Error Analysis and Management for MLC NAND Flash Memory

Onur Mutlu
onur@cmu.edu

(joint work with Yu Cai, Gulay Yalcin, Erich Haratsch, Ken Mai, Adrian Cristal, Osman Unsal)

August 7, 2014

Flash Memory Summit 2014, Santa Clara, CA

SAFARI

Carnegie Mellon

Executive Summary

- Problem: MLC NAND flash memory reliability/endurance is a key challenge for satisfying future storage systems' requirements

- Our Goals: (1) Build reliable error models for NAND flash memory via experimental characterization, (2) Develop efficient techniques to improve reliability and endurance

- This talk provides a "flash" summary of our recent results published in the past 3 years:

  - Experimental error and threshold voltage characterization **[DATE'12&13]**
  - Retention-aware error management **[ICCD'12]**
  - Program interference analysis and read reference V prediction **[ICCD'13]**
  - Neighbor-assisted error correction **[SIGMETRICS'14]**

SAFARI

2

Agenda

- **Background, Motivation and Approach**
- Experimental Characterization Methodology
- Error Analysis and Management
  - Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
 - Neighbor-Assisted Error Correction
- Summary

SAFARI

Evolution of NAND Flash Memory
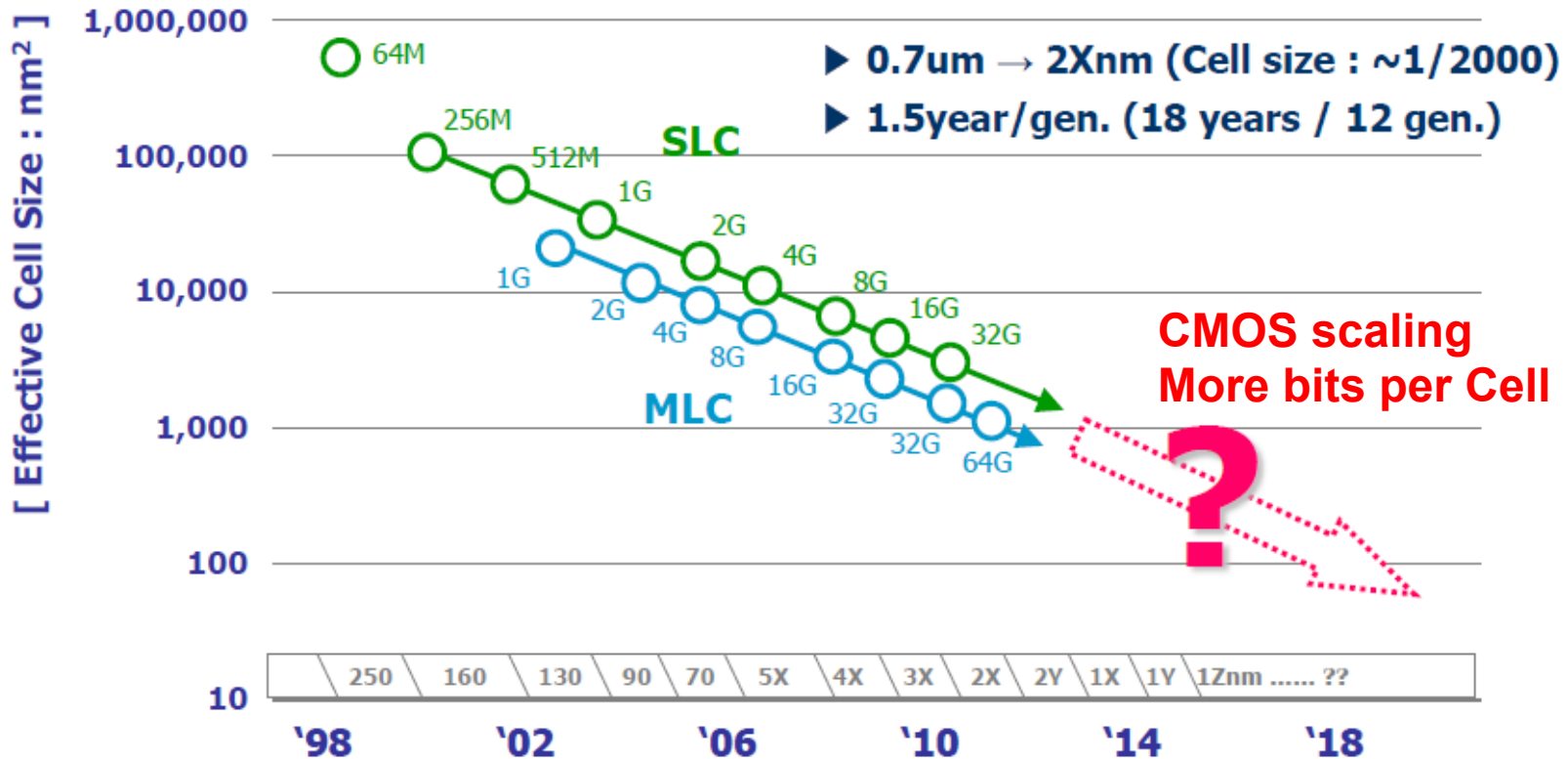
Seaung Suk Lee, "Emerging Challenges in NAND Flash Technology", Flash Summit 2011 (Hynix)

- ■ Flash memory is widening its range of applications
  - ❑ Portable consumer devices, laptop PCs and enterprise servers

Flash Challenges: Reliability and Endurance

NAND Flash Memory Endurance Properties

SLC • MLC • TLC

Lithography nm
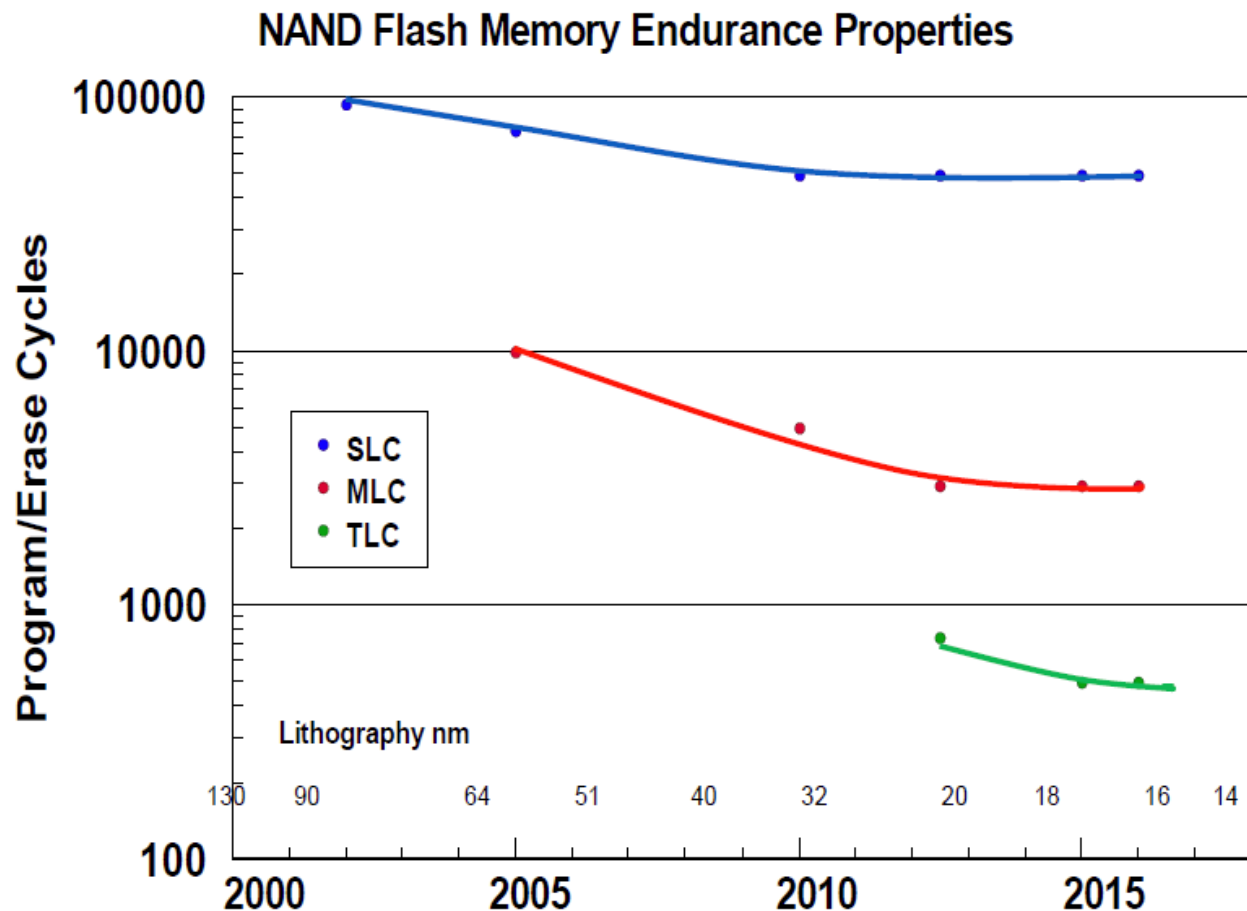
E. Grochowski et al., "Future technology challenges for NAND flash and HDD products", Flash Memory Summit 2012

- **P/E cycles (provided)**

  A few thousand

- **P/E cycles (required)**

  Writing the full capacity of the drive 10 times per day for 5 years (STEC)
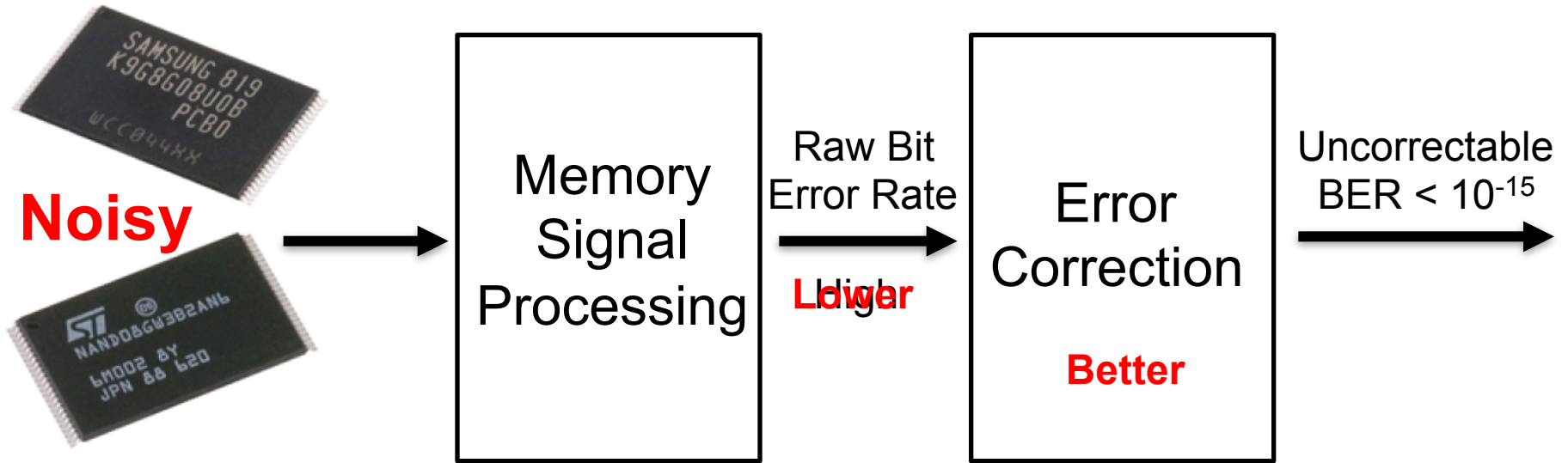
  > 50k P/E cycles

NAND Flash Memory is Increasingly Noisy

Write $\longrightarrow$ Noisy NAND $\longrightarrow$ Read

Future NAND Flash-based Storage Architecture

Noisy → Memory Signal Processing → Raw Bit Error Rate **Lower** → Error Correction **Better** → Uncorrectable BER $< 10^{-15}$

Our Goals:

Build reliable error models for NAND flash memory

Design efficient reliability mechanisms based on the model

NAND Flash Error Model

Write → Noisy NAND → Read

Experimentally characterize and model dominant errors

Cai et al., "Error Patterns in MLC NAND Flash Memory: Measurement, Characterization, and Analysis"", **DATE 2012**

Write →
┌─────────────────────┐   ┌─────────────────────┐   ┌─────────────────────┐
│ ▪ **Erase block**   │ →   │ ▪ **Neighbor page** │ →   │ ▪ **Retention**   │ → Read
│ ▪ **Program page** │   │ **program (c-to-c** │   │   │
│   │   │ **interference)**   │   │   │
└─────────────────────┘   └─────────────────────┘   └─────────────────────┘

Cai et al., "Threshold voltage distribution in MLC NAND Flash Memory: Characterization, Analysis, and Modeling", **DATE 2013**

Cai et al., "Program Interference in MLC NAND Flash Memory: Characterization, Modeling, and Mitigation", **ICCD 2013**

Cai et al., "Neighbor-Cell Assisted Error Correction in MLC NAND Flash Memories", **SIGMETRICS 2014**

Cai et al., "Flash Correct-and-Refresh: Retention-aware error management for increased flash memory lifetime", **ICCD 2012**

Cai et al., "Error Analysis and Retention-Aware Error Management for NAND Flash Memory, **ITJ 2013**

SAFARI

Our Goals and Approach

- **Goals:**

  - Understand error mechanisms and develop reliable predictive models for MLC NAND flash memory errors

  - Develop efficient error management techniques to mitigate errors and improve flash reliability and endurance

- **Approach:**

  - Solid experimental analyses of errors in real MLC NAND flash memory → drive the understanding and models

