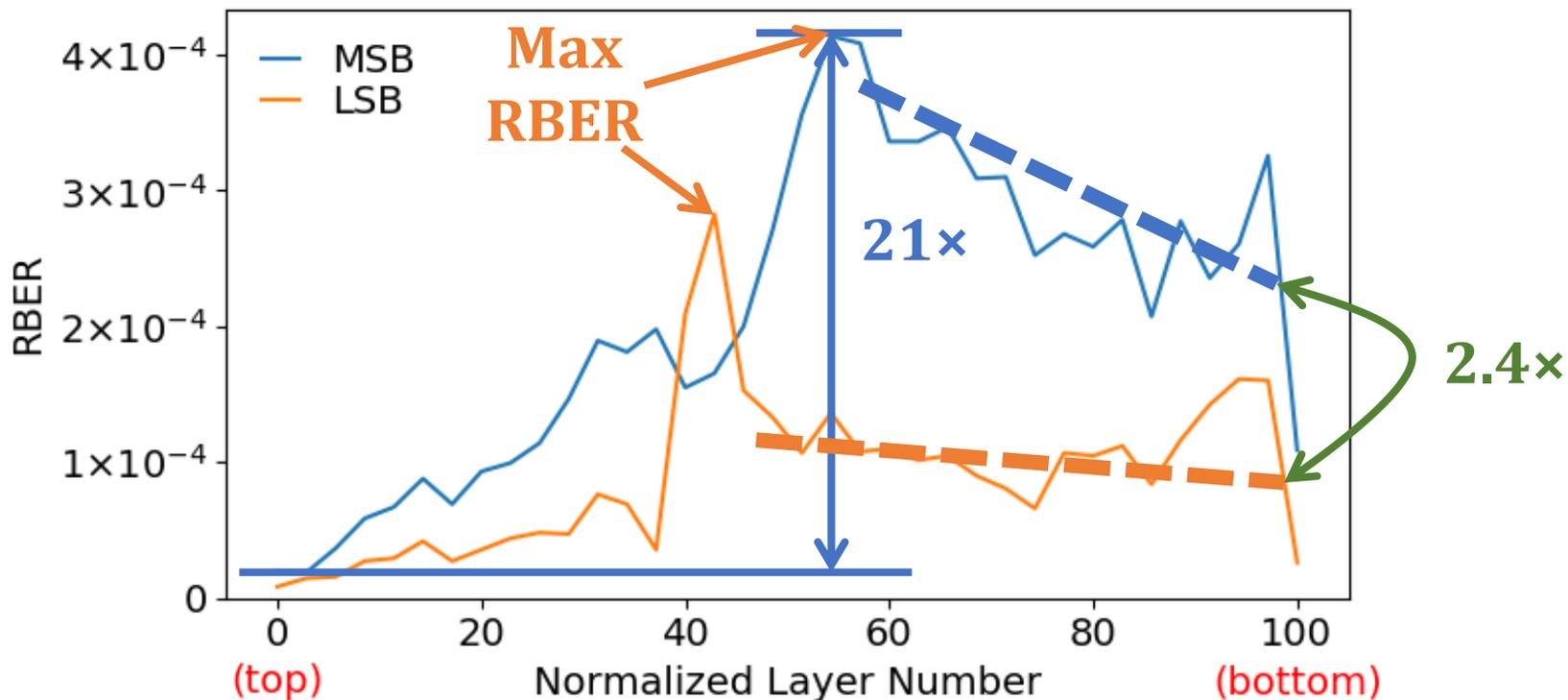
Flash cells on different layers may have different error characteristics

Characterization Methodology

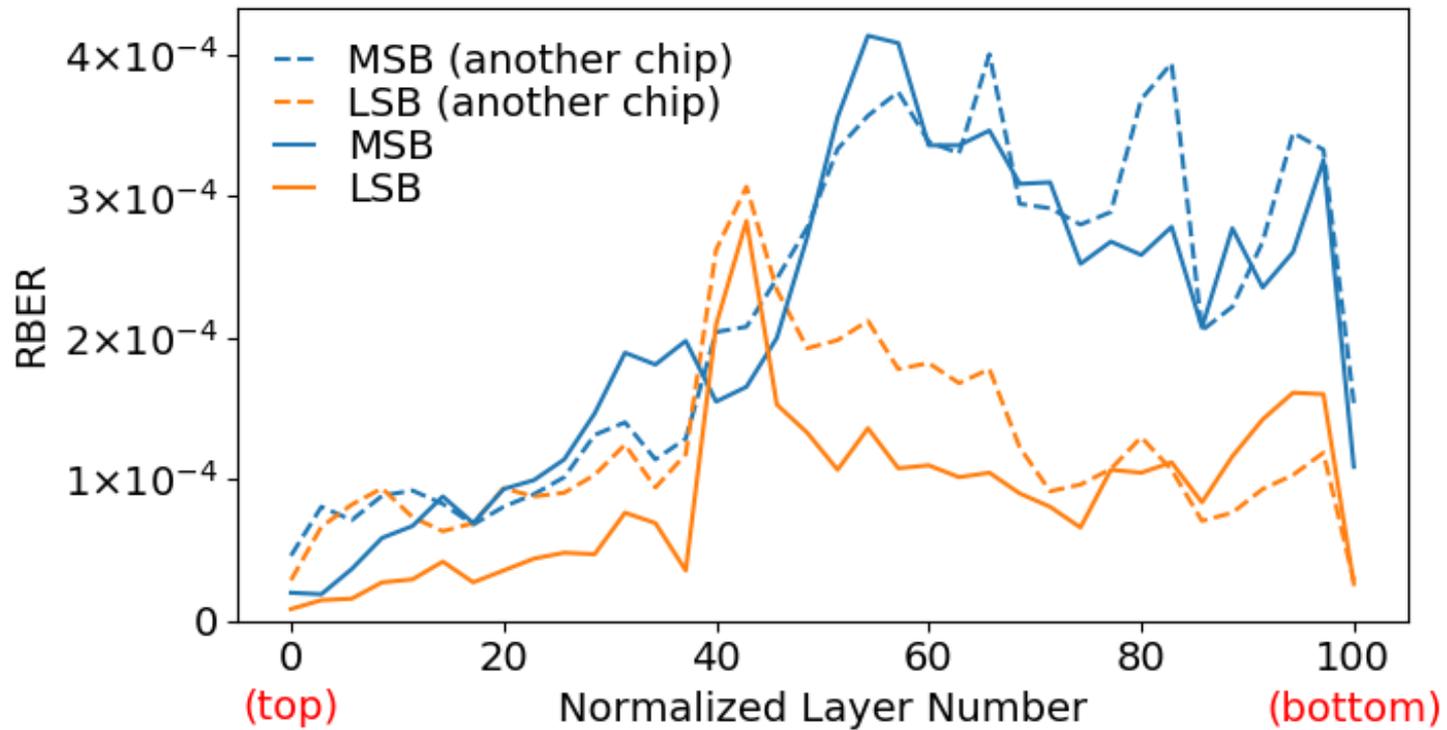
- **Modified firmware version in the flash controller**
 - Controls the read reference voltage of the flash chip
 - Bypasses ECC to get raw data (with raw bit errors)
- **Analysis and post-processing of the data on the server**



Layer-to-Layer Process Variation



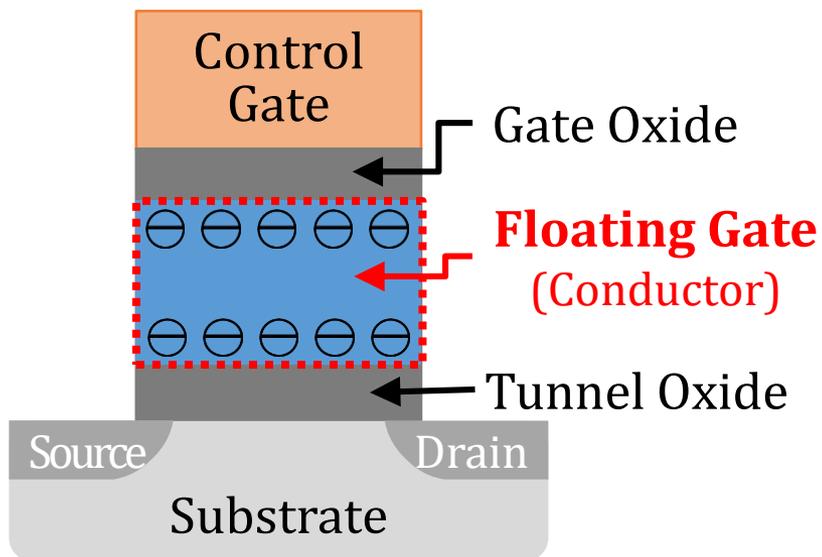
Layer-to-Layer Process Variation



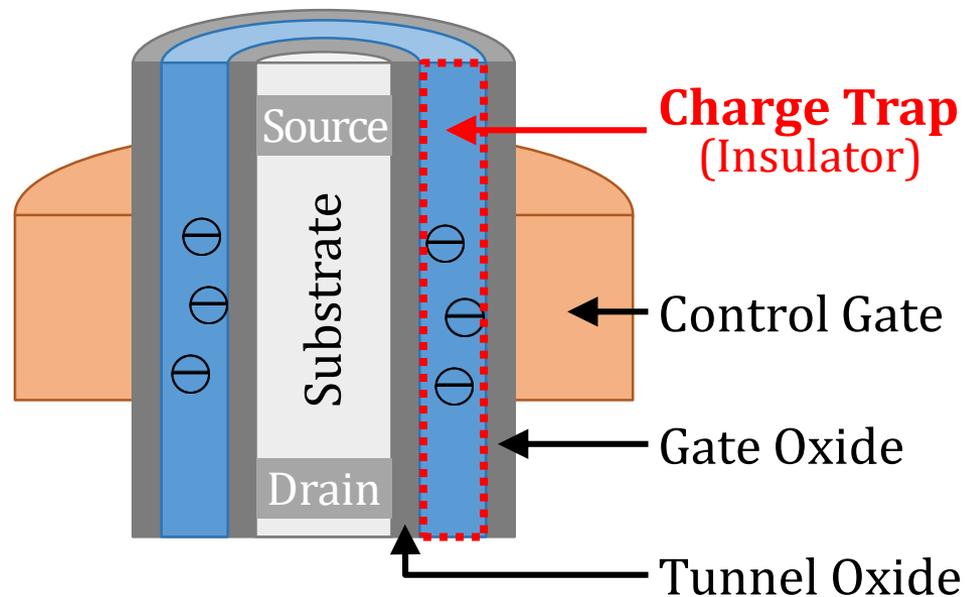
**Large RBER variation
across layers and LSB-MSB pages**