  - Understanding, models and creativity → drive the new techniques

Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
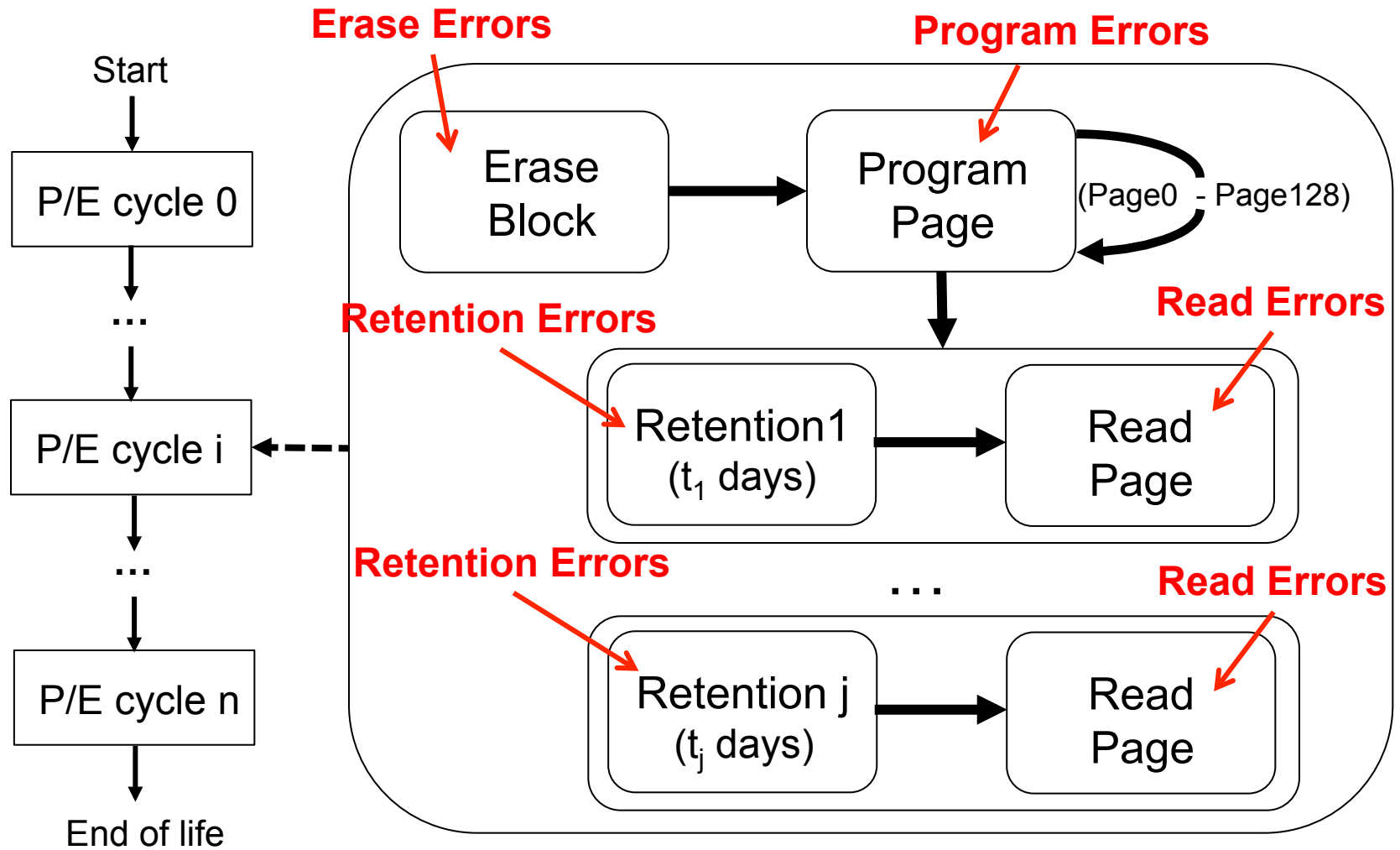 - Neighbor-Assisted Error Correction
- Summary

SAFARI

Experimental Testing Platform
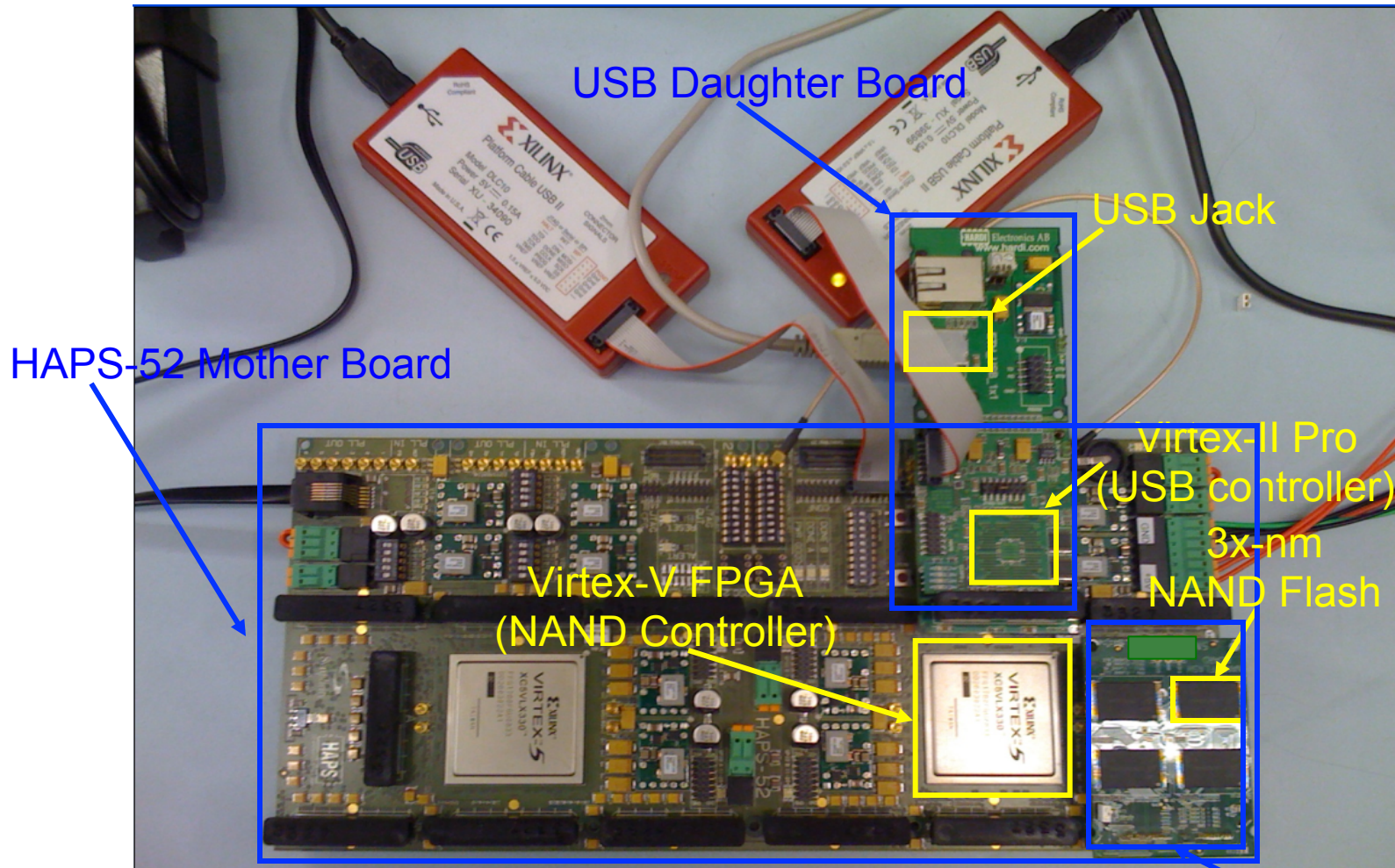
[Cai+, FCCM 2011, DATE 2012, ICCD 2012, DATE 2013, ITJ 2013, ICCD 2013, SIGMETRICS 2014]

Cai et al., FPGA-based Solid-State Drive prototyping platform, FCCM 2011.

NAND Flash Usage and Error Model

Methodology: Error and ECC Analysis

- **Characterized errors and error rates** of 3x and 2y-nm MLC NAND flash using an experimental FPGA-based platform
  - **[Cai+, DATE'12, ICCD'12, DATE'13, ITJ'13, ICCD'13, SIGMETRICS'14]**

- **Quantified Raw Bit Error Rate (RBER) at a given P/E cycle**
  - Raw Bit Error Rate: Fraction of erroneous bits without any correction

- **Quantified error correction capability** (and area and power consumption) of various BCH-code implementations
  - Identified how much RBER each code can tolerate
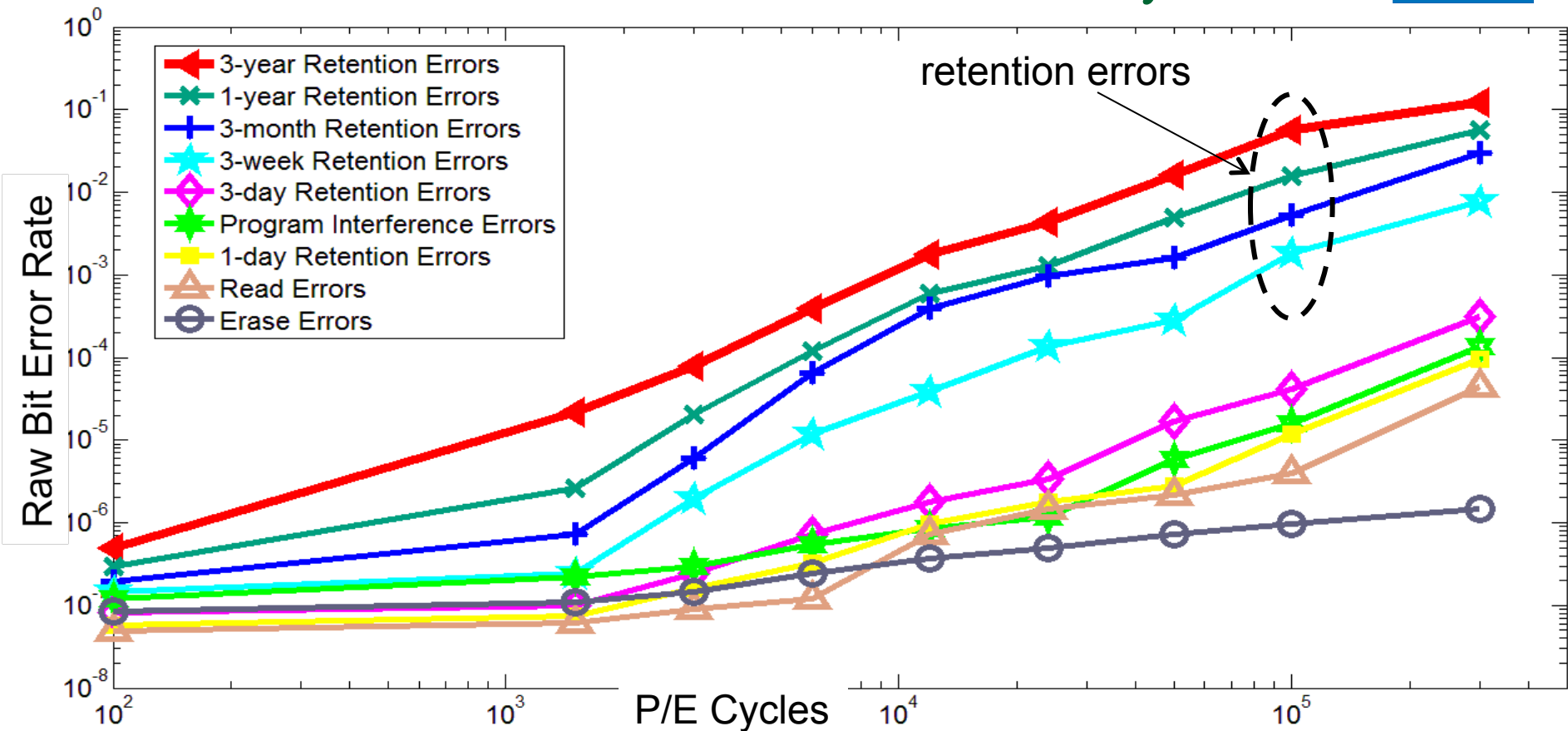   - → how many P/E cycles (flash lifetime) each code can sustain

NAND Flash Error Types

- Four types of errors [Cai+, DATE 2012]

- Caused by common flash operations
  - Read errors
  - Erase errors
  - Program (interference) errors

- Caused by flash cell losing charge over time
  - Retention errors
    - Whether an error happens depends on required retention time
    - Especially problematic in MLC flash because threshold voltage window to determine stored value is smaller

Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
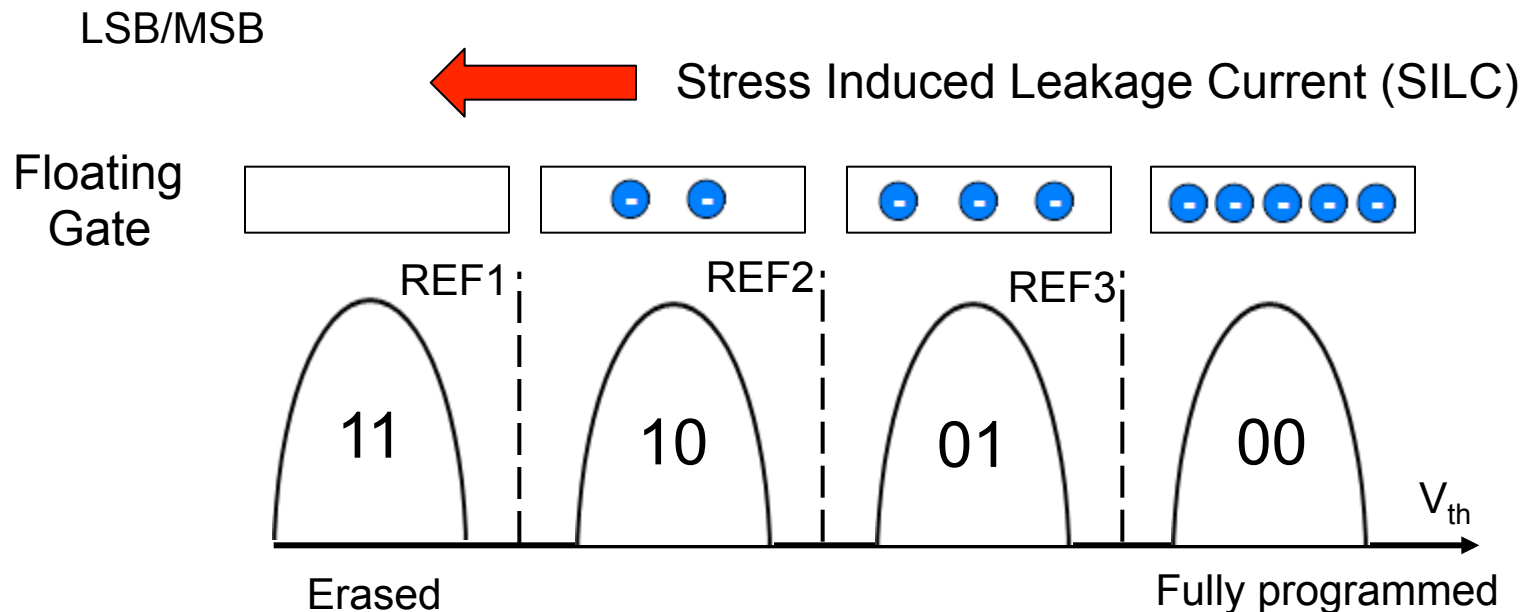 - Neighbor-Assisted Error Correction
- Summary

SAFARI

Observations: Flash Error Analysis

Legend:
- 3-year Retention Errors
- 1-year Retention Errors
- 3-month Retention Errors
- 3-week Retention Errors
- 3-day Retention Errors
- Program Interference Errors
- 1-day Retention Errors
- Read Errors
- Erase Errors