Retention Loss Phenomenon

Planar NAND Cell

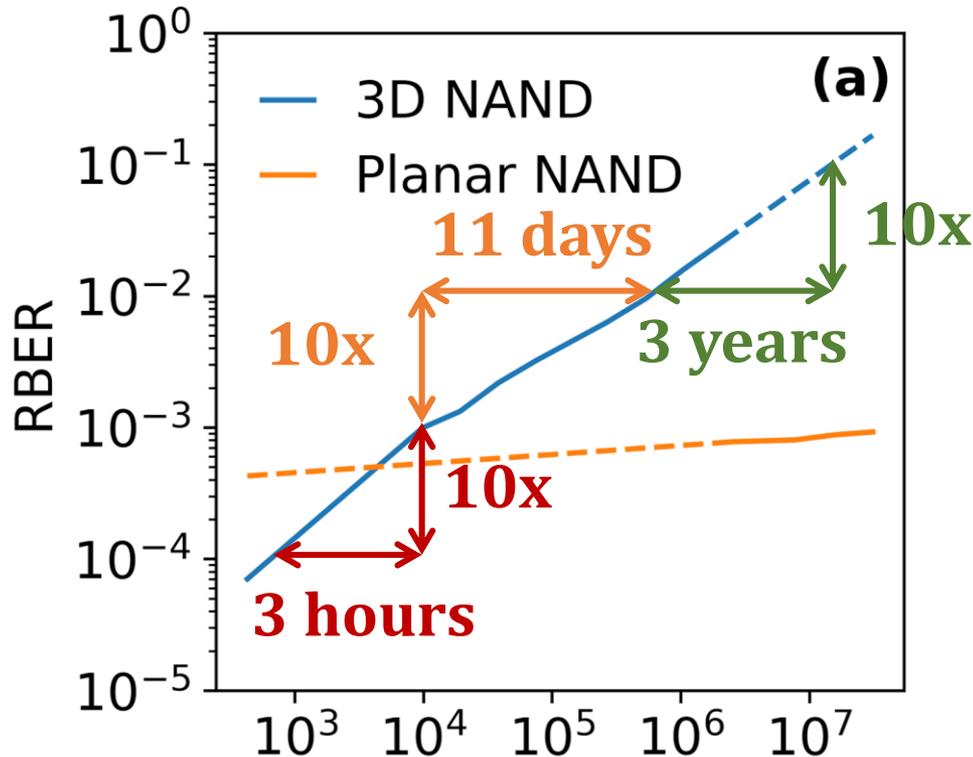


3D NAND Cell



**Most dominant type of error in planar NAND.
Is this true for 3D NAND as well?**

Early Retention Loss



Retention errors increase quickly immediately after programming

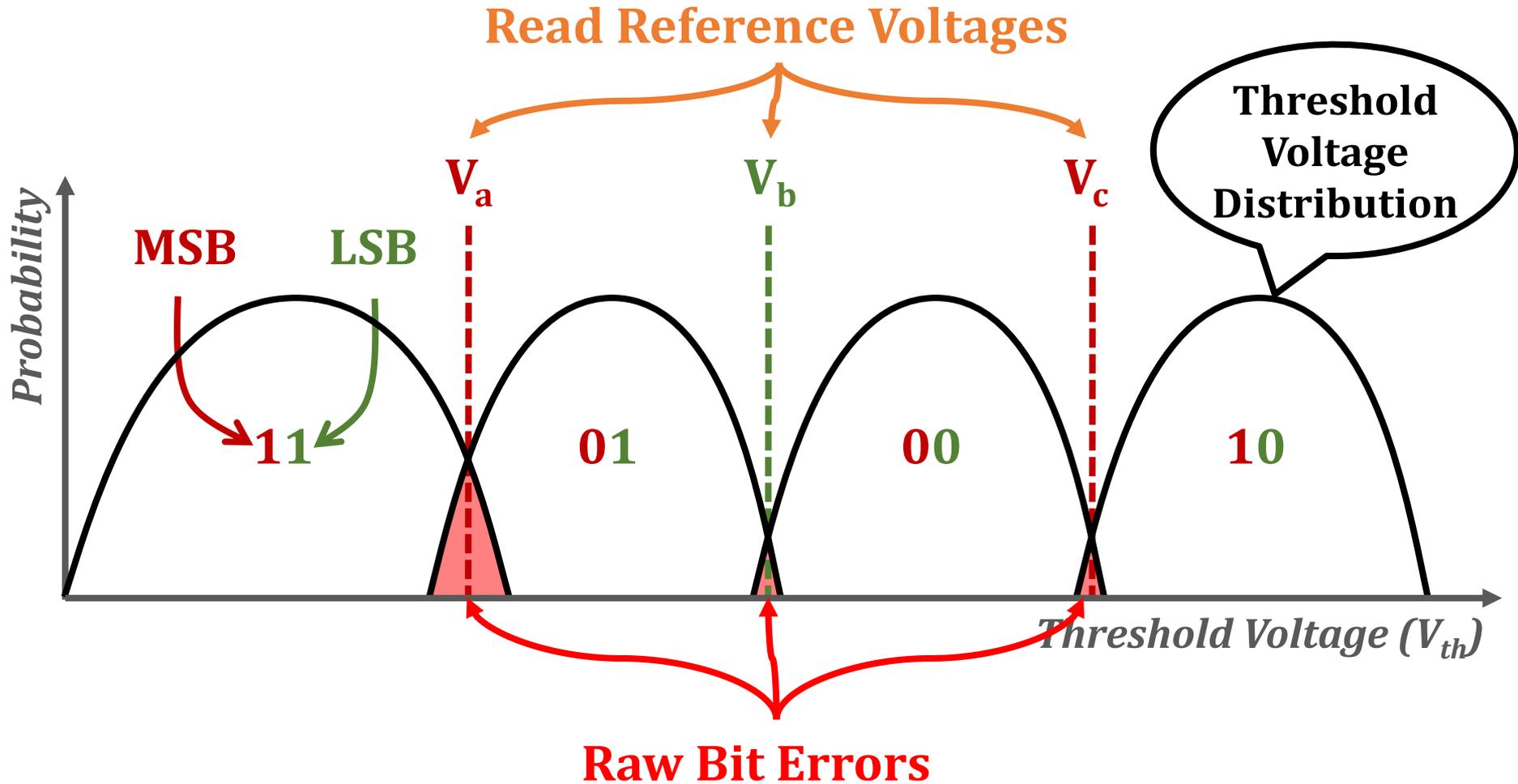
Characterization Summary

- **Layer-to-layer process variation**
 - Large RBER variation across layers and LSB-MSB pages
 - → Need new mechanisms to tolerate RBER variation!
- **Early retention loss**
 - RBER increases quickly after programming
 - → Need new mechanisms to tolerate retention errors!
- **Retention interference**
 - Amount of retention loss correlated with neighbor cells' states
 - → Need new mechanisms to tolerate retention interference!
- **More *threshold voltage* and *RBER* results in the paper:**
3D NAND P/E cycling, program interference, read disturb, read variation, bitline-to-bitline process variation
- **Our approach** based on insights developed via our experimental characterization: Develop **error models**, and build online **error mitigation mechanisms** using the models

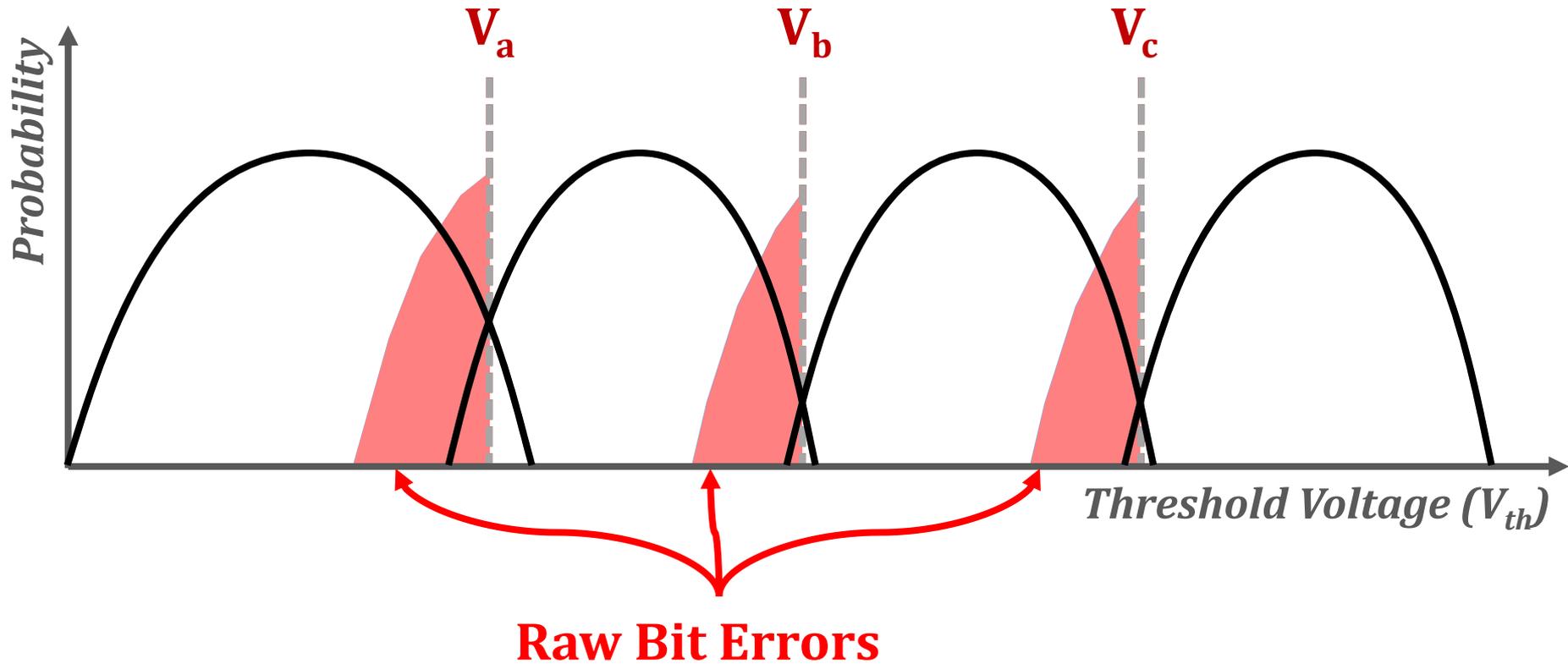
Agenda

- Background & Introduction
- Contribution 1: Characterize real 3D NAND flash chips
- Contribution 2: Model RBER and threshold voltage
 - Retention loss model
 - RBER variation model
- Contribution 3: Mitigate 3D NAND flash errors
- Conclusion

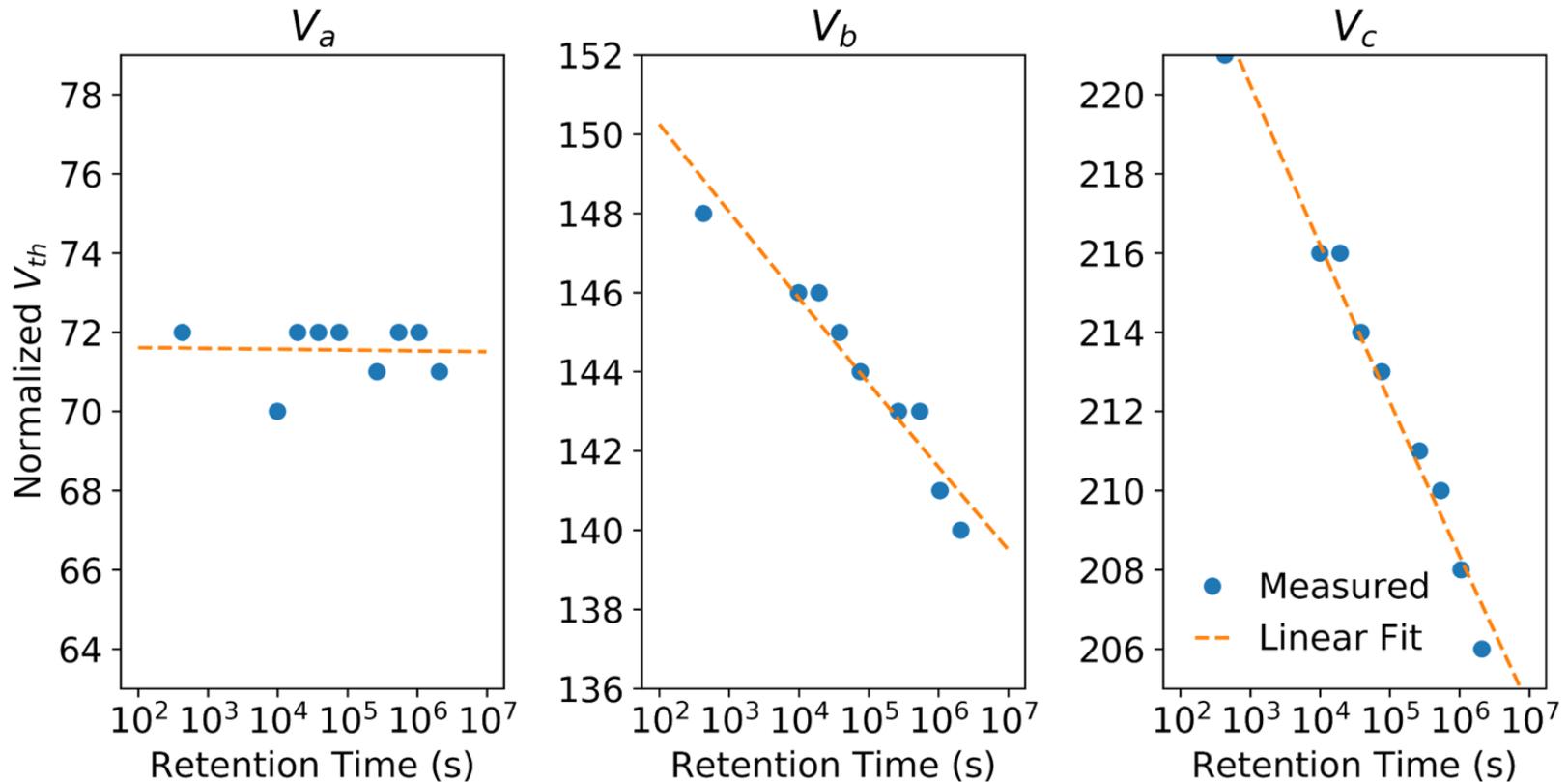
What Do We Model?



Optimal Read Reference Voltage



Retention Loss Model

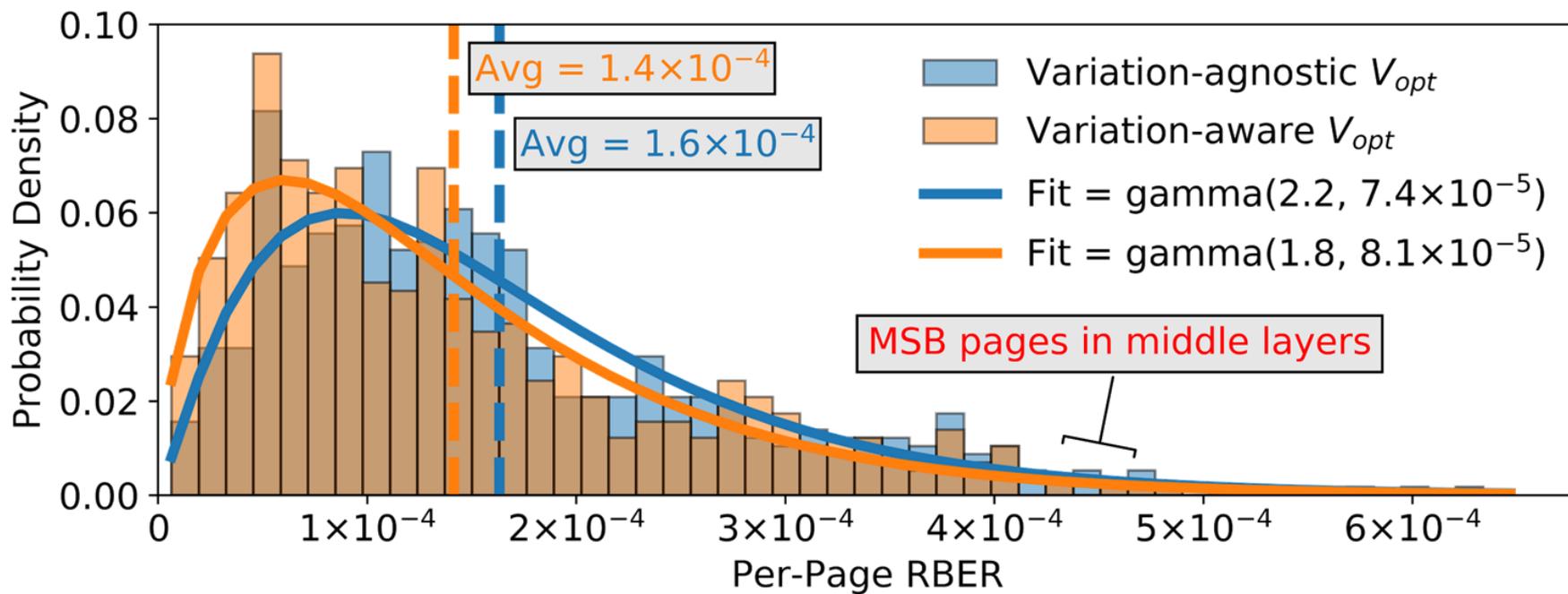


Early retention loss can be modeled as a simple linear function of $\log(\text{retention time})$

Retention Loss Model

- Goal: Develop a simple linear model that can be used online
- Models
 - Optimal read reference voltage (V_b and V_c)
 - Raw bit error rate ($\log(RBER)$)
 - Mean and standard deviation of threshold voltage distribution (μ and σ)
- As a function of
 - Retention time ($\log(t)$)
 - P/E cycle count (PEC)
- e.g., $V_{opt} = (\alpha \times PEC + \beta) \times \log(t) + \gamma \times PEC + \delta$
- Model error <1 step for V_b and V_c
- Adjusted $R^2 > 89\%$

RBER Variation Model



Variation-agnostic V_{opt}

- Same V_{ref} for all layers optimized for the entire block

RBER distribution follows gamma distribution

KL-divergence error = 0.09

Agenda

- Background & Introduction
- Contribution 1: Characterize real 3D NAND flash chips
- Contribution 2: Model RBER and threshold voltage
- **Contribution 3: Mitigate 3D NAND flash errors**
 - **LaVAR: Layer Variation Aware Reading**
 - **LI-RAID: Layer-Interleaved RAID**
 - **ReMAR: Retention Model Aware Reading**
- Conclusion

LaVAR: Layer Variation Aware Reading

- **Layer-to-layer process variation**
 - Error characteristics are different in each layer
- **Goal:** Adjust read reference voltage **for each layer**
- **Key Idea:** Learn a **voltage offset (Offset)** for each layer
 - $V_{opt}^{Layer\ aware} = V_{opt}^{Layer\ agnostic} + Offset$
- **Mechanism**
 - **Offset:** Learned once for each chip & stored in a table
 - *Uses (2 × Layers) Bytes memory per chip*
 - $V_{opt}^{Layer\ agnostic}$: Predicted by any existing V_{opt} model
 - *E.g., ReMAR [Luo+Sigmetrics'18], HeatWatch [Luo+HPCA'18], OFCM [Luo+JSAC'16], ARVT [Papandreou+GLSVLSI'14]*
- Reduces RBER on average by **43%**
(based on our characterization data)