X-axis: P/E Cycles (10^2 to 10^5)
Y-axis: Raw Bit Error Rate (10^{-8} to 10^0)

retention errors

- Raw bit error rate increases exponentially with P/E cycles
- Retention errors are dominant (>99% for 1-year ret. time)
- Retention errors increase with retention time requirement

SAFARI Cai et al., Error Patterns in MLC NAND Flash Memory, DATE 2012. 16

Retention Error Mechanism

LSB/MSB

Stress Induced Leakage Current (SILC)

Floating Gate

REF1   REF2   REF3

11   10   01   00

V_{th}

Erased   Fully programmed

- **Electron loss from the floating gate causes retention errors**
  - Cells with more programmed electrons suffer more from retention errors
  - Threshold voltage is more likely to shift by one window than by multiple

SAFARI

Retention Error Value Dependency
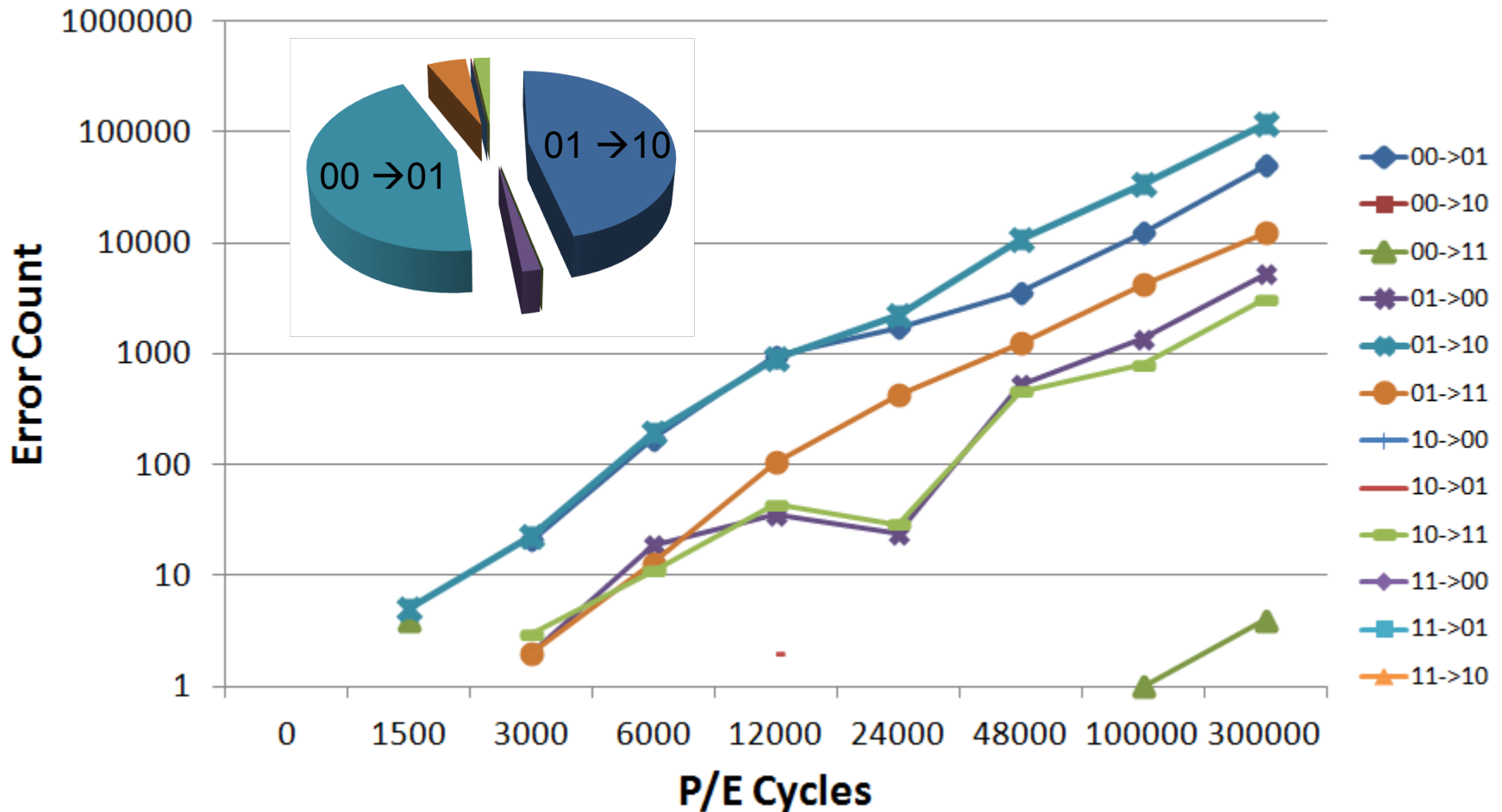
- Cells with more programmed electrons tend to suffer more from retention noise (i.e. 00 and 01)

More on Flash Error Analysis

- Yu Cai, Erich F. Haratsch, Onur Mutlu, and Ken Mai, **"Error Patterns in MLC NAND Flash Memory: Measurement, Characterization, and Analysis"** *Proceedings of the Design, Automation, and Test in Europe Conference (**DATE**)*, Dresden, Germany, March 2012. Slides (ppt)

Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
- Summary

SAFARI

Flash Correct-and-Refresh (FCR)

- **Key Observations:**
  - ❑ Retention errors are the dominant source of errors in flash memory **[Cai+ DATE 2012][Tanakamaru+ ISSCC 2011]**
    - → limit flash lifetime as they increase over time
  - ❑ Retention errors can be corrected by "refreshing" each flash page periodically

- **Key Idea:**
  - ❑ Periodically read each flash page,
  - ❑ Correct its errors using "weak" ECC, and
  - ❑ Either remap it to a new physical page or reprogram it in-place,
  - ❑ Before the page accumulates more errors than ECC-correctable
  - ❑ Optimization: Adapt refresh rate to endured P/E cycles

Cai et al., Flash Correct and Refresh, ICCD 2012.

FCR: Two Key Questions

- How to refresh?
    - Remap a page to another one
    - Reprogram a page (in-place)
    - Hybrid of remap and reprogram

- When to refresh?
    - Fixed period
    - Adapt the period to retention error severity

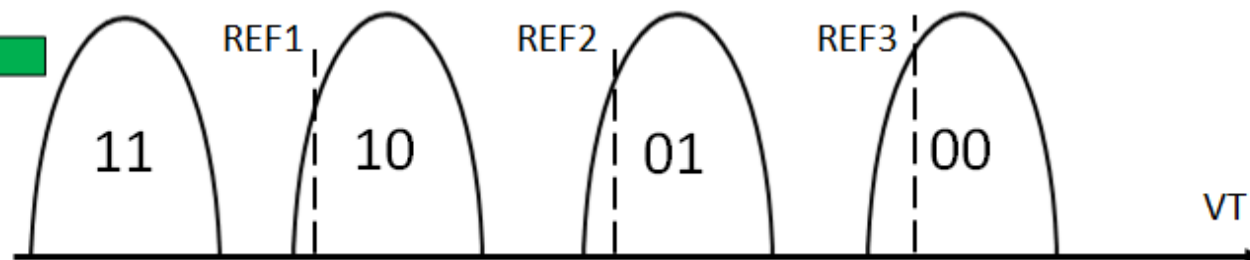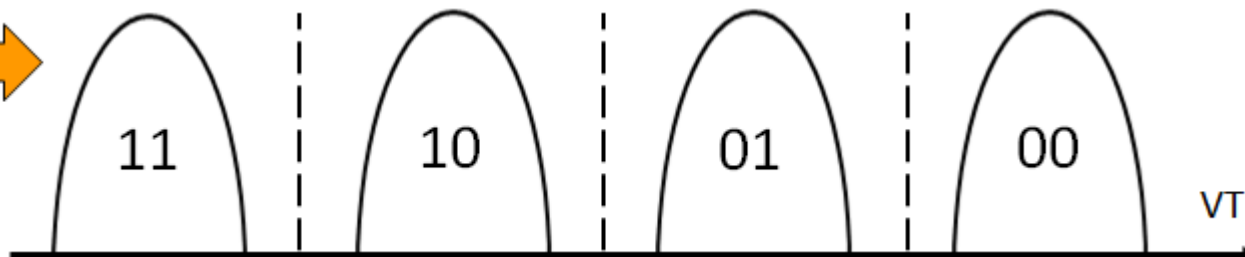# In-Place Reprogramming of Flash Cells

Floating Gate

Floating Gate Voltage Distribution for each Stored Value
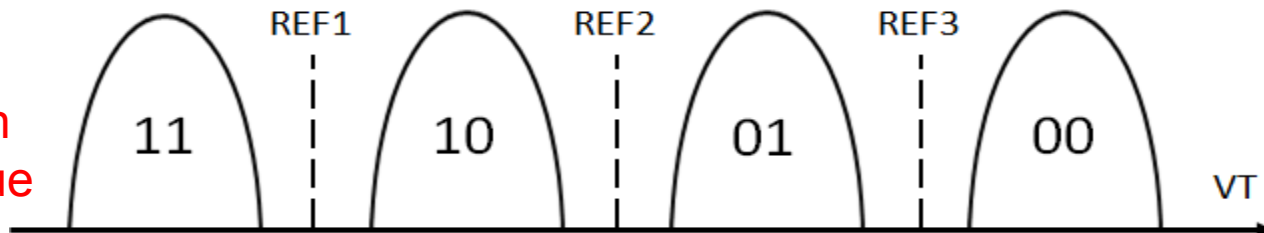
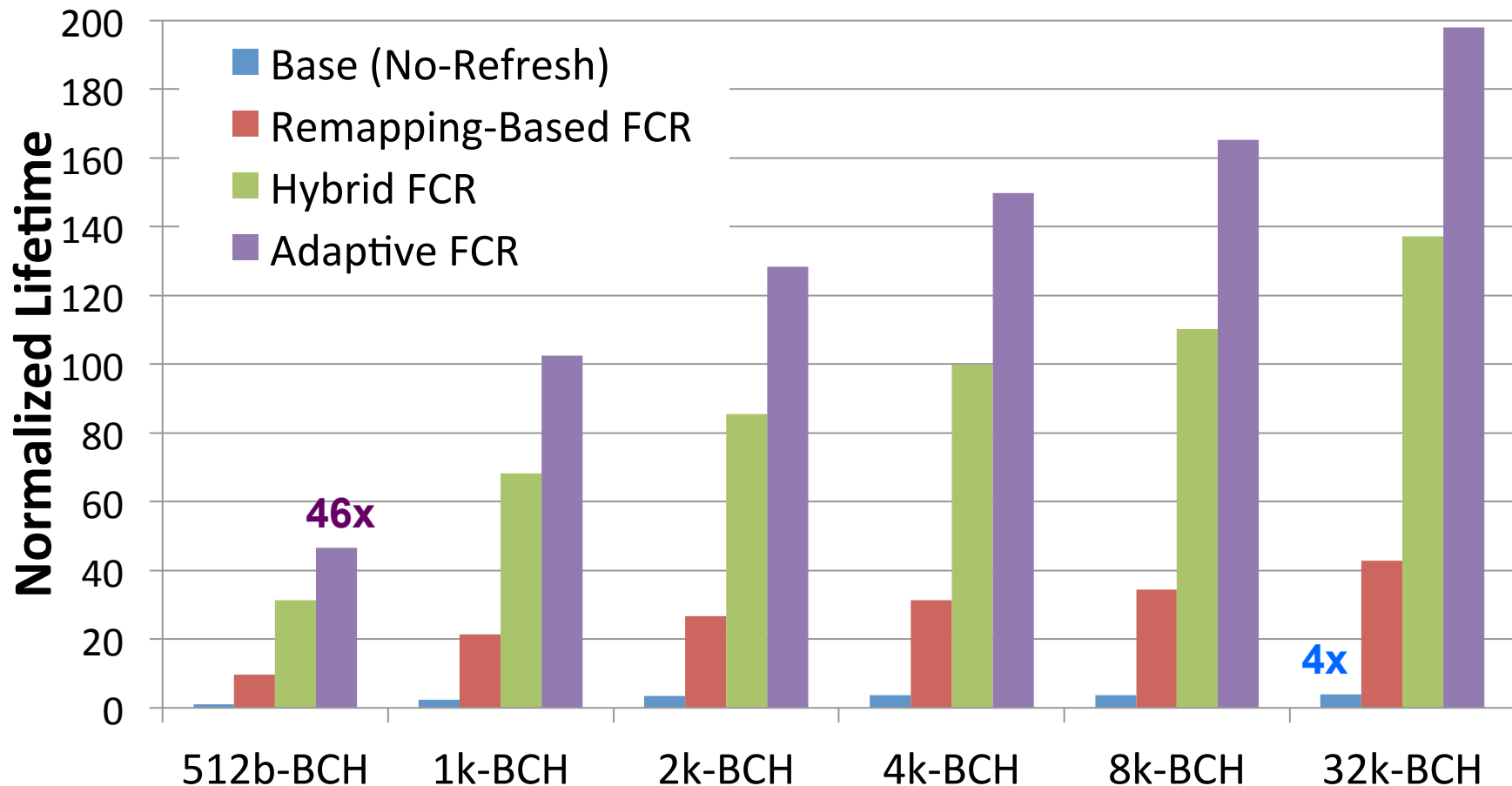Retention errors are caused by cell voltage shifting to the left

ISPP moves cell voltage to the right; fixes retention errors

- Pro: No remapping needed → no additional erase operations
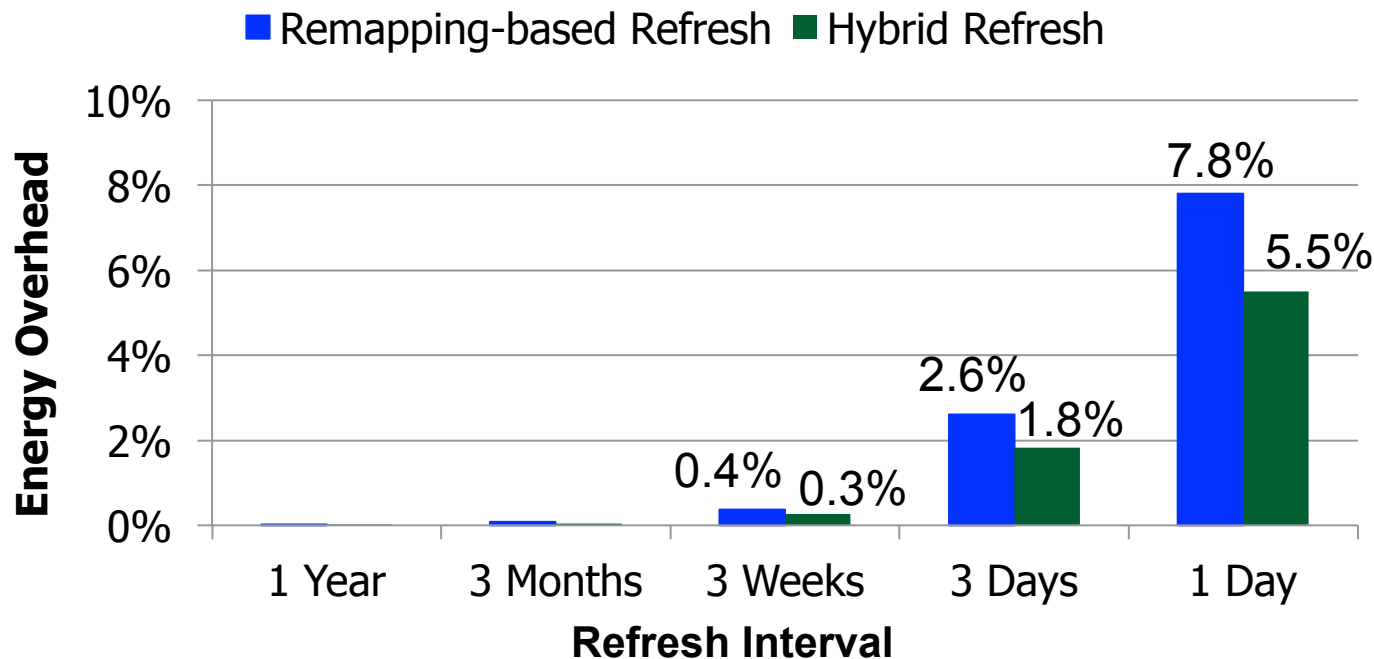- Con: Increases the occurrence of program errors

SAFARI

Normalized Flash Memory Lifetime

Lifetime of FCR much higher than lifetime of stronger ECC

Energy Overhead

- Adaptive-rate refresh: <1.8% energy increase until daily refresh is triggered

# More Detail and Analysis on FCR

- Yu Cai, Gulay Yalcin, Onur Mutlu, Erich F. Haratsch, Adrian Cristal, Osman Unsal, and Ken Mai,
"Flash Correct-and-Refresh: Retention-Aware Error Management for Increased Flash Memory Lifetime"
Proceedings of the
30th IEEE International Conference on Computer Design
(**ICCD**), Montreal, Quebec, Canada, September 2012.
Slides (ppt) (pdf)

Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
  - Neighbor-Assisted Error Correction
- Summary

SAFARI

Key Questions

- How does threshold voltage (Vth) distribution of different programmed states change over flash lifetime?

- Can we model it accurately and predict the Vth changes?

- Can we build mechanisms that can correct for Vth changes? (thereby reducing read error rates)

SAFARI

Threshold Voltage Distribution Model

Characterized on 2Y-nm chips using the read-retry feature
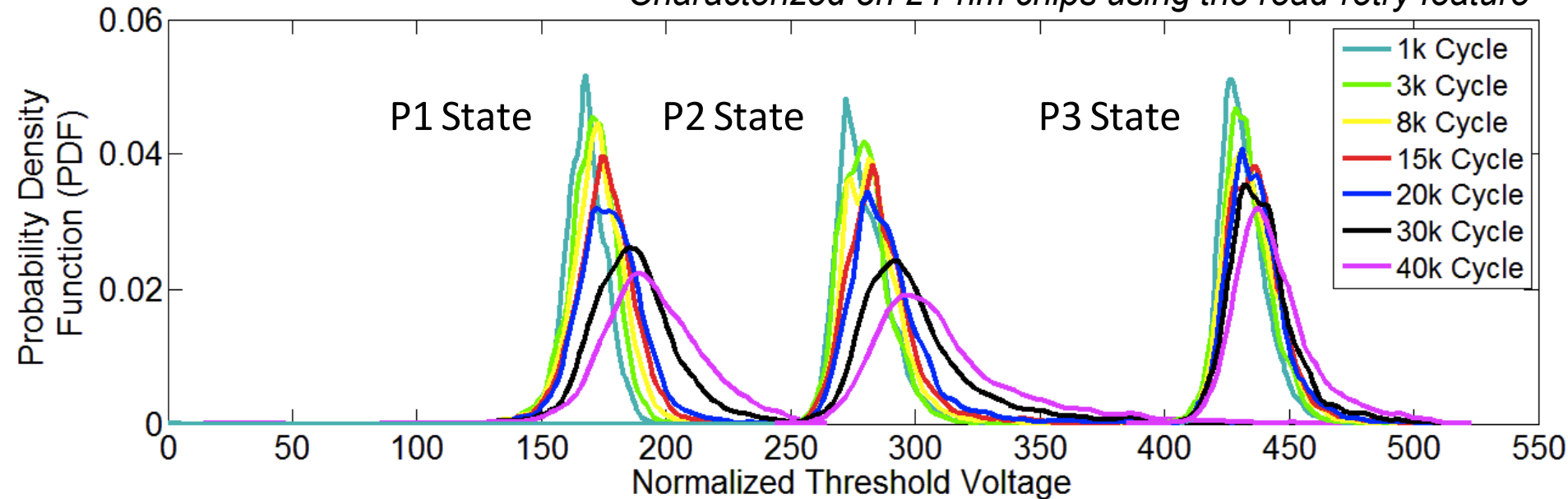
Gaussian distribution with additive white noise

As P/E cycles increase ...

- Distribution shifts to the right
- Distribution becomes wider

Cai et al., Threshold Voltage Distribution in MLC NAND Flash Memory, DATE 2013.

Threshold Voltage Distribution Model

- Vth distribution can be modeled with ~95% accuracy as a Gaussian distribution with additive white noise

- Distortion in Vth over P/E cycles can be modeled and predicted as an exponential function of P/E cycles
    - With more than 95% accuracy

More Detail on Threshold Voltage Model

- Yu Cai, Erich F. Haratsch, Onur Mutlu, and Ken Mai,
"Threshold Voltage Distribution in MLC NAND Flash Memory: Characterization, Analysis and Modeling"
Proceedings of the
Design, Automation, and Test in Europe Conference
(**DATE**), Grenoble, France, March 2013. Slides (ppt)
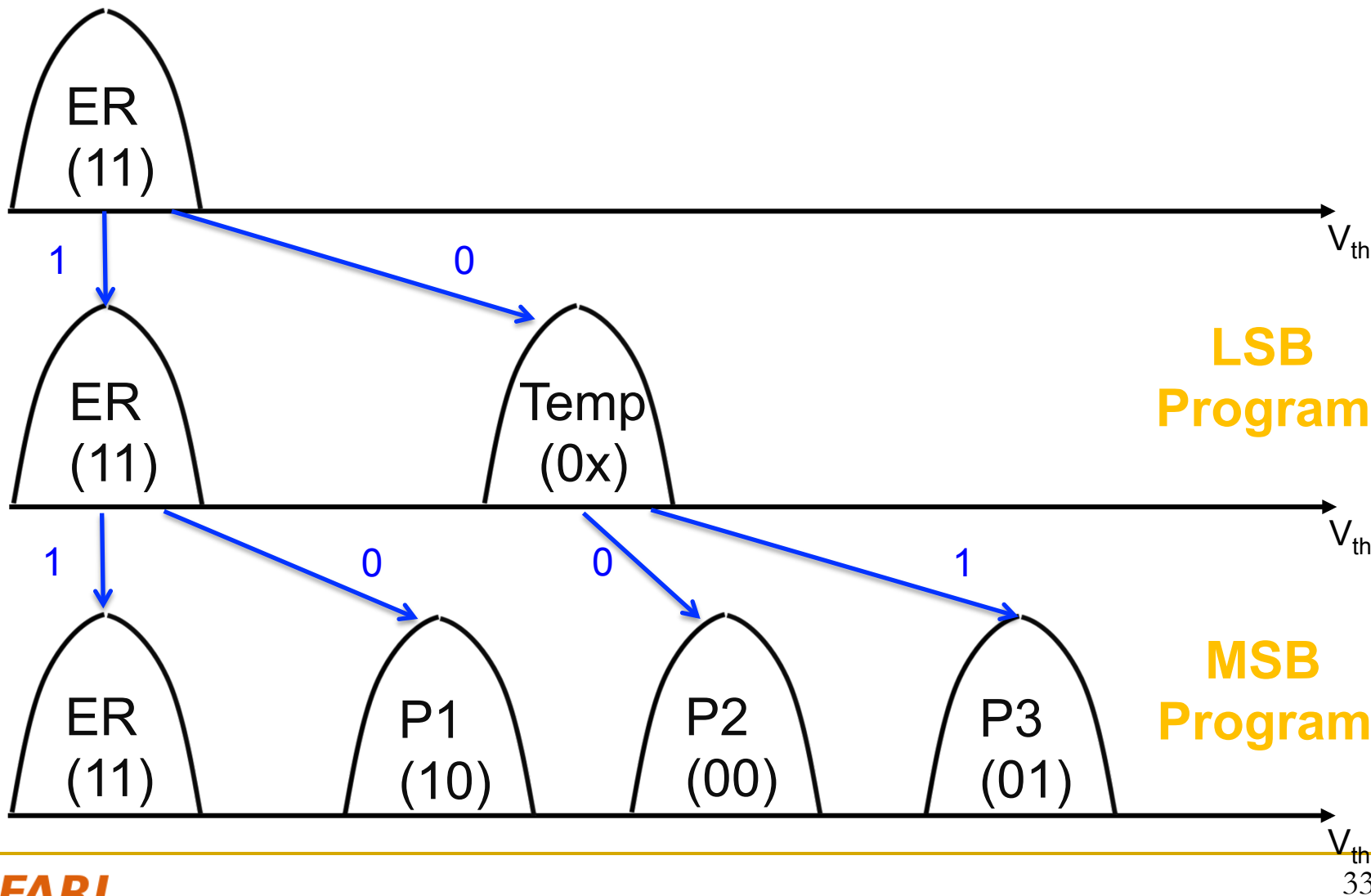
SAFARI

Program Interference Errors

- When a cell is being programmed, voltage level of a neighboring cell changes (unintentionally) due to parasitic capacitance coupling

  → can change the data value stored

- Also called program interference error

- Causes neighboring cell voltage to increase (shift right)

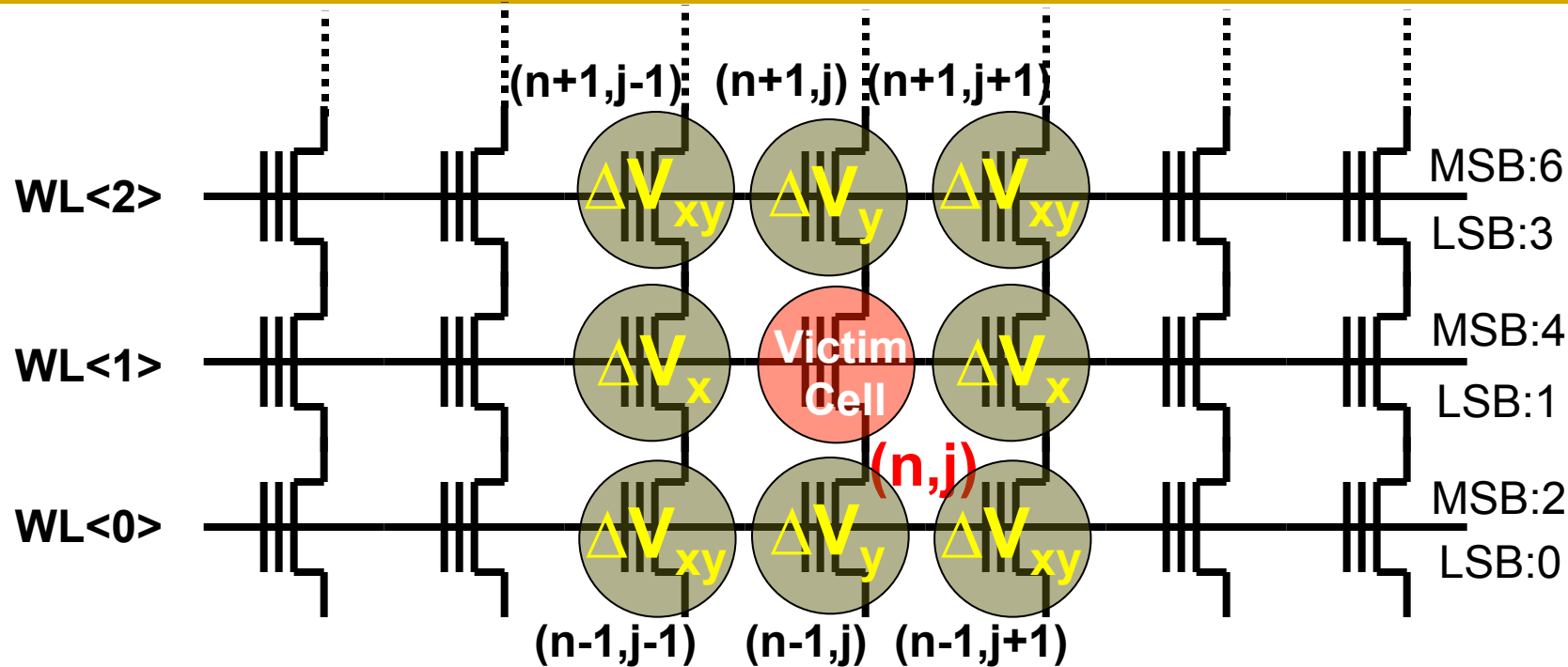- Once retention errors are minimized, these errors can become dominant
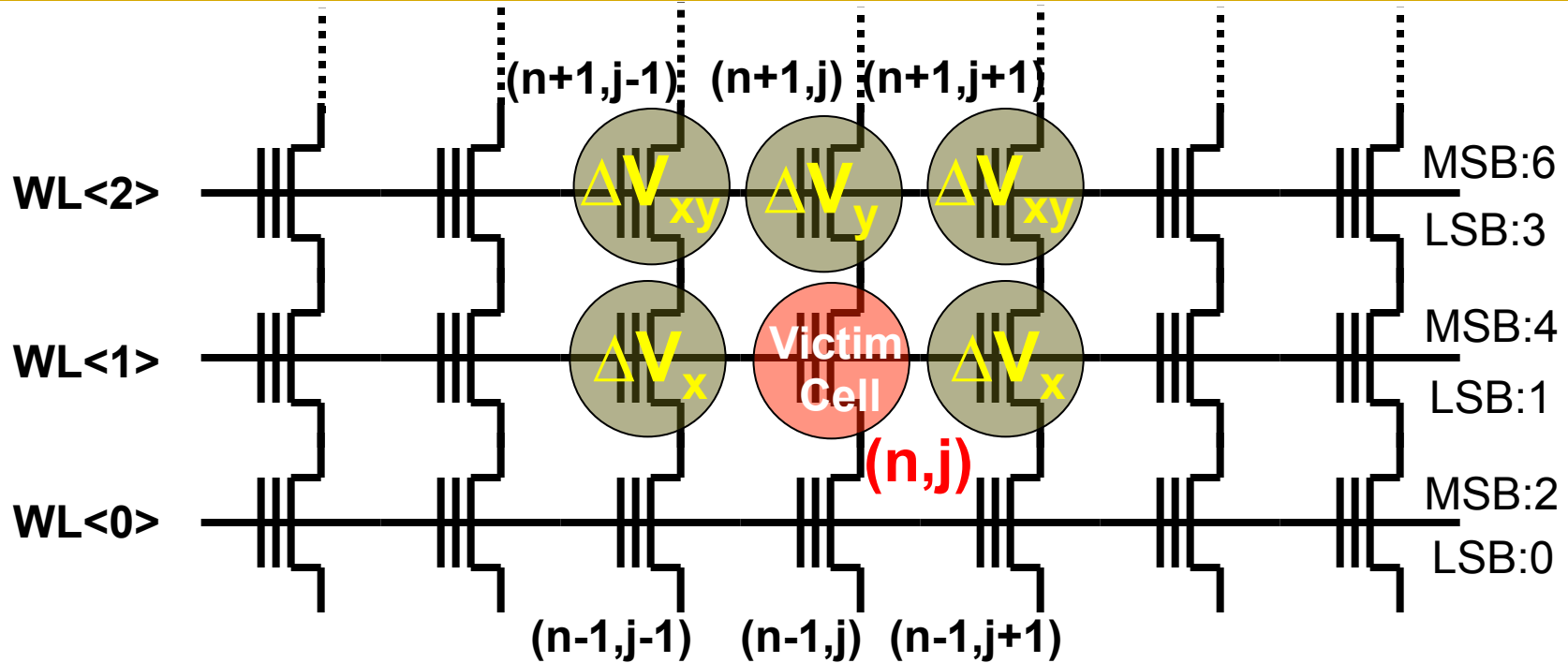
How Current Flash Cells are Programmed

- Programming 2-bit MLC NAND flash memory in two steps

ER
(11)

V_{th}

1

0

ER
(11)

Temp
(0x)

V_{th}

LSB Program

1

0

0

1

ER
(11)

P1
(10)

P2
(00)

P3
(01)

V_{th}

MSB Program

Basics of Program Interference

Traditional Model for Vth Change

- Traditional model for victim cell threshold voltage change

$$\Delta V_{victim} = (2 C_x \Delta V_x + C_y \Delta V_y + 2 C_{xy} \Delta V_{xy}) / C_{total}$$

Not accurate and requires knowledge of coupling caps!