LI-RAID: Layer-Interleaved RAID

- **Layer-to-layer process variation**
 - Worst-case RBER much higher than average RBER
- **Goal:** Significantly reduce worst-case RBER

- **Key Idea**
 - Group flash pages on *less reliable layers* with pages on *more reliable layers*
 - Group *MSB pages* with *LSB pages*

- **Mechanism**
 - Reorganize RAID layout to eliminate worst-case RBER
 - **<0.4%** storage overhead

Conventional RAID

<i>Wordline #</i>	<i>Layer #</i>	<i>Page</i>	Chip 0	Chip 1	Chip 2	Chip 3
0	0	MSB	Group 0	Group 0	Group 0	Group 0
0	0	LSB	Group 1	Group 1	Group 1	Group 1
1	1	MSB	Group 2	Group 2	Group 2	Group 2
1	1	LSB	Group 3	Group 3	Group 3	Group 3
2	2	MSB	Group 4	Group 4	Group 4	Group 4
2	2	LSB	Group 5	Group 5	Group 5	Group 5
3	3	MSB	Group 6	Group 6	Group 6	Group 6
3	3	LSB	Group 7	Group 7	Group 7	Group 7

**Worst-case RBER in any layer
limits the lifetime of conventional RAID**

LI-RAID: Layer-Interleaved RAID

<i>Wordline #</i>	<i>Layer #</i>	<i>Page</i>	Chip 0	Chip 1	Chip 2	Chip 3
0	0	MSB	Group 0	Blank	Group 4	Group 3
0	0	LSB	Group 1	Blank	Group 5	Group 2
1	1	MSB	Group 2	Group 1	Blank	Group 5
1	1	LSB	Group 3	Group 0	Blank	Group 4
2	2	MSB	Group 4	Group 3	Group 0	Blank
2	2	LSB	Group 5	Group 2	Group 1	Blank
3	3	MSB	Blank	Group 5	Group 2	Group 1
3	3	LSB	Blank	Group 4	Group 3	Group 0

Any page with worst-case RBER can be corrected by other reliable pages in the RAID group

LI-RAID: Layer-Interleaved RAID

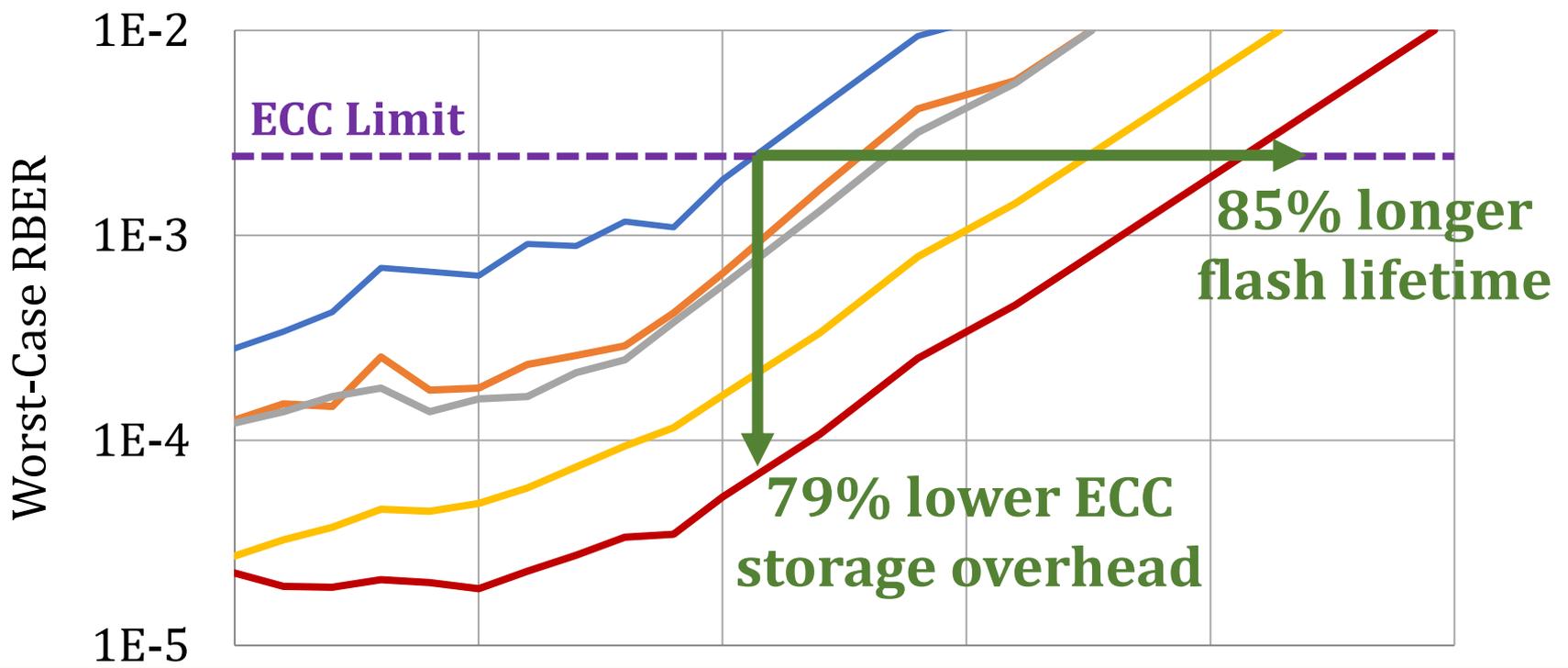
- **Layer-to-layer process variation**
 - Worst-case RBER much higher than average RBER
- **Goal:** Significantly reduce worst-case RBER
- **Key Idea**
 - Group flash pages on *less reliable layers* with pages on *more reliable layers*
 - Group *MSB pages* with *LSB pages*
- **Mechanism**
 - Reorganize RAID layout to eliminate worst-case RBER
 - **<0.8% storage overhead**
- Reduces worst-case RBER by **66.9%**
(based on our characterization data)

ReMAR: Retention Model Aware Reading

- **Early retention loss**
 - Threshold voltage shifts quickly after programming
- **Goal: Adjust read reference voltages based on retention loss**
- **Key Idea:** Learn and use a retention loss model online
- **Mechanism**
 - Periodically characterize and learn retention loss model online
 - Retention time = Read timestamp - Write timestamp
 - *Uses 800 KB memory to store program time of each block*
 - Predict retention-aware V_{opt} using the model
- Reduces RBER on average by **51.9%**
(based on our characterization data)

Impact on System Reliability

— Baseline — State-of-the-art — LaVAR
— LaVAR + LI-RAID — This Work



LaVAR, LI-RAID, and ReMAR improve flash lifetime or reduce ECC overhead significantly

Error Mitigation Techniques Summary

- **LaVAR: Layer Variation Aware Reading**
 - Learn a V_{opt} offset for each layer and apply *layer-aware V_{opt}*
- **LI-RAID: Layer-Interleaved RAID**
 - Group flash pages on *less reliable layers* with pages on *more reliable layers*
 - Group *MSB pages* with *LSB pages*
- **ReMAR: Retention Model Aware Reading**
 - Learn retention loss model and apply *retention-aware V_{opt}*
- **Benefits:**
 - Improve flash lifetime by **1.85×** or reduce ECC overhead by **78.9%**
- **ReNAC (in paper):** Reread a failed page using V_{opt} based on the *retention interference* induced by neighbor cell

Agenda

- Background & Introduction
- Contribution 1: Characterize real 3D NAND flash chips
- Contribution 2: Model RBER and threshold voltage
- Contribution 3: Mitigate 3D NAND flash errors
- **Conclusion**

Conclusion

- Problem: 3D NAND error characteristics are **not well studied**
- Goal: *Understand & mitigate* 3D NAND errors to improve lifetime
- **Contribution 1: Characterize** real 3D NAND flash chips
 - **Process variation:** *21× error rate difference across layers*
 - **Early retention loss:** *Error rate increases by 10× after 3 hours*
 - **Retention interference:** *Not observed before in planar NAND*
- **Contribution 2: Model** RBER and threshold voltage
 - *RBER (raw bit error rate) variation model*
 - *Retention loss model*
- **Contribution 3: Mitigate** 3D NAND flash errors
 - **LaVAR:** *Layer Variation Aware Reading*
 - **LI-RAID:** *Layer-Interleaved RAID*
 - **ReMAR:** *Retention Model Aware Reading*
 - *Improve flash lifetime by 1.85× or reduce ECC overhead by 78.9%*

Improving 3D NAND Flash Memory Lifetime by Tolerating Early Retention Loss and Process Variation

Yixin Luo Saugata Ghose Yu Cai Erich F. Haratsch Onur Mutlu

Carnegie Mellon



ETH zürich

SAFARI



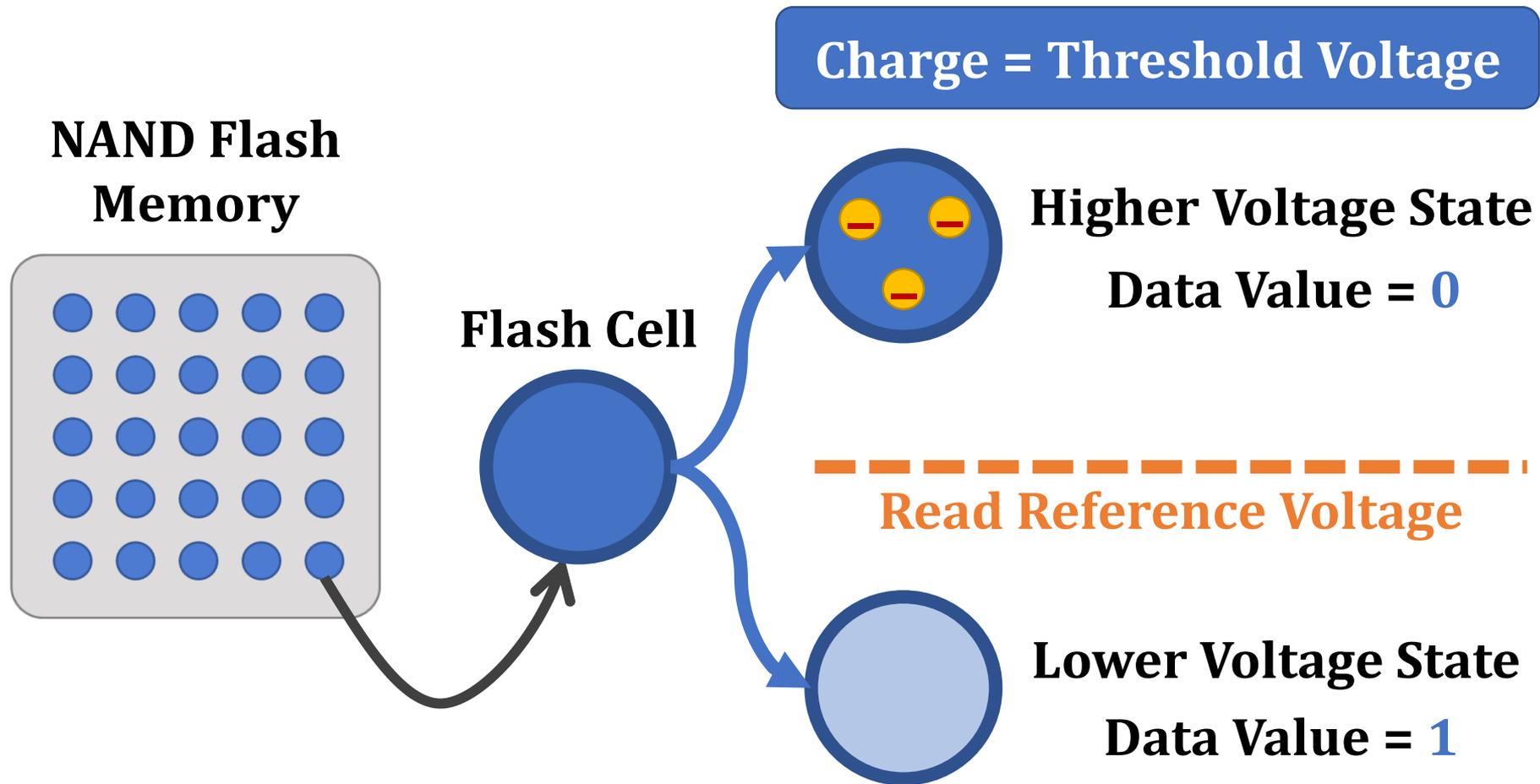
Backup Slides

LI-RAID: Layer-Interleaved RAID

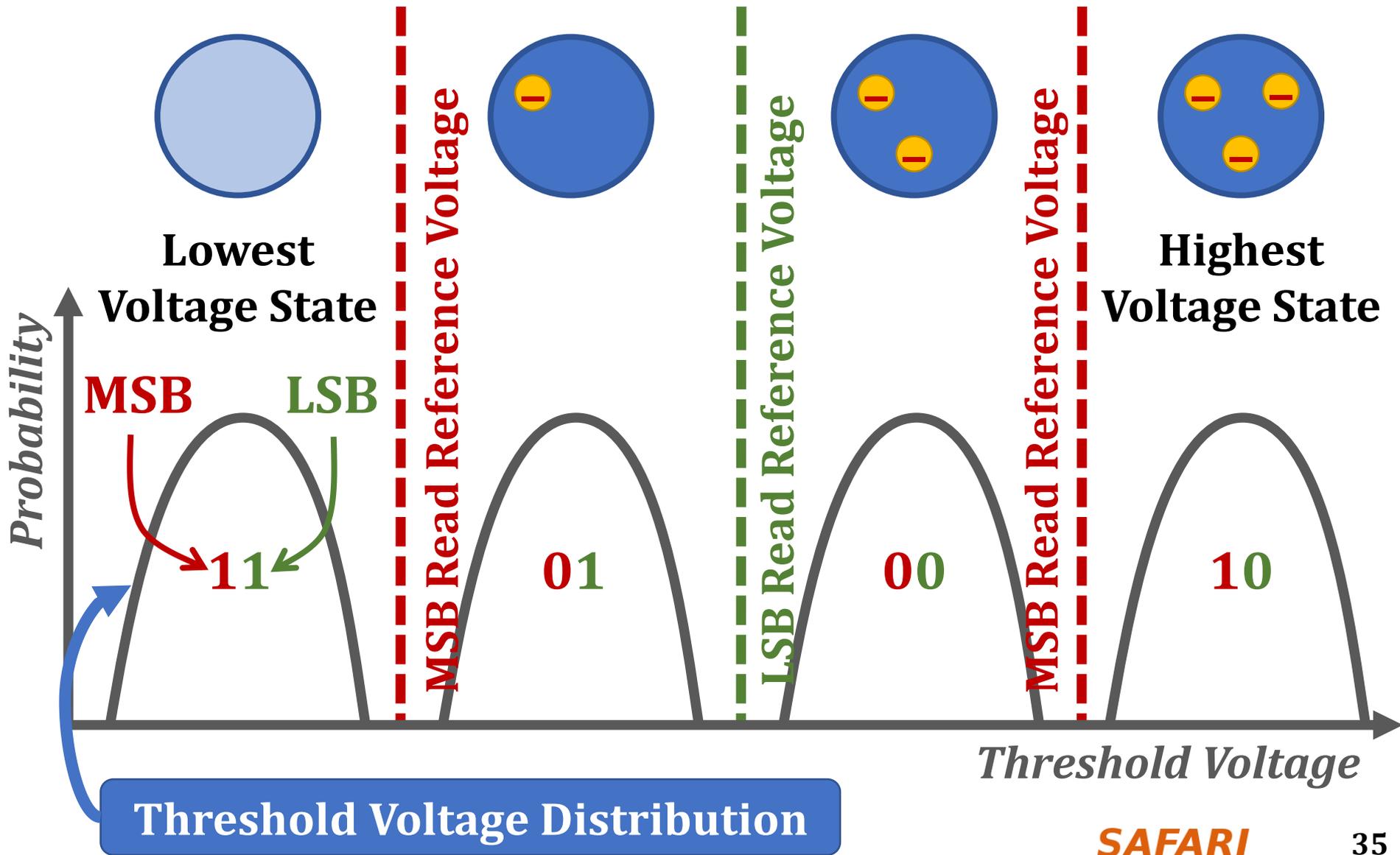
<i>Wordline #</i>	<i>Layer #</i>	<i>Page</i>	Chip 0	Chip 1	Chip 2	Chip 3
0	0	MSB	Group 0	(Group 6)	Group 4	Group 3
0	0	LSB	Group 1	(Group 7)	Group 5	Group 2
1	1	MSB	Group 2	Group 1	Blank	Group 5
1	1	LSB	Group 3	Group 0	Blank	Group 4
2	2	MSB	Group 4	Group 3	Group 0	Blank
2	2	LSB	Group 5	Group 2	Group 1	Blank
3	3	MSB	Blank	Group 5	Group 2	Group 1
3	3	LSB	Blank	Group 4	Group 3	Group 0

**Violating program sequence (in-order from top to bottom)
Groups 0&1 vulnerable to program interference by Groups 2&3**

How Does NAND Flash Memory Work?