SAFARI

Our Goal and Idea

- Develop a new, more accurate and easier to implement model for program interference

- Idea:
  - Empirically characterize and model the effect of neighbor cell Vth changes on the Vth of the victim cell
  - Fit neighbor Vth change to a linear regression model and find the coefficients of the model via empirical measurement

$$\Delta V_{victim}(n, j) = \sum_{y=j-K}^{j+K} \sum_{x=n+1}^{n=M} \alpha(x, y) \Delta V_{neighbor}(x, y) + \alpha_0 V_{victim}^{before}(n, j)$$

Can be measured

Developing a New Model via Empirical Measurement

- **Feature extraction for V_{th} changes based on characterization**
  - Threshold voltage changes on aggressor cell
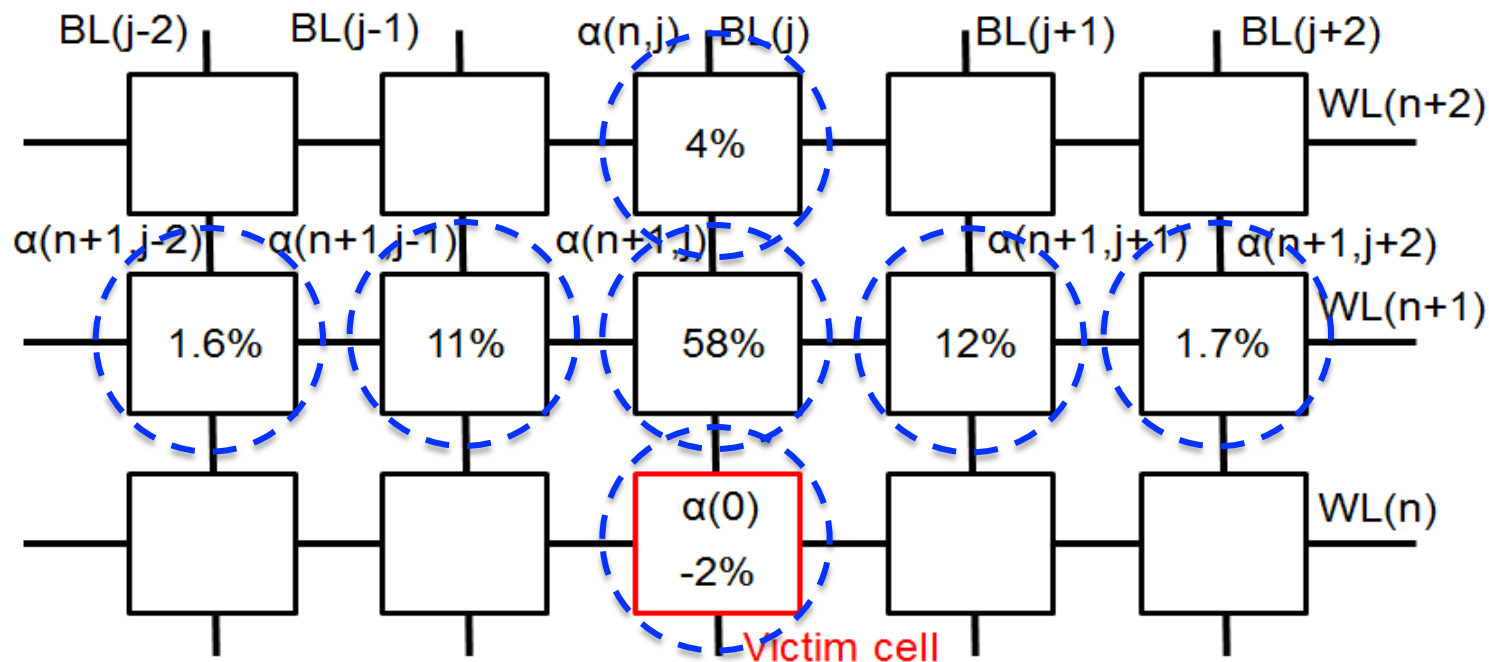 - Original state of victim cell

- **Enhanced linear regression model**

$$\Delta V_{victim}(n,j) = \sum_{y=j-K}^{j+K} \sum_{x=n+1}^{n=M} \alpha(x,y) \Delta V_{neighbor}(x,y) + \alpha_0 V_{victim}^{before}(n,j)$$

$$Y = X\alpha + \varepsilon \quad \longleftarrow \quad \text{(vector expression)}$$

- **Maximum likelihood estimation of the model coefficients**

$$\arg\min_{\alpha}(\|X \times \alpha - Y\|_2^2 + \lambda\|\alpha\|_1)$$

Effect of Neighbor Voltages on the Victim

- Immediately-above cell interference is dominant
- Immediately-diagonal neighbor is the second dominant
- Far neighbor cell interference exists
- Victim cell's Vth has negative effect on interference

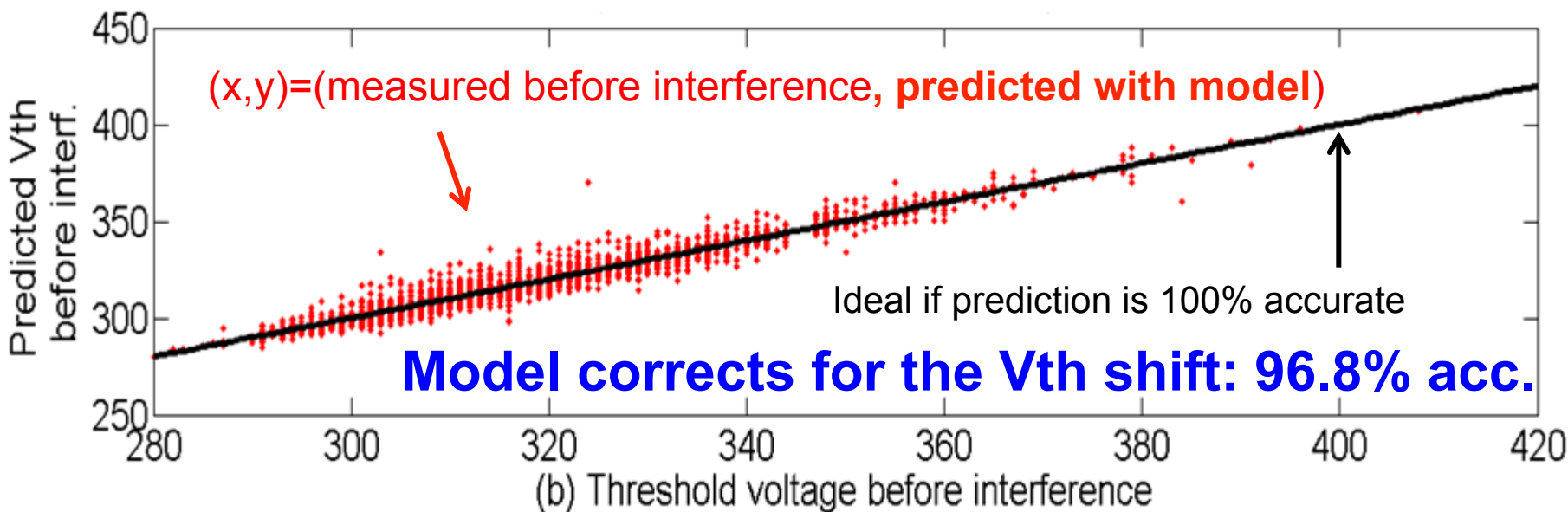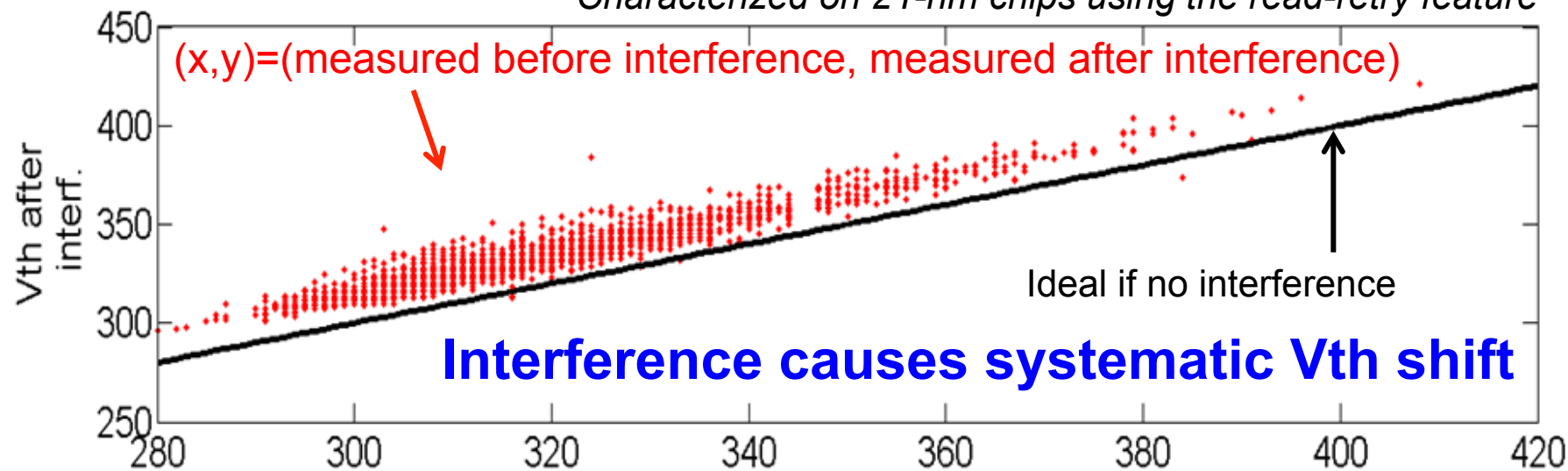Cai et al., Program Interference in MLC NAND Flash Memory, ICCD 2013

SAFARI

New Model for Program Interference
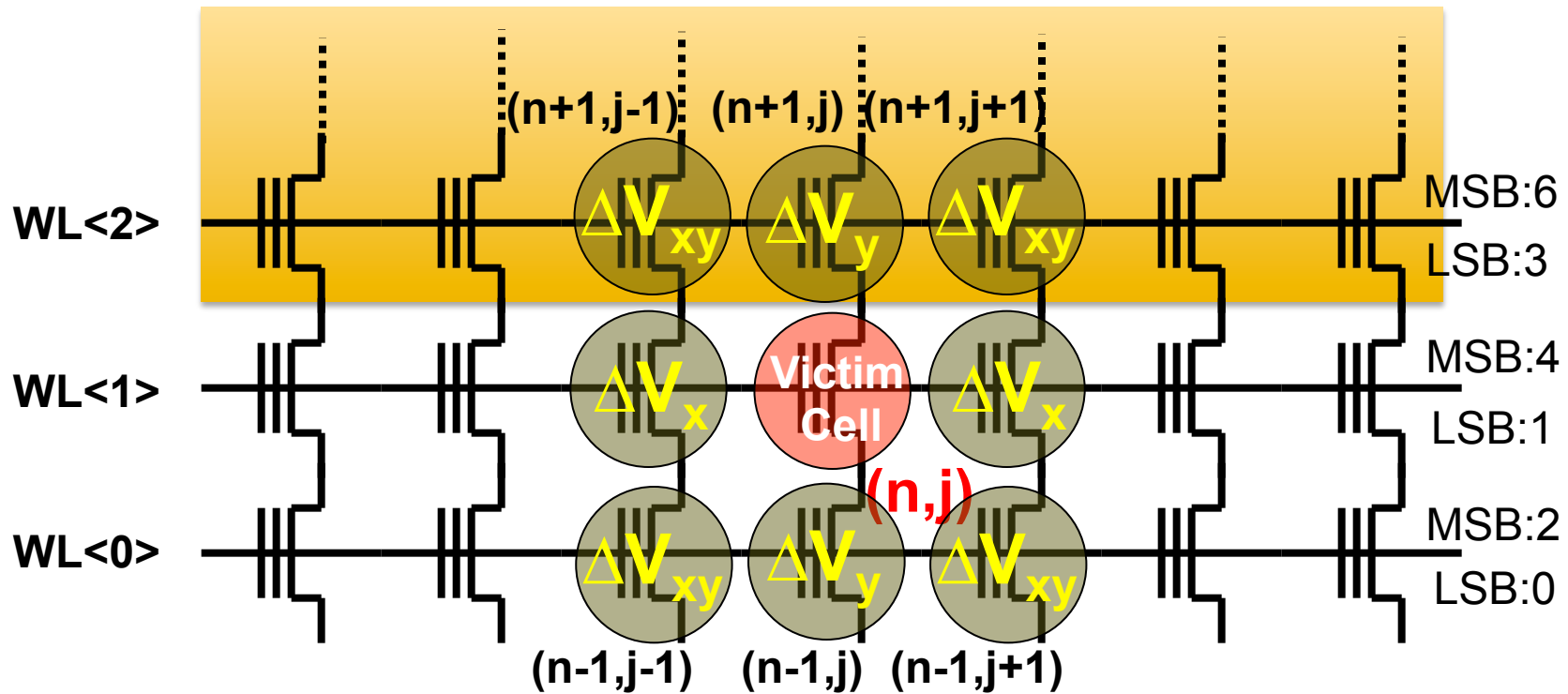
$$\Delta V_{victim}(n,j) = \sum_{y=j-K}^{j+K} \sum_{x=n+1}^{n+M} \alpha(x,y) \Delta V_{neighbor}(x,y) + \alpha_0 V_{victim}^{before}(n,j)$$

Model Accuracy

Characterized on 2Y-nm chips using the read-retry feature

(x,y)=(measured before interference, measured after interference)

Ideal if no interference

Interference causes systematic Vth shift

(x,y)=(measured before interference, **predicted with model**)

Ideal if prediction is 100% accurate

Model corrects for the Vth shift: 96.8% acc.

(b) Threshold voltage before interference

SAFARI

Many Other Results in the Paper

- Yu Cai, Onur Mutlu, Erich F. Haratsch, and Ken Mai,
  "Program Interference in MLC NAND Flash Memory: Characterization, Modeling, and Mitigation"
  Proceedings of the
  31st IEEE International Conference on Computer Design
  (**ICCD**), Asheville, NC, October 2013. Slides (pptx) (pdf)
  Lightning Session Slides (pdf)

SAFARI

Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
  - Main Characterization Results
  - Retention-Aware Error Management
  - Threshold Voltage and Program Interference Analysis
  - Read Reference Voltage Prediction
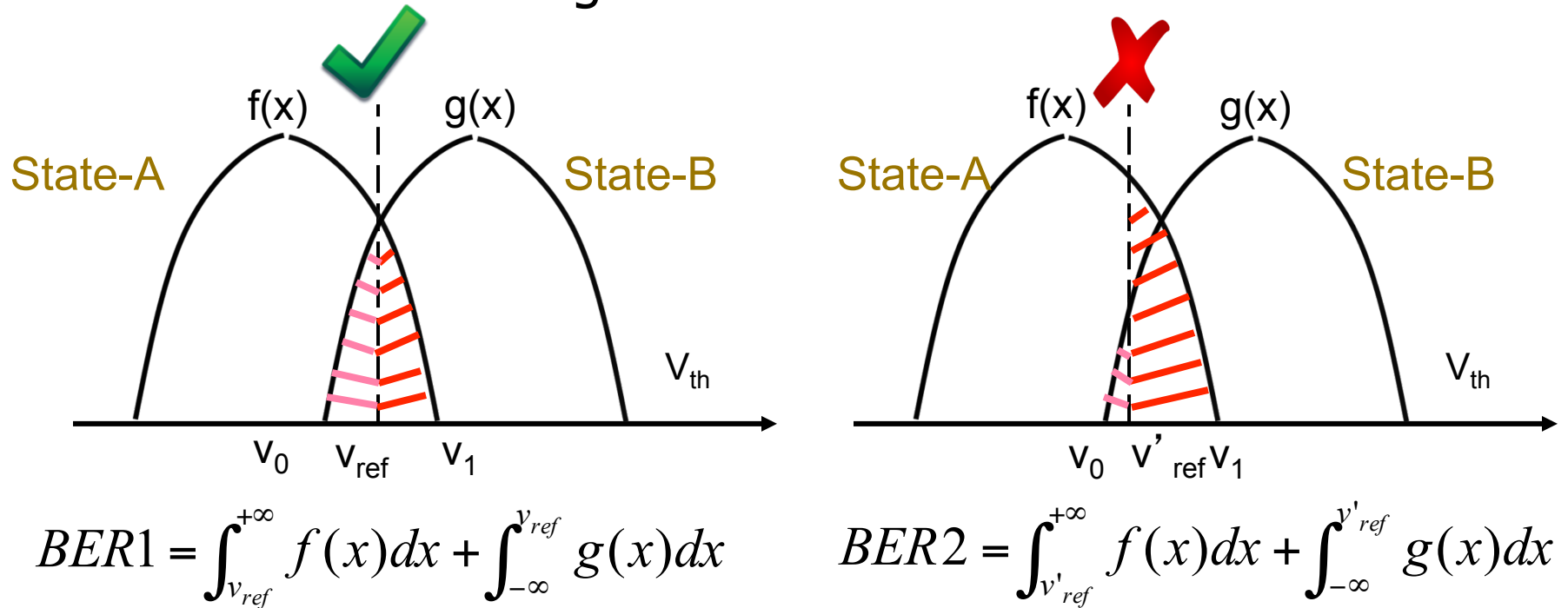 - Neighbor-Assisted Error Correction
- Summary

SAFARI

Mitigation: Applying the Model

- So, what can we do with the model?

- Goal: Mitigate the effects of program interference caused voltage shifts

Optimum Read Reference for Flash Memory

- Read reference voltage affects the raw bit error rate

$$BER1 = \int_{v_{ref}}^{+\infty} f(x)dx + \int_{-\infty}^{v_{ref}} g(x)dx \qquad BER2 = \int_{v'_{ref}}^{+\infty} f(x)dx + \int_{-\infty}^{v'_{ref}} g(x)dx$$
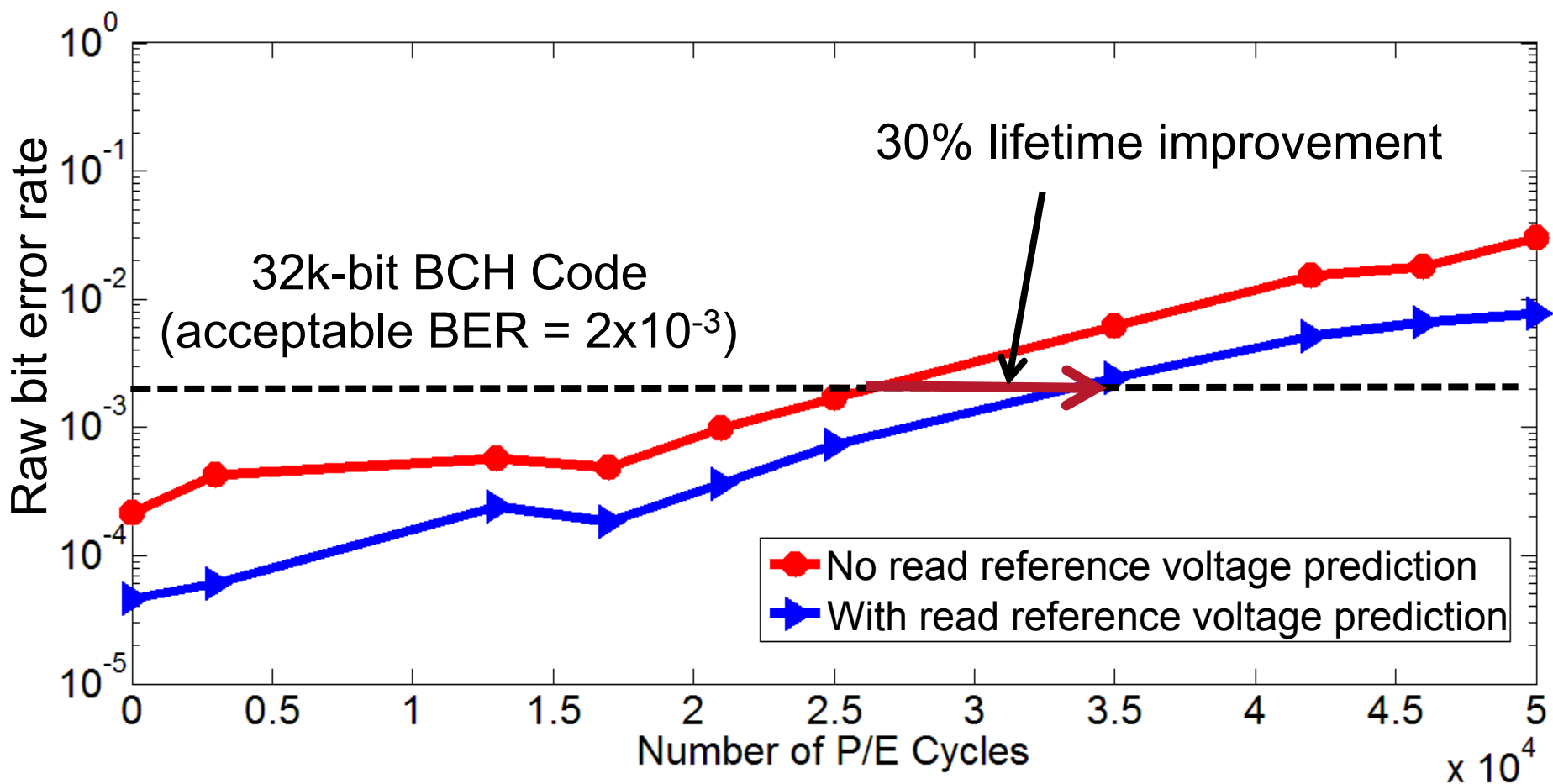
- There exists an optimal read reference voltage
  - Predictable if the statistics (i.e. mean, variance) of threshold voltage distributions are characterized and modeled

Optimum Read Reference Voltage Prediction
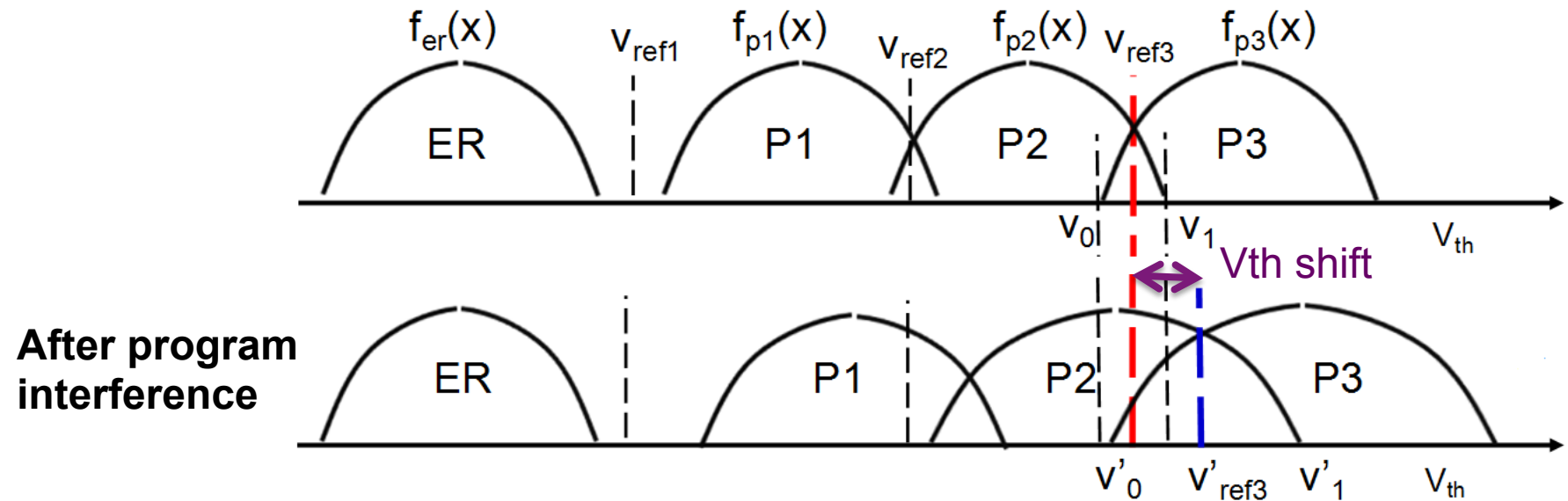
- ## Vth shift learning (done every ~1k P/E cycles)
  - Program sample cells with known data pattern and test Vth
  - Program aggressor neighbor cells and test victim Vth after interference
  - Characterize the mean shift in Vth (i.e., program interference noise)

- ## Optimum read reference voltage prediction
  - Default read reference voltage + Predicted mean Vth shift by model

Effect of Read Reference Voltage Prediction

32k-bit BCH Code
(acceptable BER = 2×10^{-3})

30% lifetime improvement

No read reference voltage prediction
With read reference voltage prediction

Number of P/E Cycles

Raw bit error rate

- Read reference voltage prediction reduces raw BER (by 64%) and increases the P/E cycle lifetime (by 30%)

More on Read Reference Voltage Prediction

- Yu Cai, Onur Mutlu, Erich F. Haratsch, and Ken Mai,
"Program Interference in MLC NAND Flash Memory: Characterization, Modeling, and Mitigation"
Proceedings of the
31st IEEE International Conference on Computer Design
(***ICCD***), Asheville, NC, October 2013. Slides (pptx) (pdf)
Lightning Session Slides (pdf)

SAFARI

Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
    - Main Characterization Results
    - Retention-Aware Error Management
    - Threshold Voltage and Program Interference Analysis
    - Read Reference Voltage Prediction
    - Neighbor-Assisted Error Correction
- Summary

Goal

- Develop a better error correction mechanism for cases where ECC fails to correct a page

Observations So Far

- **Immediate neighbor cell has the most effect on the victim cell** when programmed

- **A single set of read reference voltages is used** to determine the value of the (victim) cell

- The set of read reference voltages is determined based on the ***overall threshold voltage distribution of all cells*** in flash memory

New Observations [Cai+ SIGMETRICS'14]

- Vth distributions of **cells with different-valued immediate-neighbor cells** are significantly different
  - Because neighbor value affects the amount of Vth shift

- Corollary: If we know the value of the immediate-neighbor, we can find a more accurate set of read reference voltages based on the "conditional" threshold voltage distribution

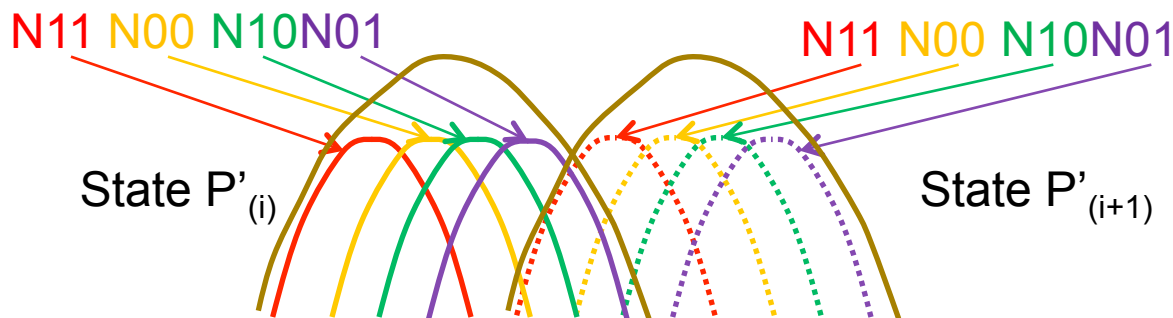Cai et al., Neighbor-Cell Assisted Error Correction for MLC NAND Flash Memories, SIGMETRICS 2014.