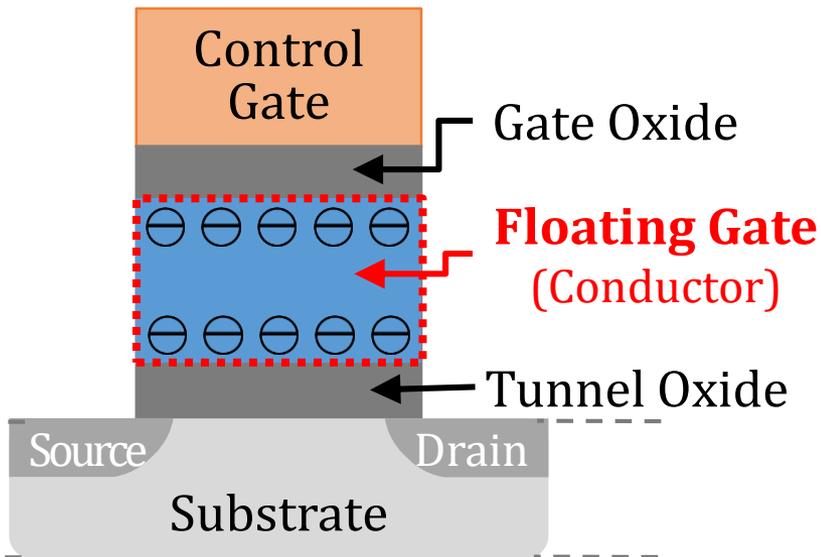


MLC Threshold Voltage Distribution

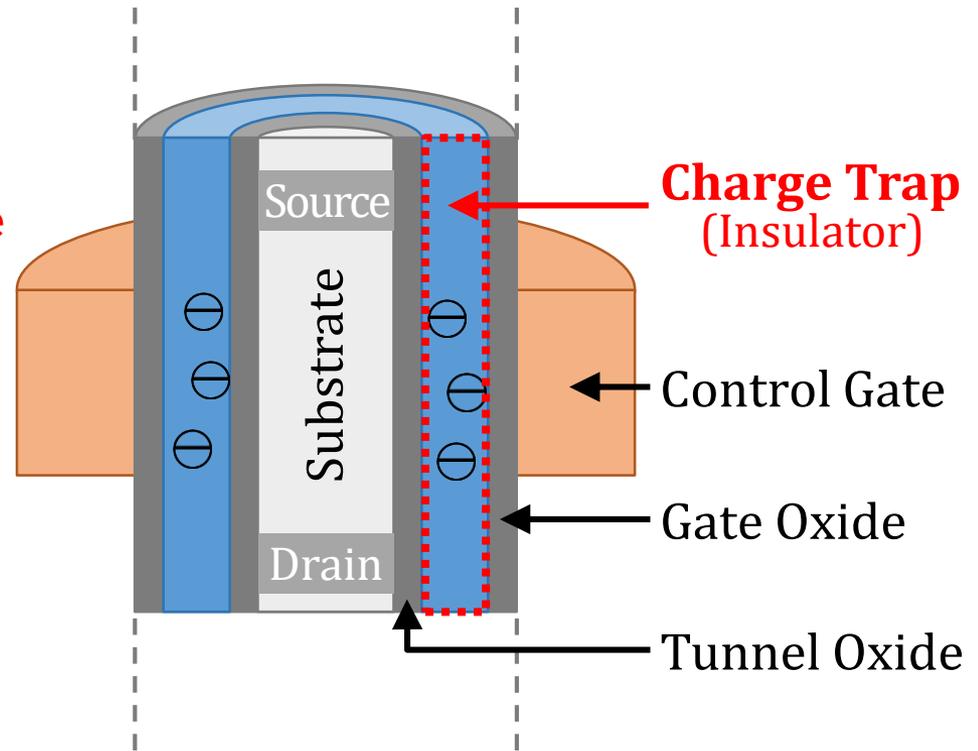


3D NAND Flash Cell

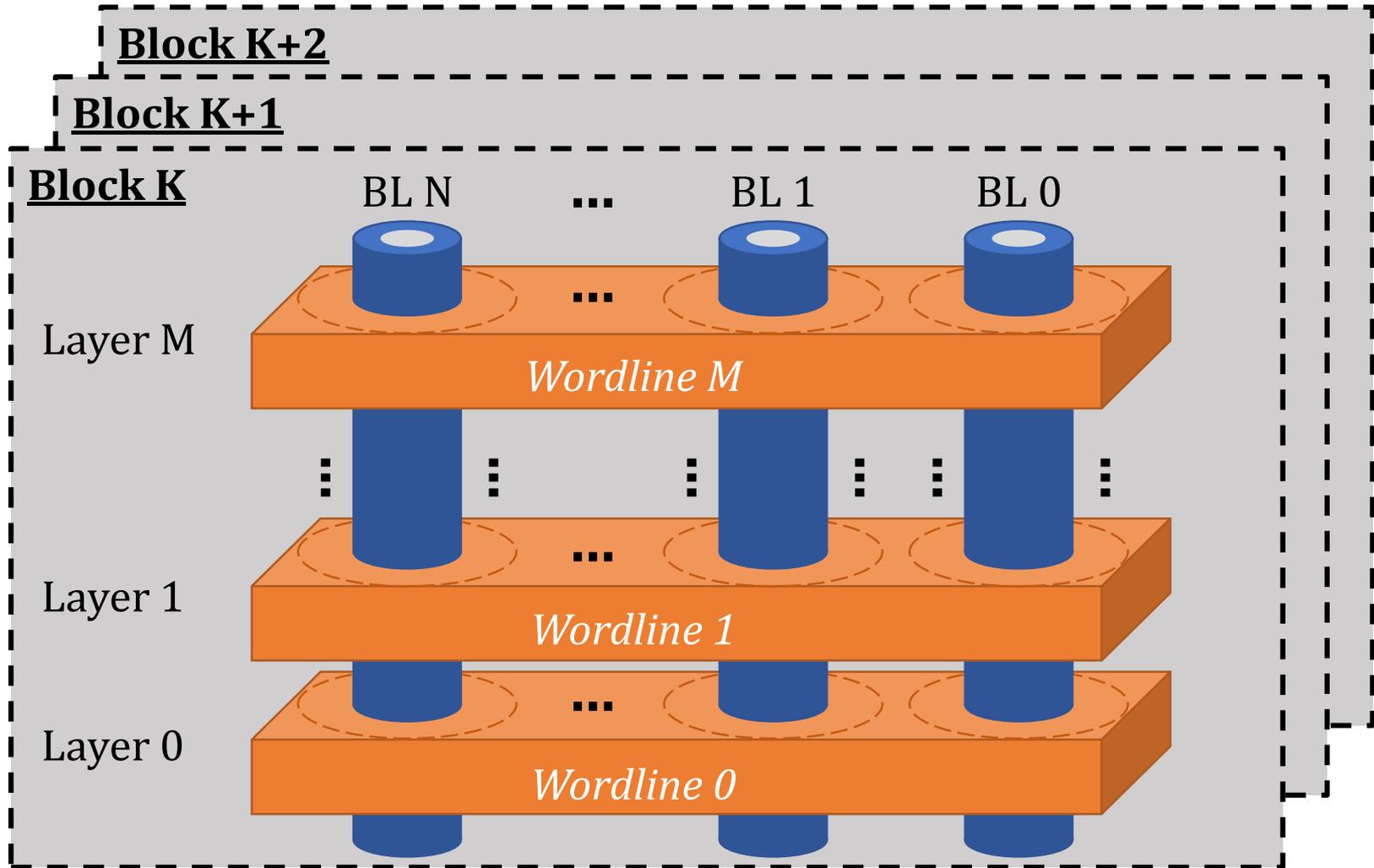
Floating Gate Cell



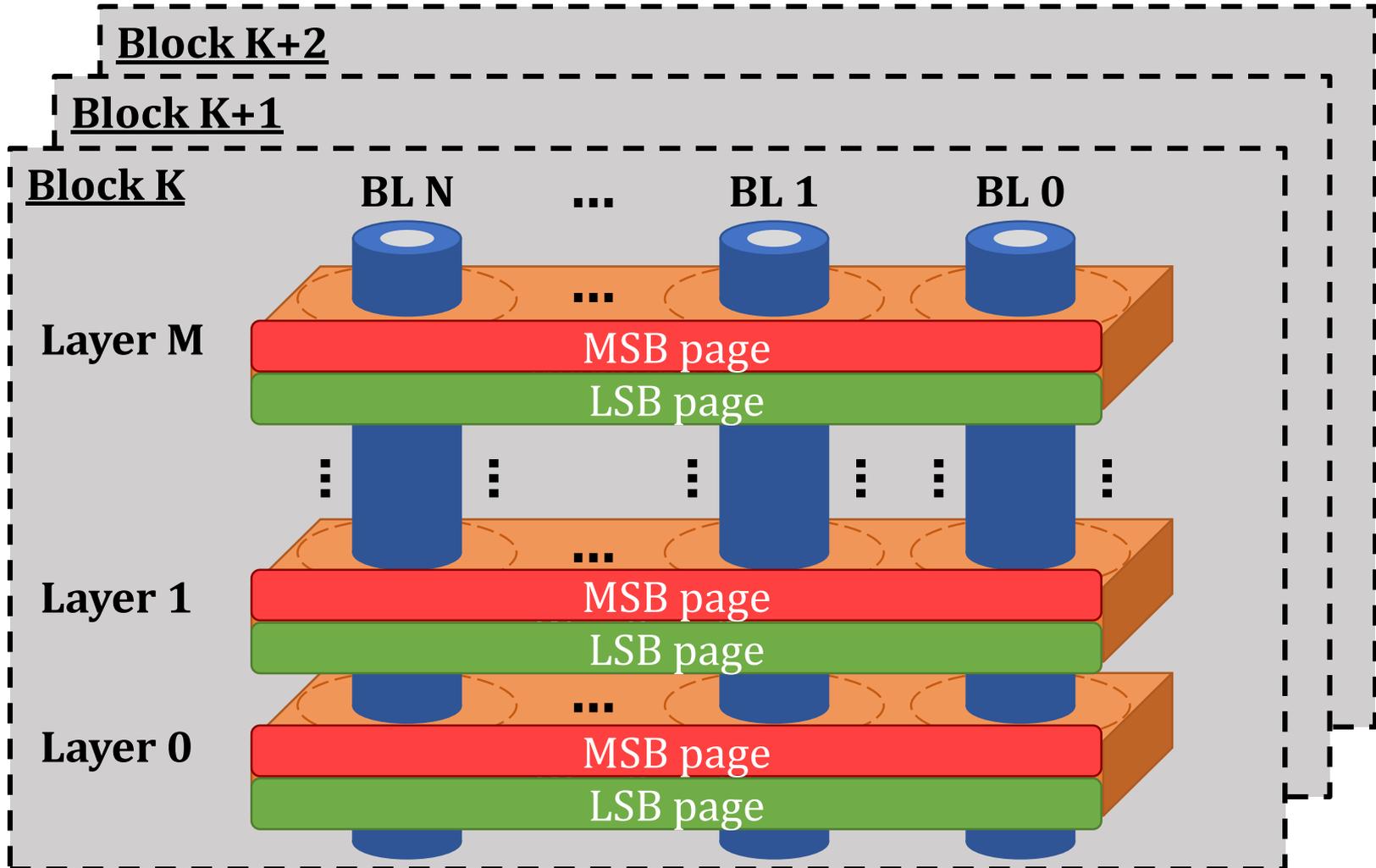
3D Charge Trap Cell



3D NAND Organization

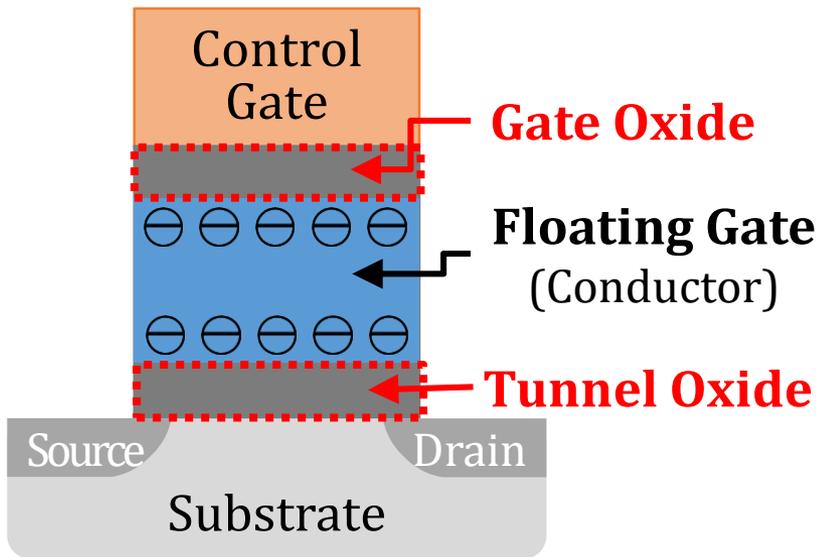


MLC NAND Page Organization

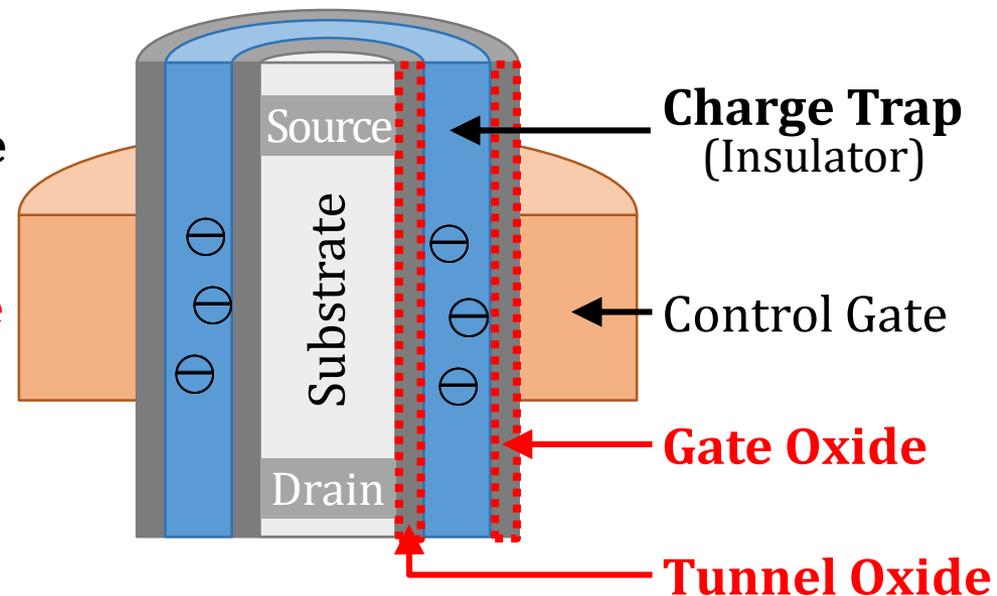


Root Cause of Early Retention Loss

Floating Gate Cell



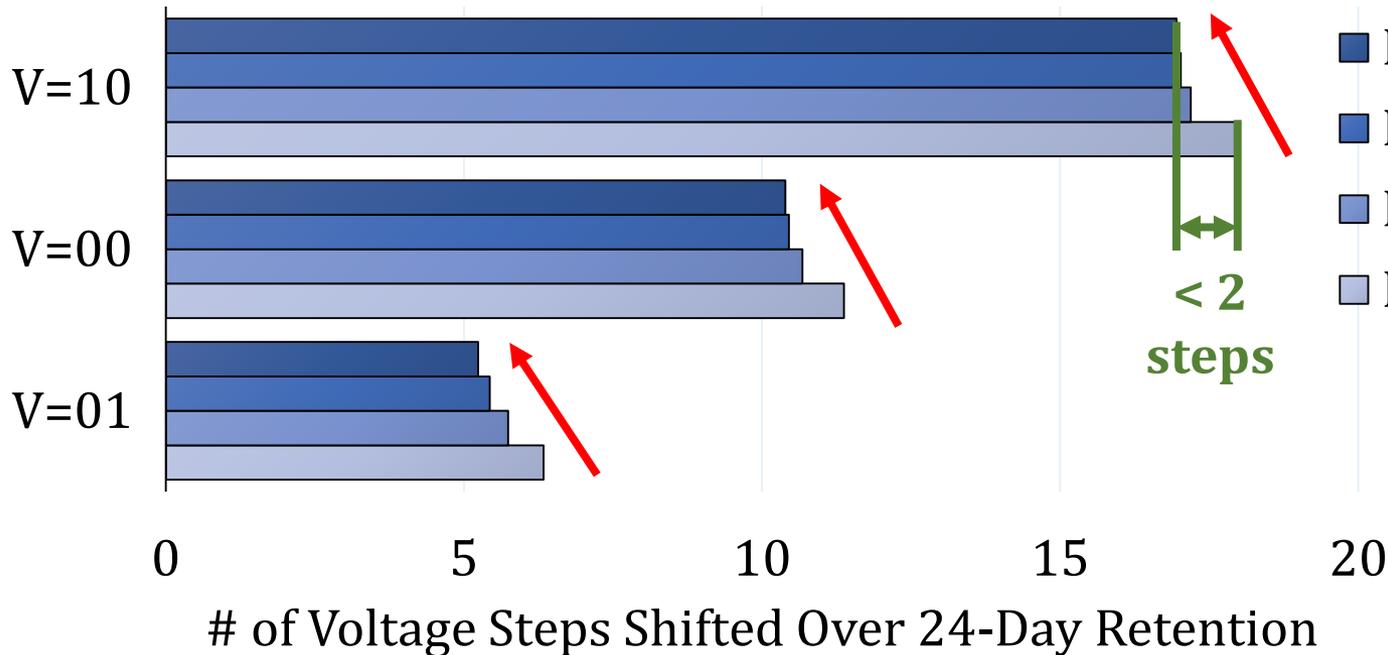
3D Charge Trap Cell



- Oxide layers are designed to be thinner in 3D NAND [Samsung WhitePaper'14]
→ Charges near the surface of charge trap layer leaks faster

Retention Interference

V: Victim



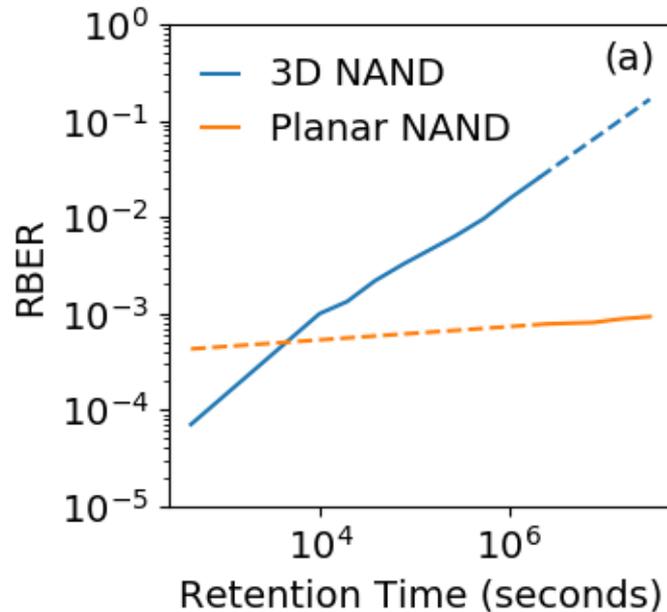
N: Neighbor

- N=10
- N=00
- N=01
- N=11



Retention loss speed correlated with neighbor cells' state

Retention Loss Model



- Because of early retention loss, V_{opt} shifts quickly after programming
 - **Linear correlation** between V_{opt} and $\log(t : \text{retention time})$
 - **Linear correlation** between $\log(\text{RBER})$ and $\log(t)$
- **Develop a simple linear model that can be used online**

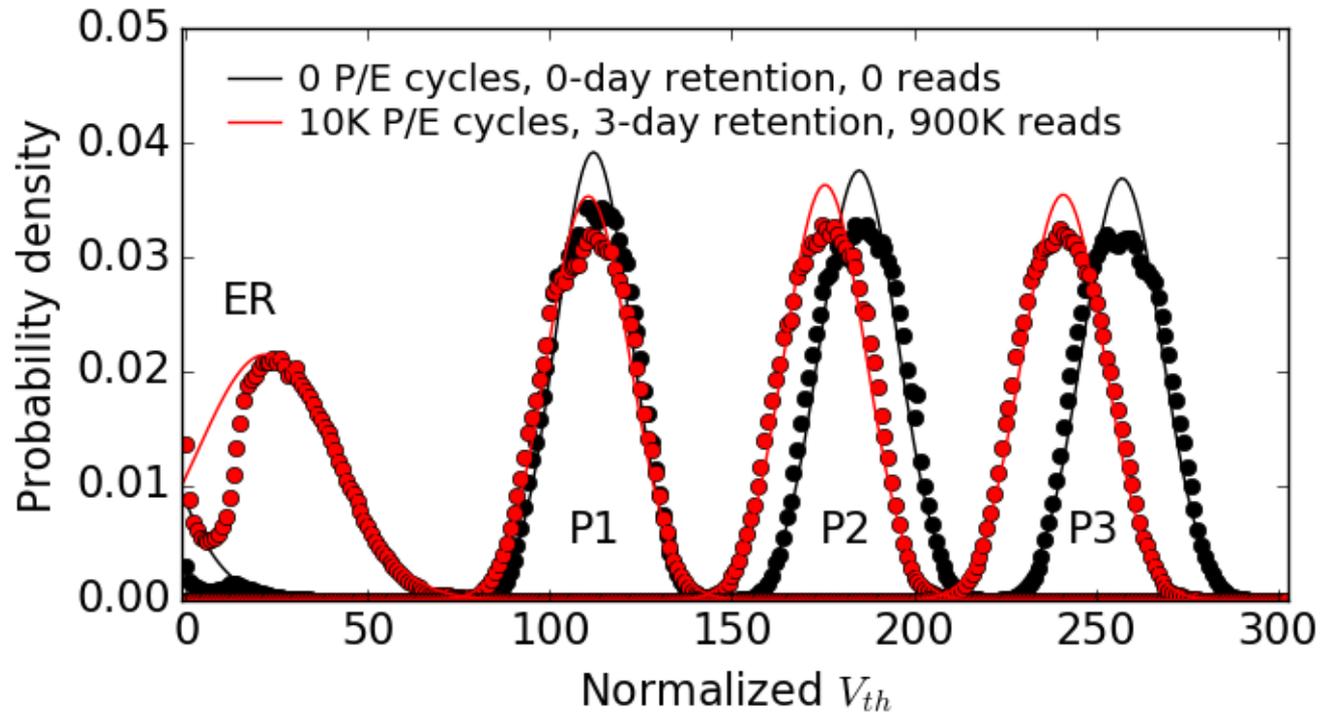
Error Mitigation Techniques