Secrets of Threshold Voltage Distributions

Aggressor WL $\boxed{11}$ $\boxed{10}$ $\boxed{01}$ $\boxed{00}$ $\boxed{01}$ $\boxed{10}$ $\boxed{11}$ $\boxed{00}$ ......
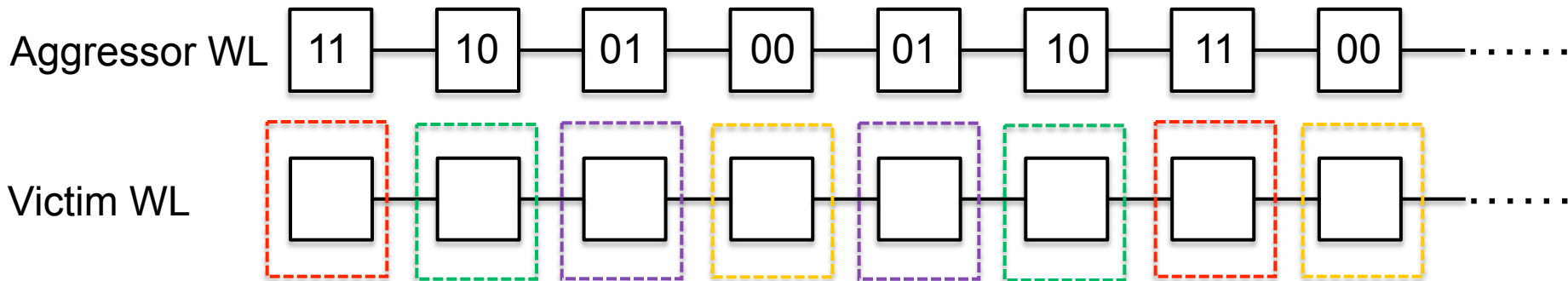
Victim WL ......

Victim WL **before** MSB page of aggressor WL are programmed

State $P_{(i)}$ State $P_{(i+1)}$

Victim WL **after** MSB page of aggressor WL are programmed

N11 N00 N10 N01   N11 N00 N10 N01

State $P'_{(i)}$ State $P'_{(i+1)}$
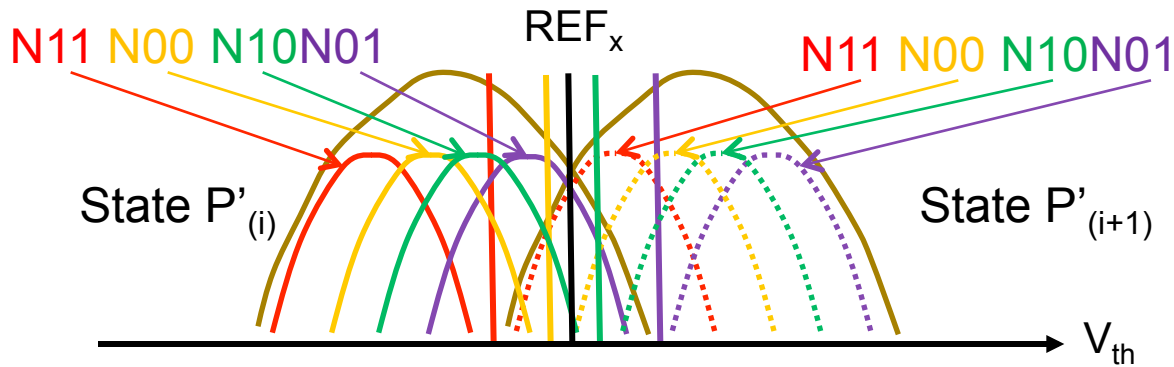
# If We Knew the Immediate Neighbor …

- Then, we could choose a different read reference voltage to more accurately read the "victim" cell

Overall vs Conditional Reading

- Using the optimum read reference voltage based on the overall distribution leads to more errors

- Better to use the optimum read reference voltage based on the conditional distribution (i.e., value of the neighbor)
  - Conditional distributions of two states are farther apart from each other

SAFARI

Measurement Results
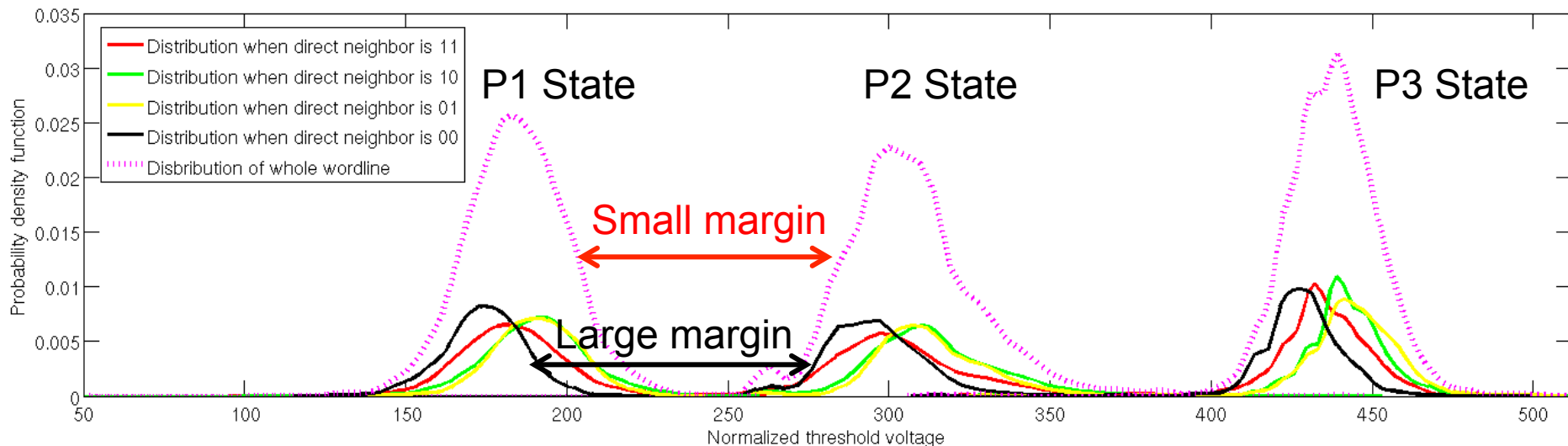
	Overall	x_{11} (ER)	x_{10} (P1)	x_{00} (P2)	X_{01} (P3)
Distance	65.4	65.4	64.7	66.4	65.8
Variance	385.9	286.2	256.7	242.8	252.1
SNR	3.4	3.8	3.9	4.2	4.1
BER	3×10^{-4}	7×10^{-5}	5×10^{-5}	2×10^{-5}	3×10^{-5}

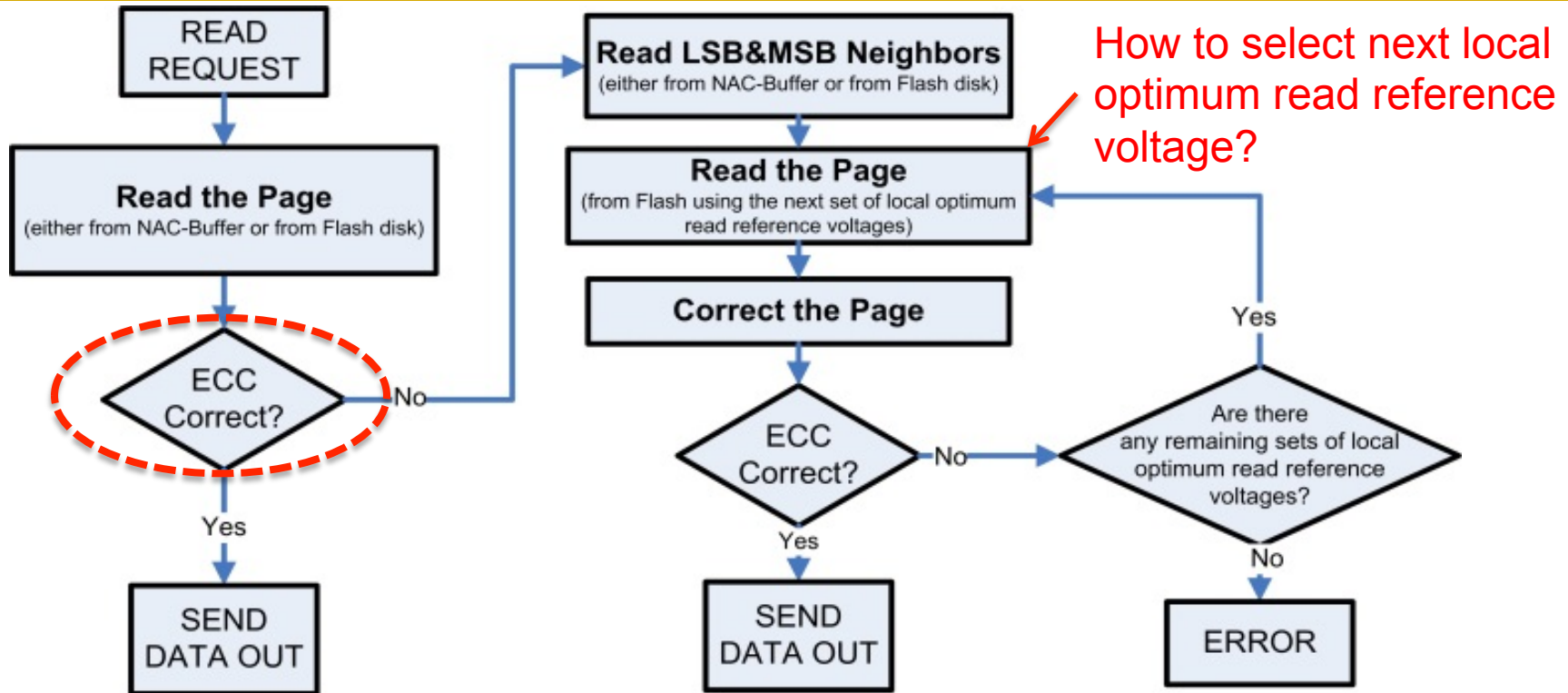**Raw BER of conditional reading is much smaller than overall reading**

Idea: Neighbor Assisted Correction (NAC)

- Read a page with the read reference voltages based on overall Vth distribution (same as today) and buffer it

- If ECC fails:
  - Read the immediate-neighbor page
  - Re-read the page using the read reference voltages corresponding to the voltage distribution assuming a particular immediate-neighbor value
  - Replace the buffered values of the cells with that particular immediate-neighbor cell value
 - Apply ECC again

SAFARI

Neighbor Assisted Correction Flow

READ REQUEST → **Read the Page** (either from NAC-Buffer or from Flash disk) → **ECC Correct?**
- Yes → **SEND DATA OUT**
- No → **Read LSB&MSB Neighbors** (either from NAC-Buffer or from Flash disk)

Read LSB&MSB Neighbors (either from NAC-Buffer or from Flash disk) → **Read the Page** (from Flash using the next set of local optimum read reference voltages) → **Correct the Page** → **ECC Correct?**
- Yes → **SEND DATA OUT**
- No → **Are there any remaining sets of local optimum read reference voltages?**
  - Yes → **Read the Page** (loop back)
  - No → **ERROR**

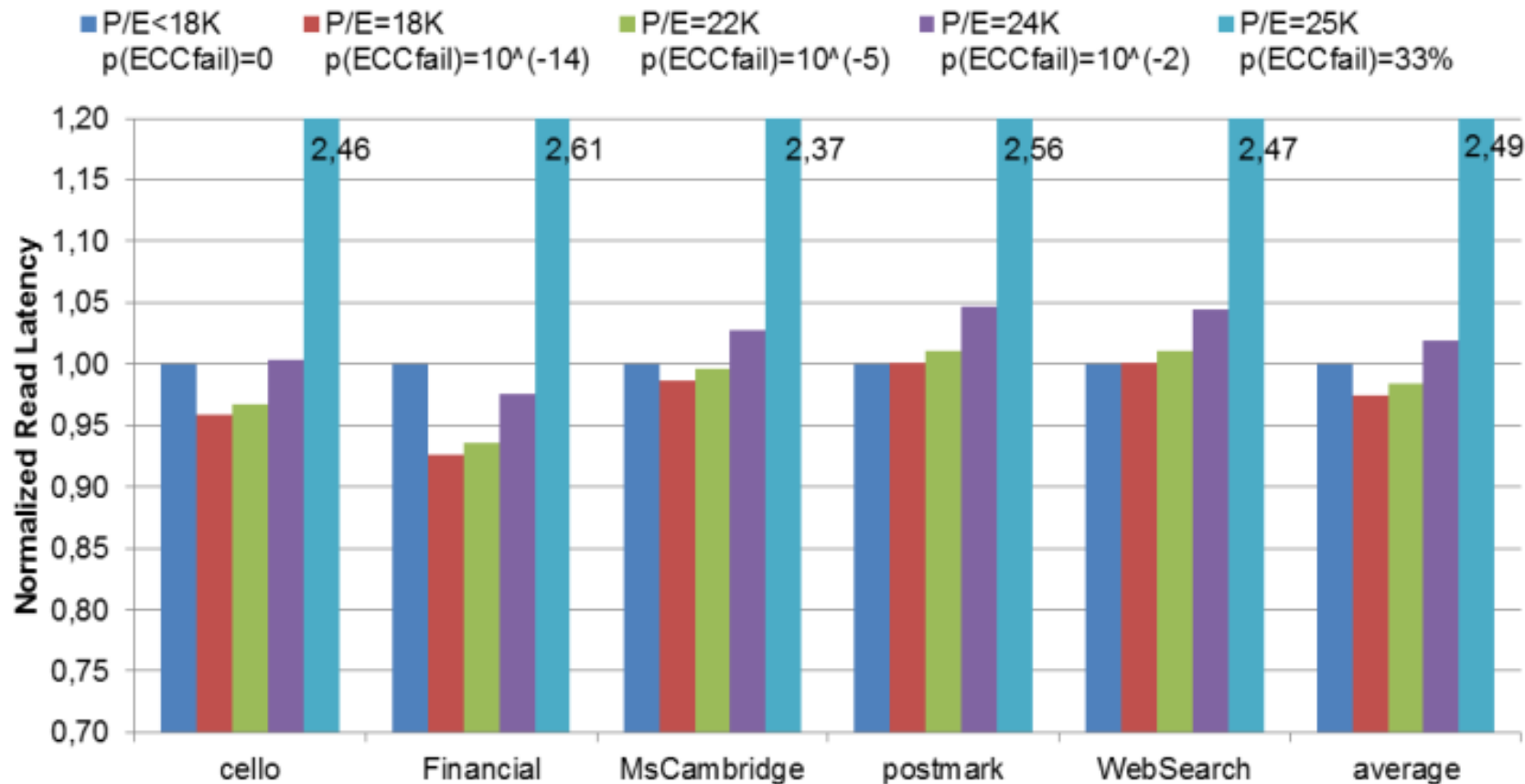*How to select next local optimum read reference voltage?*