- **LaVAR: Layer Variation Aware Reading**
 - Learn a V_{opt} offset for each layer and apply *variation-aware V_{opt}*
 - Reduces RBER on average by **43%**
 - Uses **(2 × Layers)** Byte memory per chip
- **LI-RAID: Layer-Interleaved RAID**
 - Group flash pages on *less reliable layers* with pages on *more reliable layers*
 - Group *MSB pages* with *LSB pages*
 - Reduce worst-case RBER by **66.9%**
 - **<0.4%** storage overhead
- **ReMAR: Retention Model Aware Reading**
 - Learn retention loss model and apply *retention-aware V_{opt}*
 - Reduce RBER on average by **51.9%**
 - Uses **800 KB** memory to store program time of each block

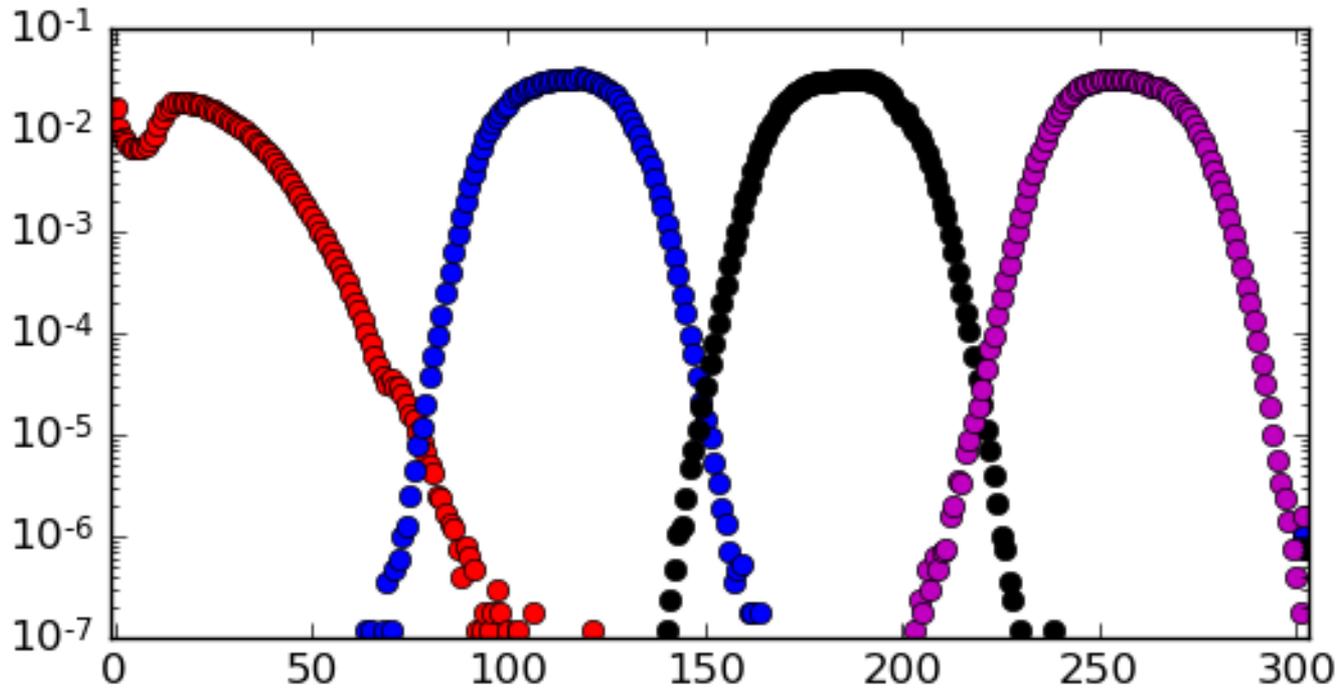
LI-RAID Data Layout

<i>Page/Wordline #/Layer #</i>	Chip 0	Chip 1	Chip 2	Chip 3
<i>MSB/Wordline 0/Layer 0</i>	Group 0	Blank	Group 4	Group 3
<i>LSB/Wordline 0/Layer 0</i>	Group 1	Blank	Group 5	Group 2
<i>MSB/Wordline 1/Layer 1</i>	Group 2	Group 1	Blank	Group 5
<i>LSB/Wordline 1/Layer 1</i>	Group 3	Group 0	Blank	Group 4
<i>MSB/Wordline 2/Layer 2</i>	Group 4	Group 3	Group 0	Blank
<i>LSB/Wordline 2/Layer 2</i>	Group 5	Group 2	Group 1	Blank
<i>MSB/Wordline 3/Layer 3</i>	Blank	Group 5	Group 2	Group 1
<i>LSB/Wordline 3/Layer 3</i>	Blank	Group 4	Group 3	Group 0

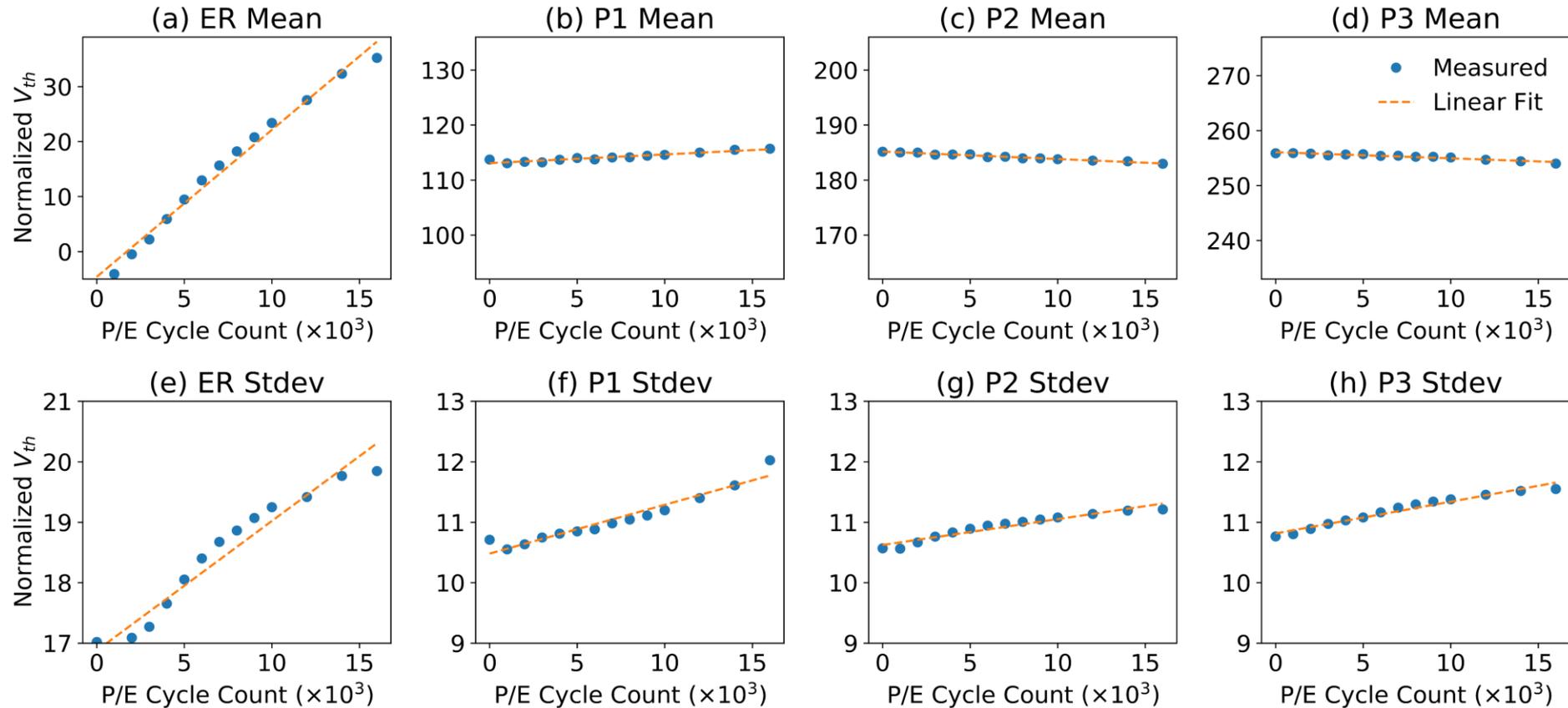
Threshold Voltage Shift



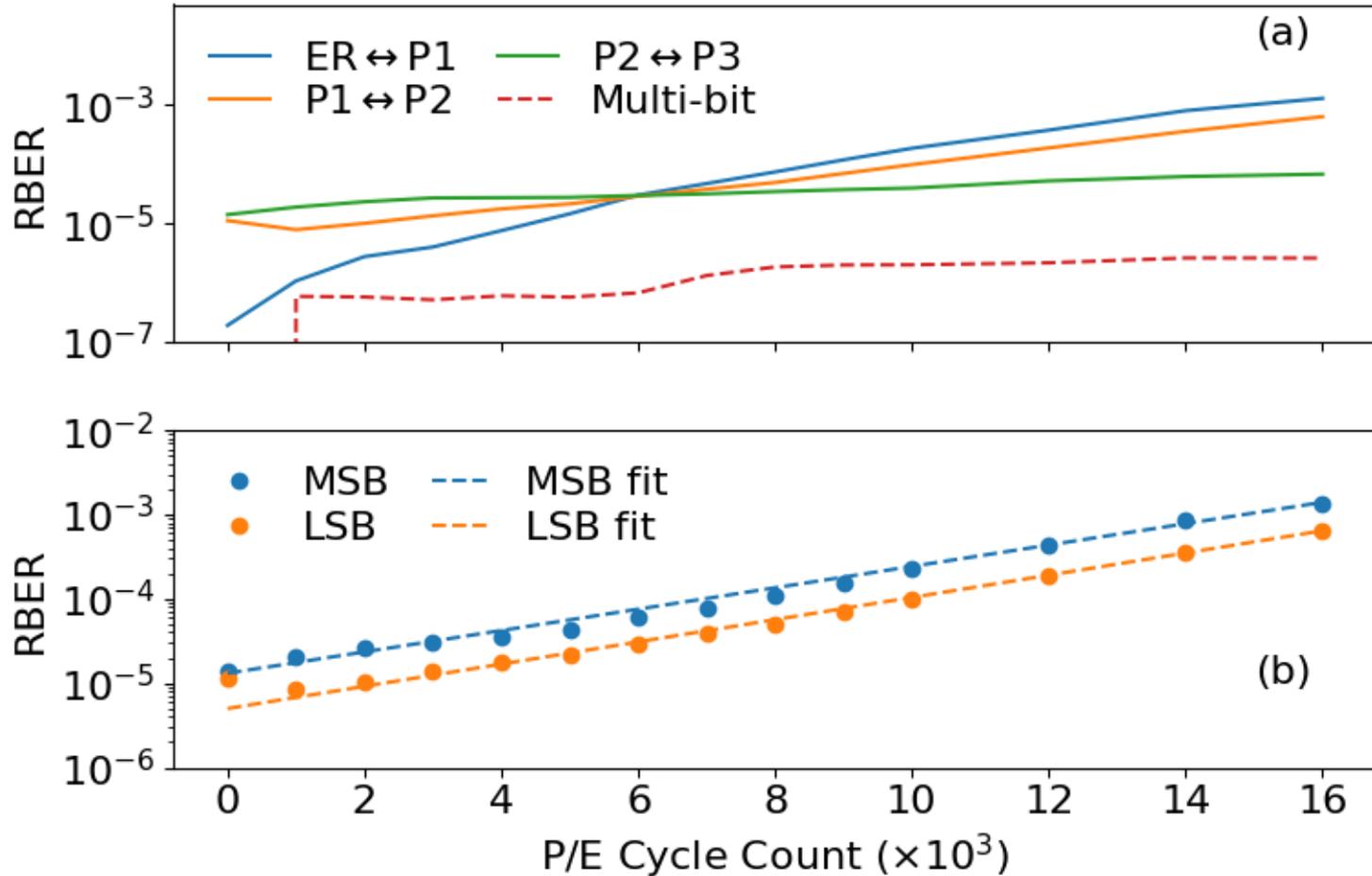
No Programming Errors



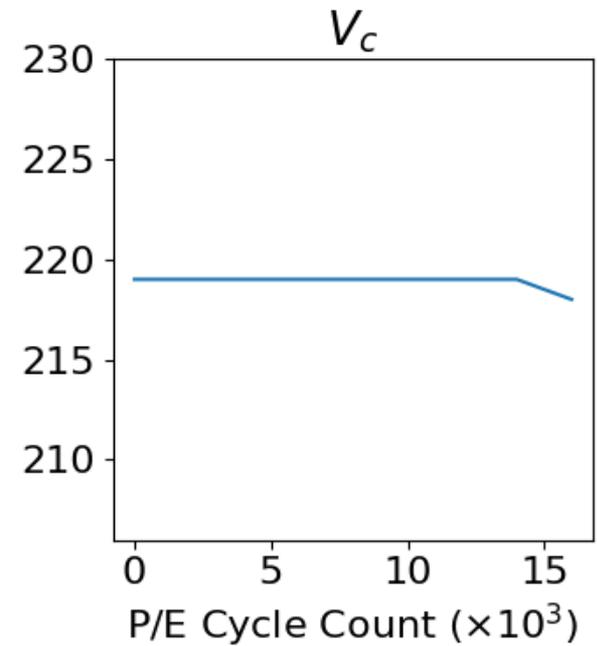
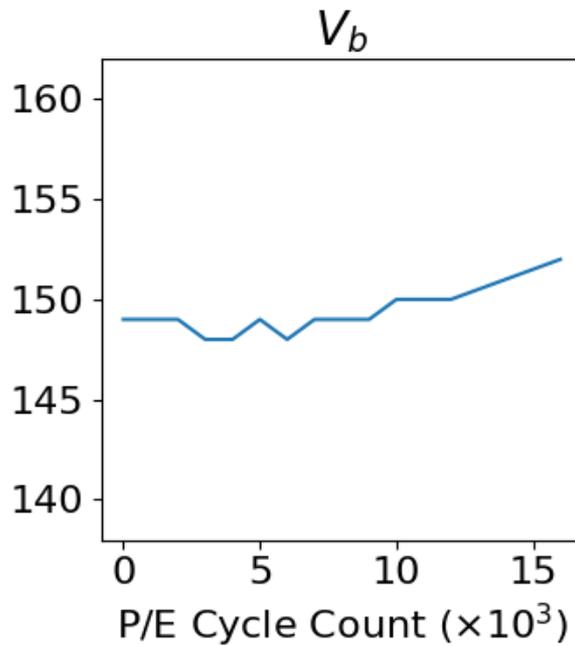
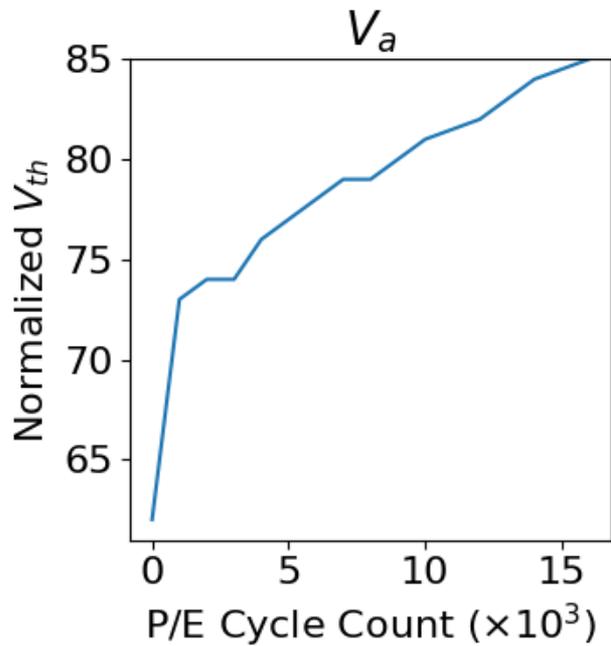
P/E Cycling Effect on Threshold Voltage



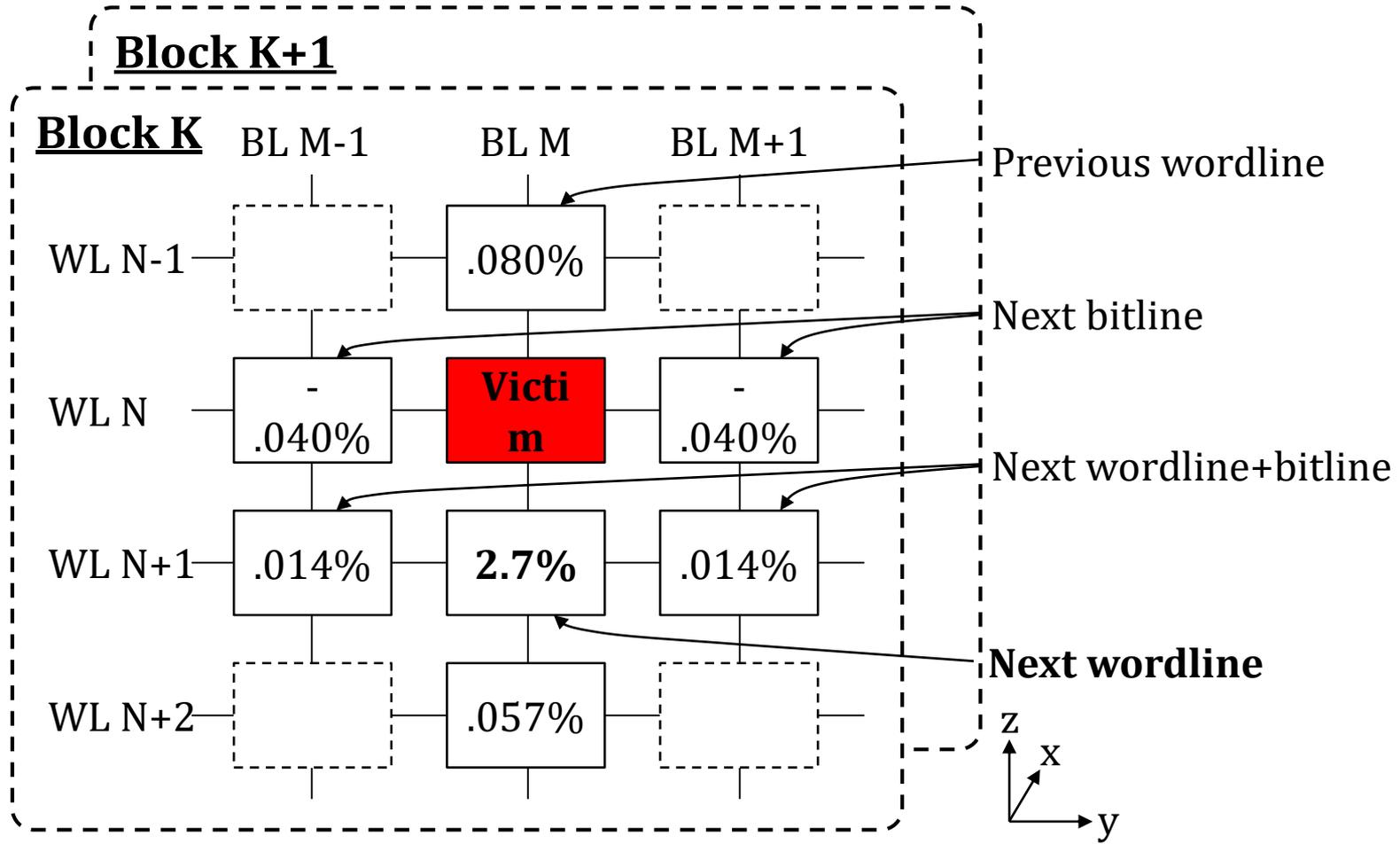
P/E Cycling Errors



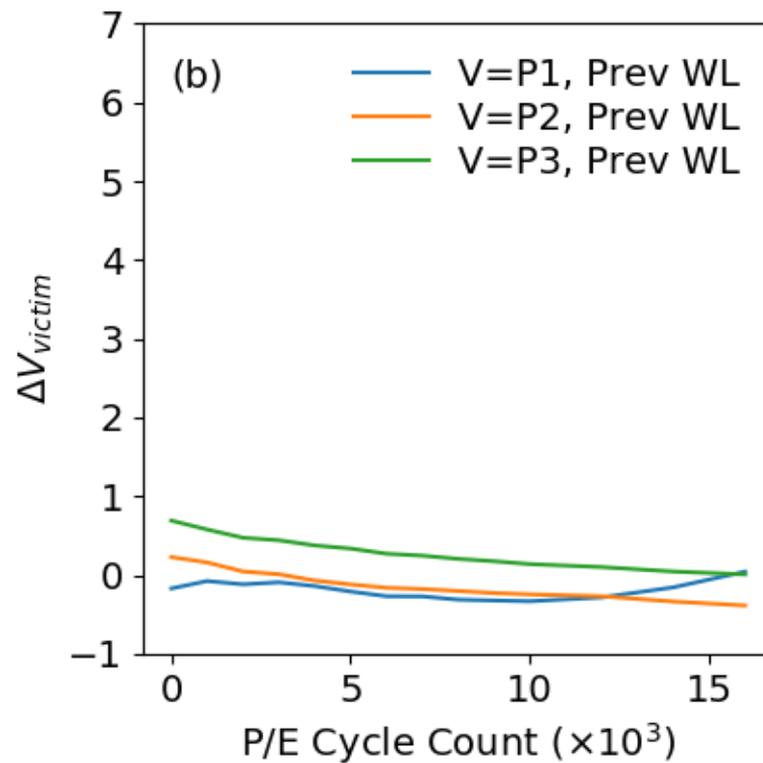
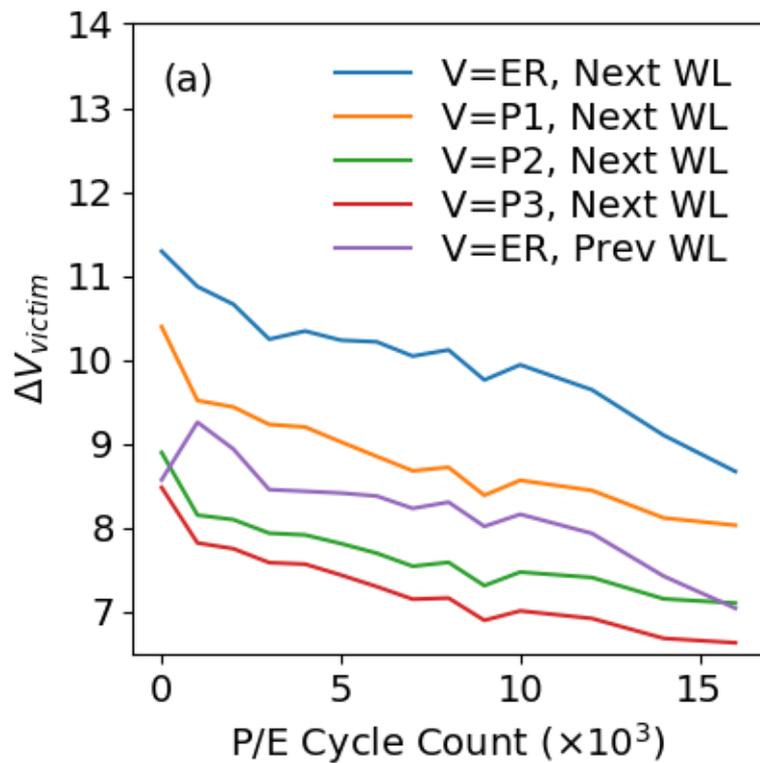
P/E Cycling Effect on Optimal Read Reference Voltages



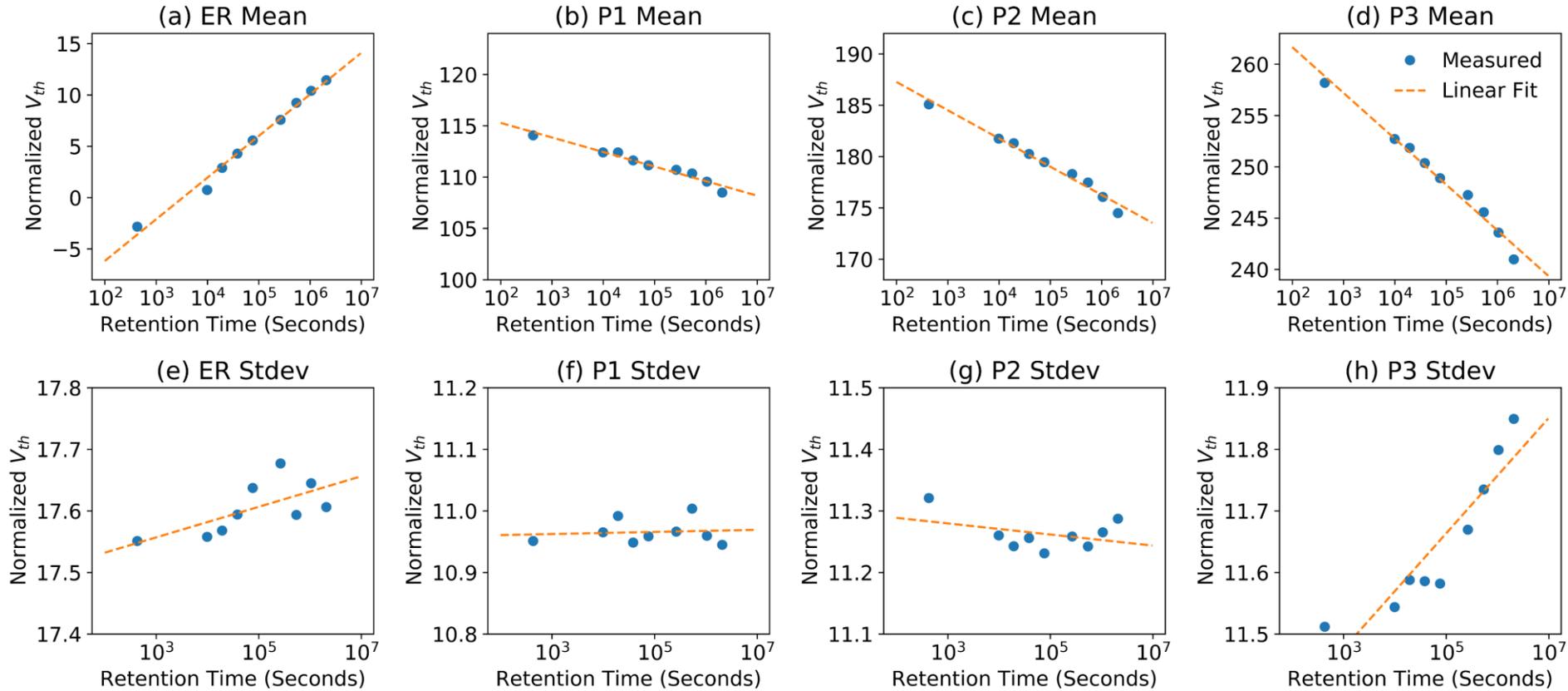
Program Interference Effect & Interference Correlation



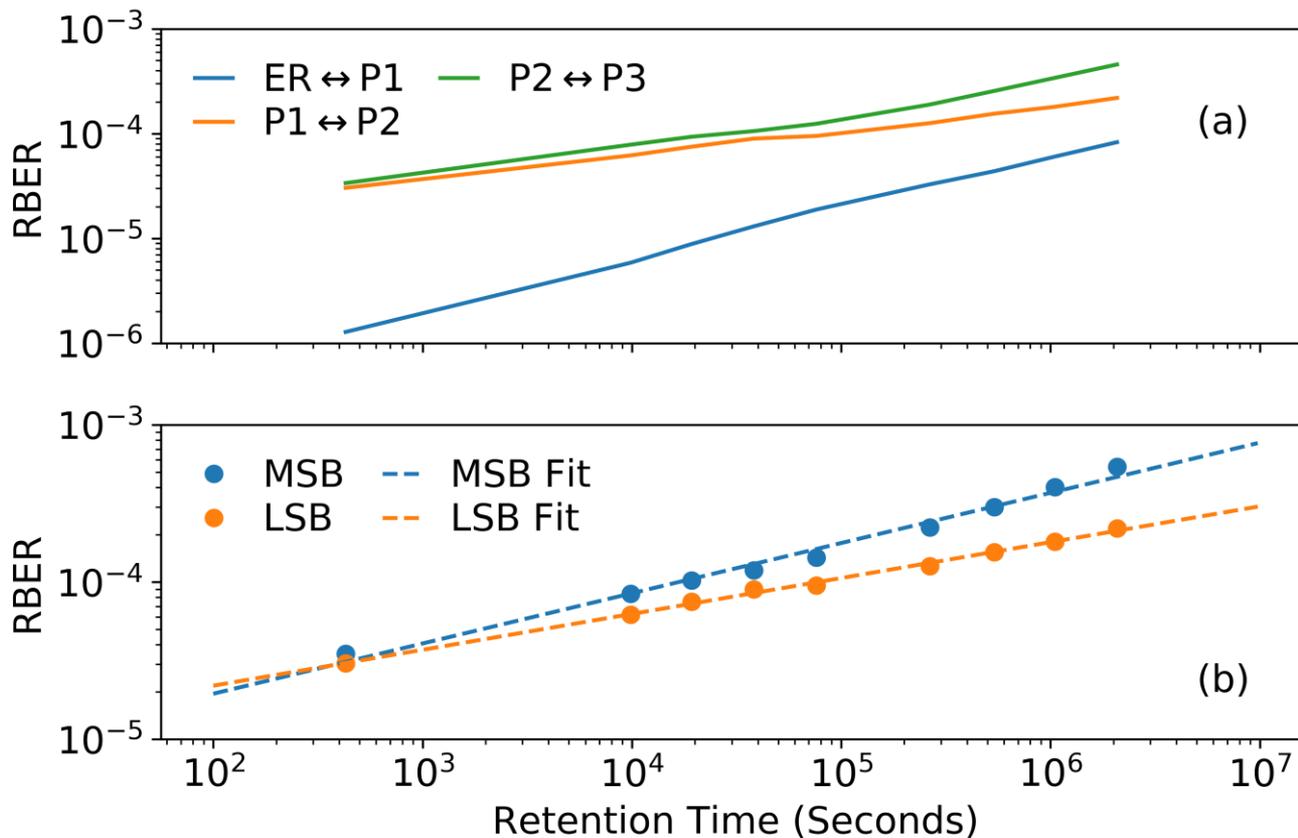
Program Interference vs. PEC



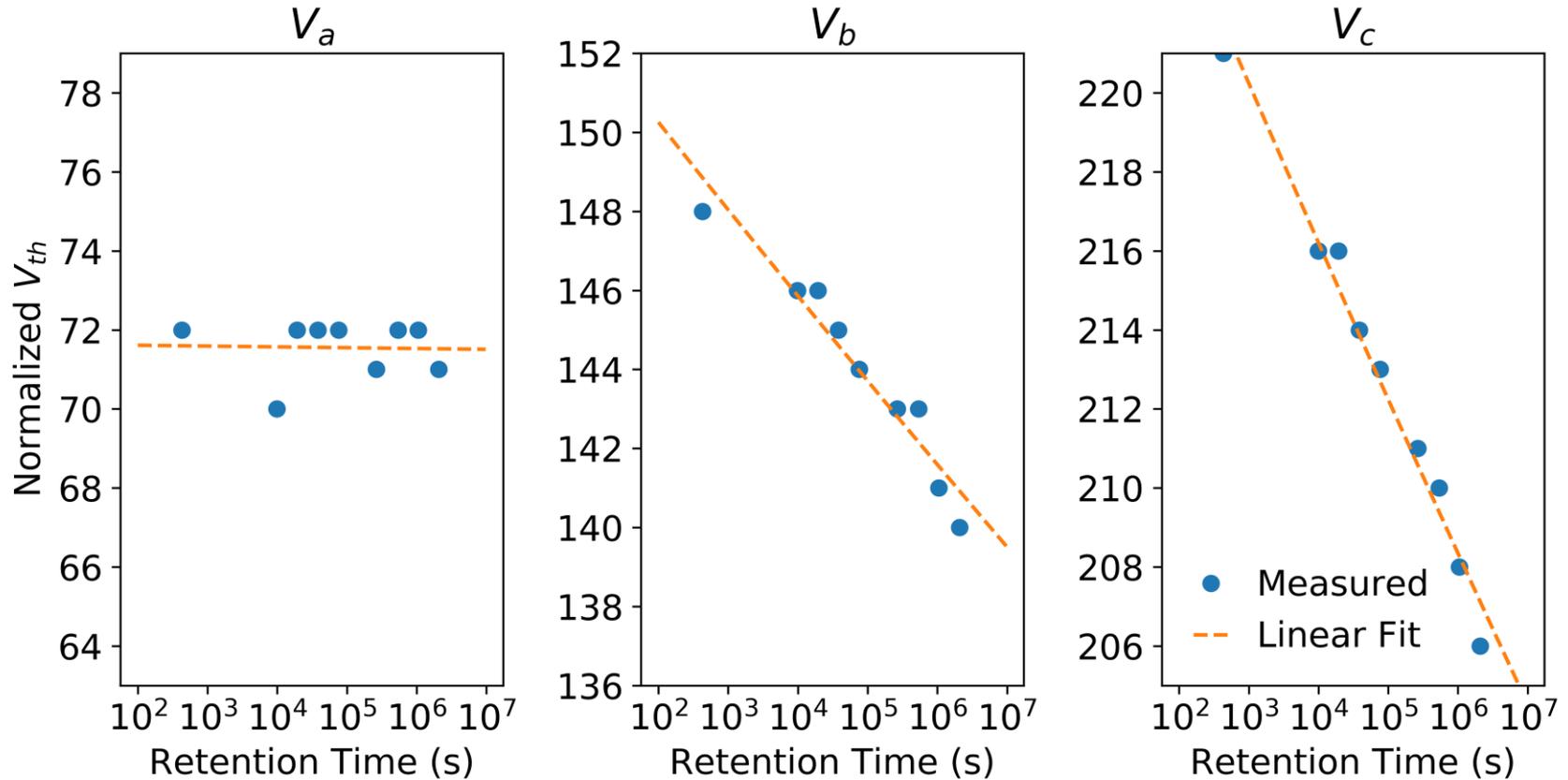
Early Retention Loss Effect on Threshold Voltage



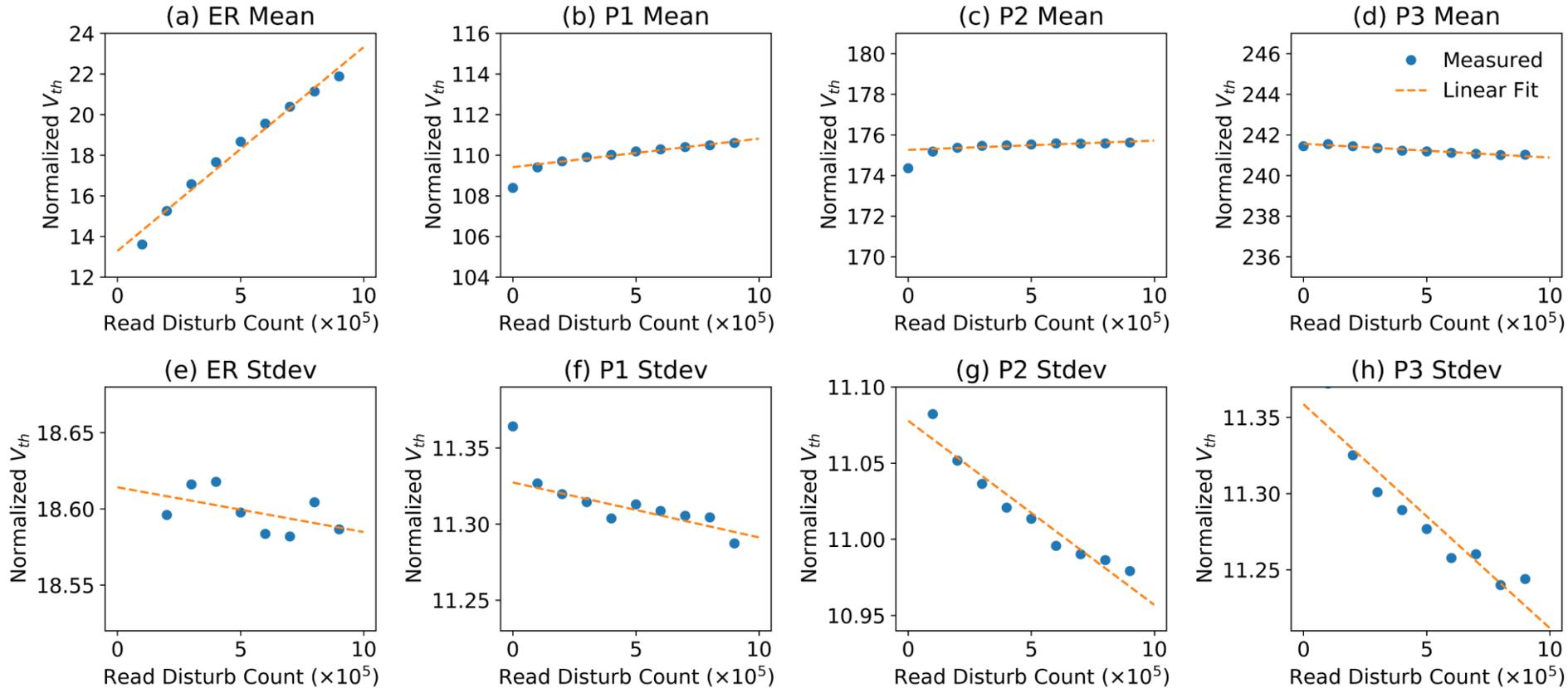
Early Retention Loss Errors



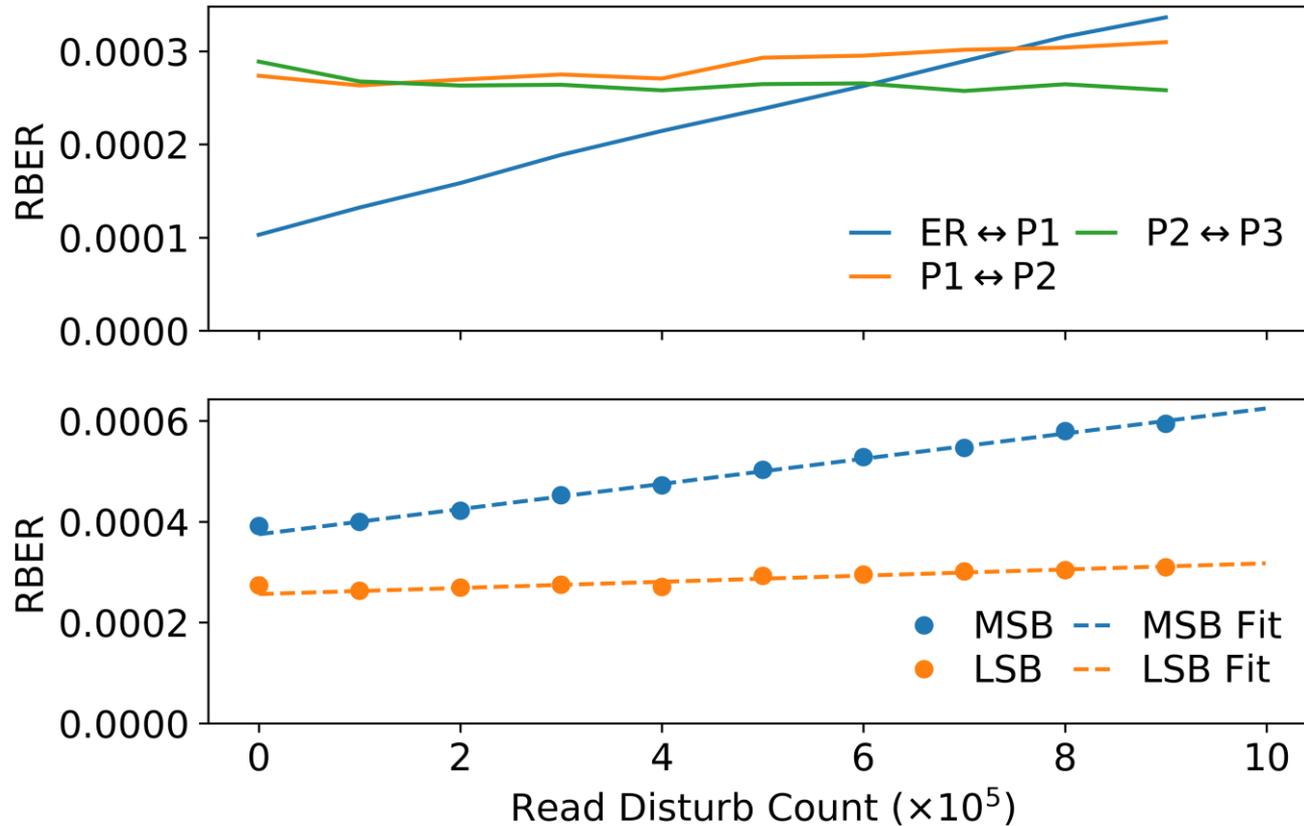
Early Retention Loss Effect on Optimal Read Reference Voltages



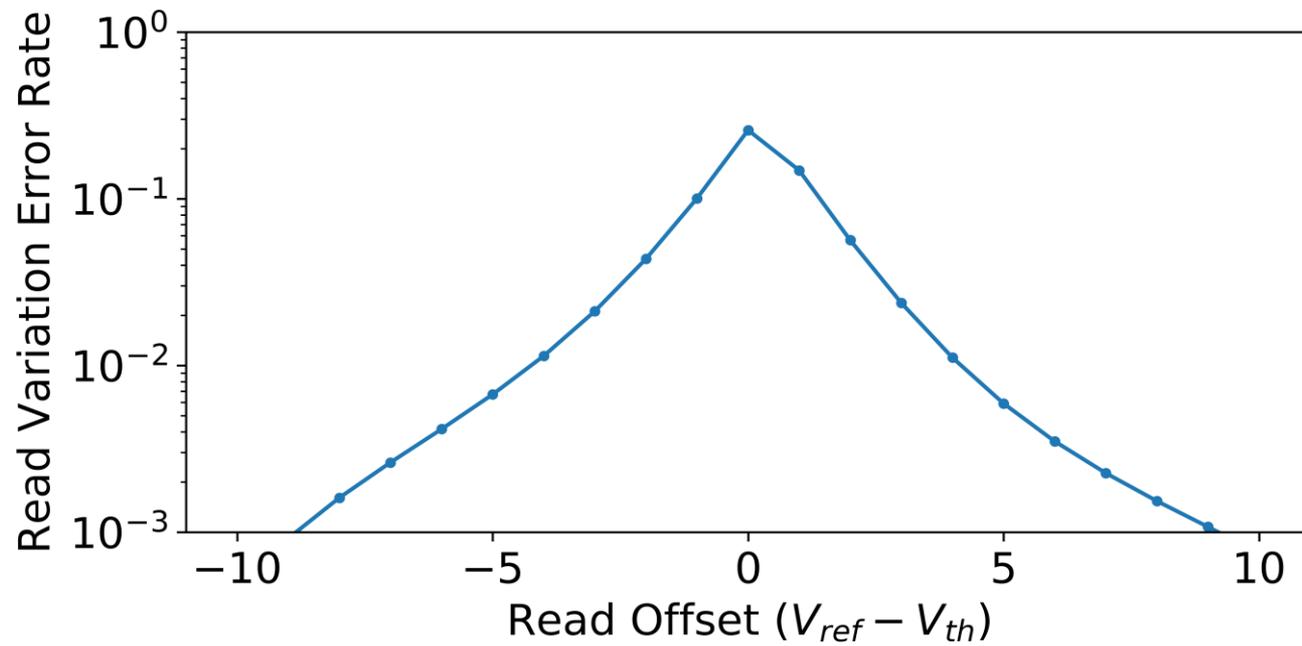
Read Disturb Effect on Threshold Voltage



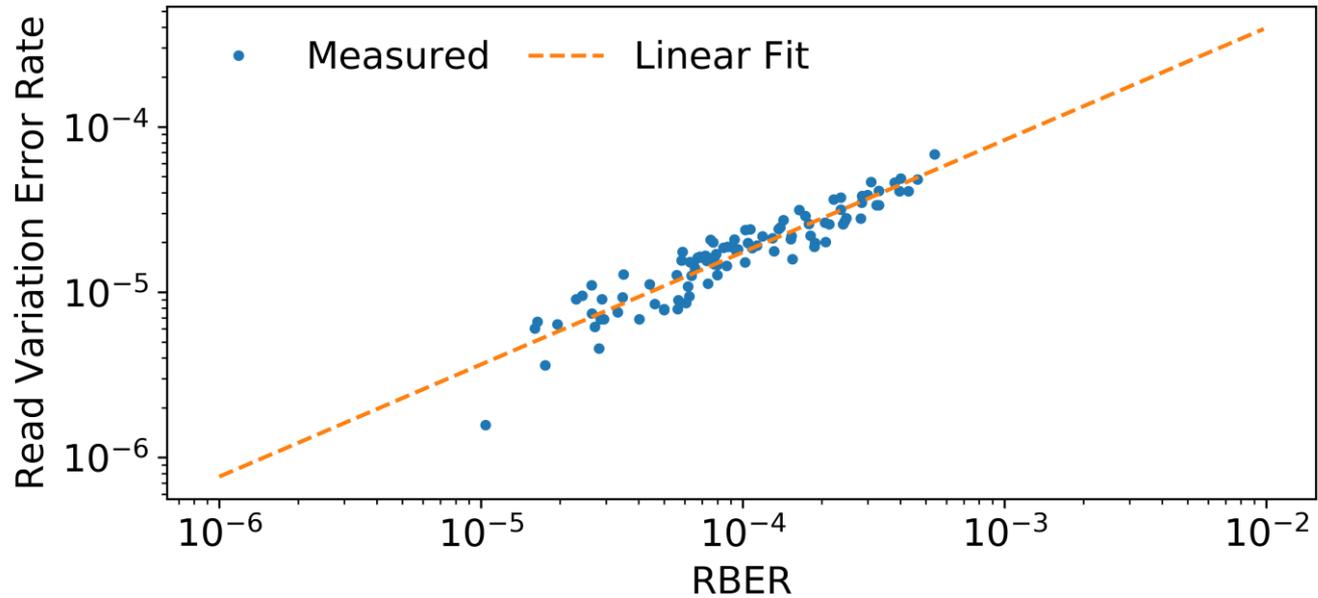
Read Disturb Errors



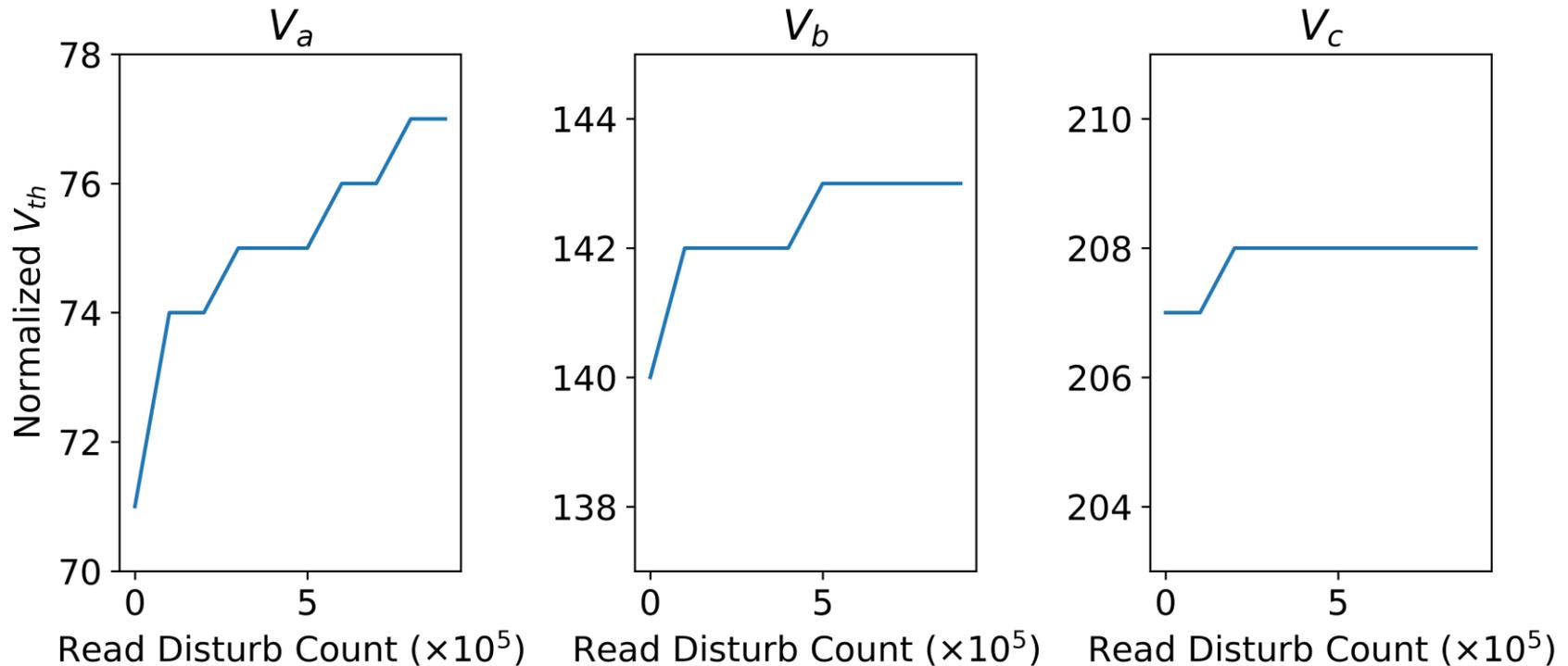
Read Variation Errors



Read Variation Errors vs. RBER



Read Disturb Effect on Optimal Read Reference Voltages



RBER Breakdown