- Trigger neighbor-assisted reading only when ECC fails
- Read neighbor values and use corresponding read reference voltages in a prioritized order until ECC passes

Lifetime Extension with NAC

33% lifetime improvement at no performance loss
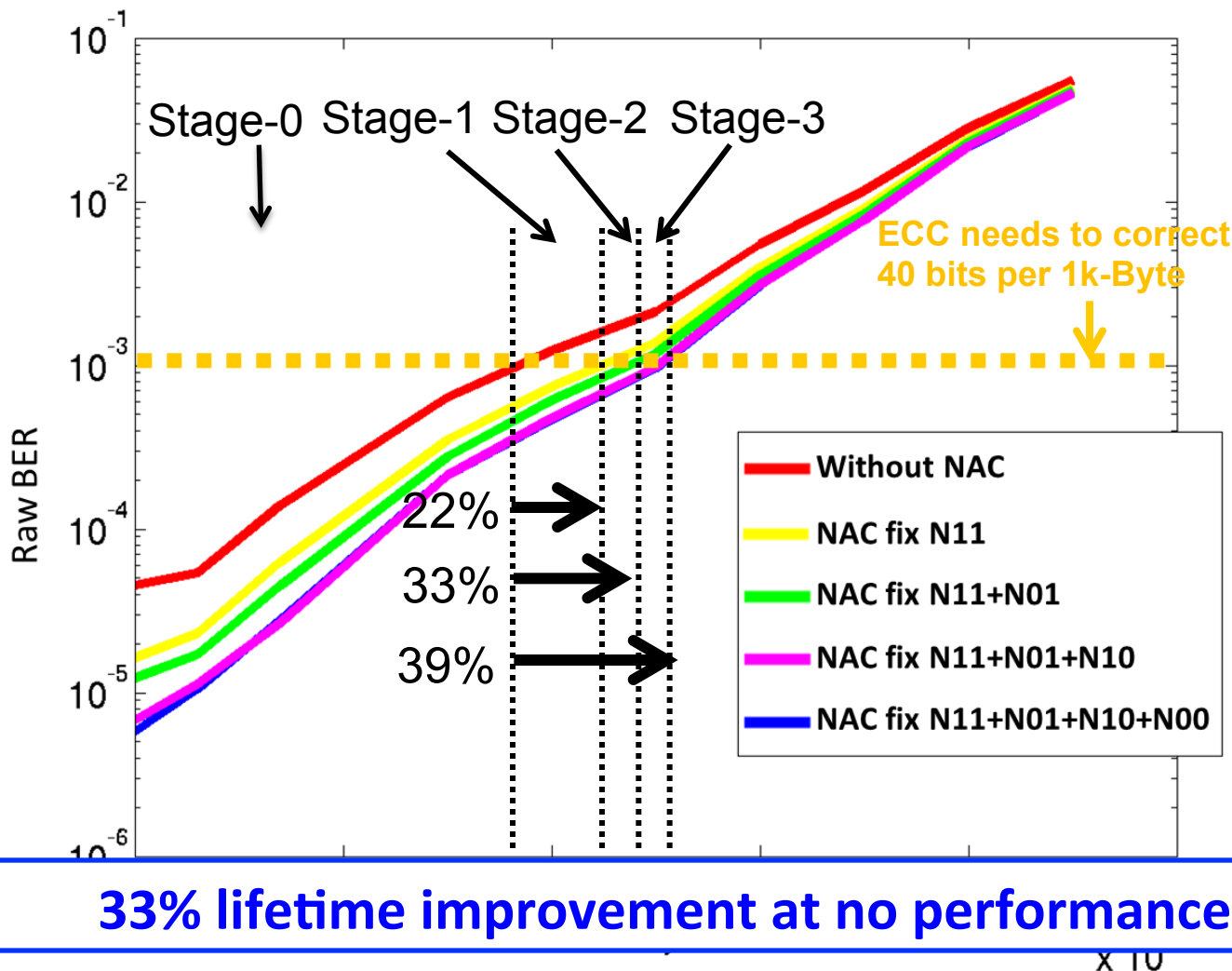
Performance Analysis of NAC

No performance loss within nominal lifetime
and with reasonable (1%) ECC fail rates

More on Neighbor-Assisted Correction

- Yu Cai, Gulay Yalcin, Onur Mutlu, Eric Haratsch, Osman Unsal, Adrian Cristal, and Ken Mai, **"Neighbor-Cell Assisted Error Correction for MLC NAND Flash Memories"** *Proceedings of the ACM International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS)*, Austin, TX, June 2014. Slides (ppt) (pdf)

SAFARI

Agenda

- Background, Motivation and Approach
- Experimental Characterization Methodology
- Error Analysis and Management
    - Main Characterization Results
    - Retention-Aware Error Management
    - Threshold Voltage and Program Interference Analysis
    - Read Reference Voltage Prediction
    - Neighbor-Assisted Error Correction
- Summary

Executive Summary

- Problem: MLC NAND flash memory reliability/endurance is a key challenge for satisfying future storage systems' requirements

- We are: (1) Building reliable error models for NAND flash memory via experimental characterization, (2) Developing efficient techniques to improve reliability and endurance

- This talk provided a "flash" summary of our recent results published in the past 3 years:
  - Experimental error and threshold voltage characterization **[DATE'12&13]**
  - Retention-aware error management **[ICCD'12]**
  - Program interference analysis and read reference V prediction **[ICCD'13]**
  - Neighbor-assisted error correction **[SIGMETRICS'14]**

Readings (I)

- Yu Cai, Erich F. Haratsch, Onur Mutlu, and Ken Mai,
"Error Patterns in MLC NAND Flash Memory: Measurement, Characterization, and Analysis"
Proceedings of the Design, Automation, and Test in Europe Conference (**DATE**), Dresden, Germany, March 2012. Slides (ppt)

- Yu Cai, Gulay Yalcin, Onur Mutlu, Erich F. Haratsch, Adrian Cristal, Osman Unsal, and Ken Mai,
"Flash Correct-and-Refresh: Retention-Aware Error Management for Increased Flash Memory Lifetime"
Proceedings of the 30th IEEE International Conference on Computer Design (**ICCD**), Montreal, Quebec, Canada, September 2012. Slides (ppt) (pdf)

- Yu Cai, Erich F. Haratsch, Onur Mutlu, and Ken Mai,
"Threshold Voltage Distribution in MLC NAND Flash Memory: Characterization, Analysis and Modeling"
Proceedings of the Design, Automation, and Test in Europe Conference (**DATE**), Grenoble, France, March 2013. Slides (ppt)

Readings (II)

- Yu Cai, Gulay Yalcin, Onur Mutlu, Erich F. Haratsch, Adrian Cristal, Osman Unsal, and Ken Mai,
"Error Analysis and Retention-Aware Error Management for NAND Flash Memory"
Intel Technology Journal (**ITJ**) *Special Issue on Memory Resiliency*, Vol. 17, No. 1, May 2013.

- Yu Cai, Onur Mutlu, Erich F. Haratsch, and Ken Mai,
"Program Interference in MLC NAND Flash Memory: Characterization, Modeling, and Mitigation"
Proceedings of the 31st IEEE International Conference on Computer Design (**ICCD**), Asheville, NC, October 2013. Slides (pptx) (pdf) Lightning Session Slides (pdf)

- Yu Cai, Gulay Yalcin, Onur Mutlu, Eric Haratsch, Osman Unsal, Adrian Cristal, and Ken Mai,
"Neighbor-Cell Assisted Error Correction for MLC NAND Flash Memories"
Proceedings of the
ACM International Conference on Measurement and Modeling of Computer Systems (**SIGMETRICS**), Austin, TX, June 2014. Slides (ppt) (pdf)

Referenced Papers

- All are available at

  http://users.ece.cmu.edu/~omutlu/projects.htm

Related Videos and Course Materials

- **Computer Architecture Lecture Videos on Youtube**
  - https://www.youtube.com/playlist?list=PL5PHm2jkkXmidJOd59REog9jDnPDTG6IJ

- **Computer Architecture Course Materials**
  - http://www.ece.cmu.edu/~ece447/s13/doku.php?id=schedule

- **Advanced Computer Architecture Course Materials**
  - http://www.ece.cmu.edu/~ece740/f13/doku.php?id=schedule

- **Advanced Computer Architecture Lecture Videos on Youtube**
  - https://www.youtube.com/playlist?list=PL5PHm2jkkXmgDN1PLwOY_tGtUlynnyV6D

Thank you.

Feel free to email me with any questions & feedback

onur@cmu.edu
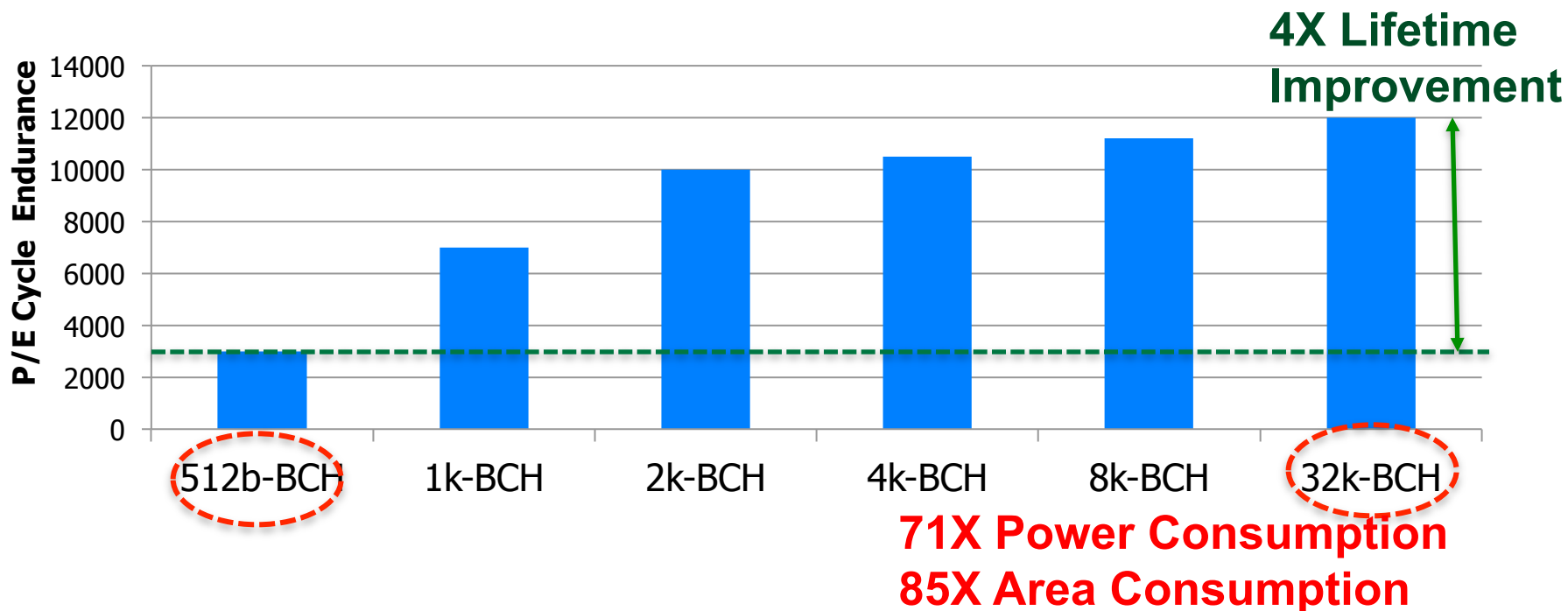
http://users.ece.cmu.edu/~omutlu/

Additional Slides

Error Types and Testing Methodology

- Erase errors
  - Count the number of cells that fail to be erased to "11" state

- Program interference errors
  - Compare the data immediately after page programming and the data after the whole block being programmed

- Read errors
  - Continuously read a given block and compare the data between consecutive read sequences

- Retention errors
  - Compare the data read after an amount of time to data written
    - Characterize short term retention errors under room temperature
    - Characterize long term retention errors by baking in the oven under 125℃

SAFARI

Improving Flash Lifetime with Strong ECC

- Lifetime improvement comparison of various BCH codes

4X Lifetime Improvement

P/E Cycle Endurance (y-axis: 0, 2000, 4000, 6000, 8000, 10000, 12000, 14000)

Bars: 512b-BCH, 1k-BCH, 2k-BCH, 4k-BCH, 8k-BCH, 32k-BCH

71X Power Consumption
85X Area Consumption

Strong ECC is very inefficient at improving lifetime

Our Goal

Develop new techniques
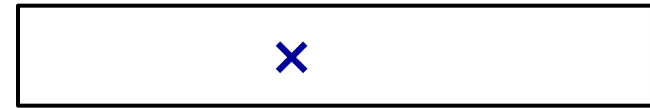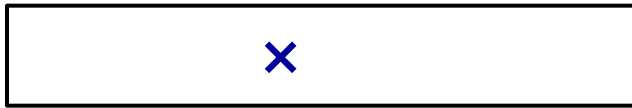
to improve flash lifetime
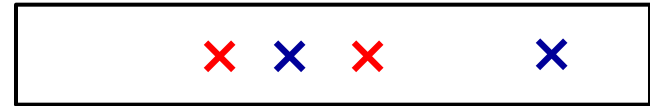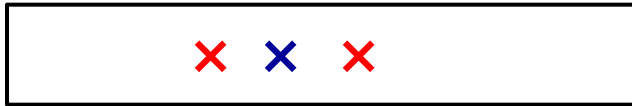
without relying on stronger ECC

FCR Intuition

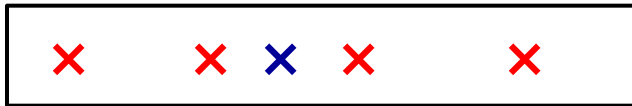|	Errors with No refresh	Errors with Periodic refresh
Program Page	×	×
After time T	× × ×	× × × ×
After time 2T	× × × × ×	× × × ×
After time 3T	× × × × × × ×	× × × ×
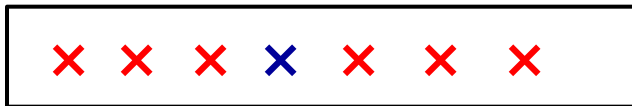
× **Retention Error**   × **Program Error**

FCR Lifetime Evaluation Takeaways

- **Significant average lifetime improvement over no refresh**
  - Adaptive-rate FCR: 46X
  - Hybrid reprogramming/remapping based FCR: 31X
  - Remapping based FCR: 9X

- **FCR lifetime improvement larger than that of stronger ECC**
  - 46X vs. 4X with 32-kbit ECC (over 512-bit ECC)
  - FCR is less complex and less costly than stronger ECC

- **Lifetime on all workloads improves with Hybrid FCR**
  - Remapping based FCR can degrade lifetime on read-heavy WL
  - Lifetime improvement highest in write-heavy workloads

Characterizing Cell Threshold w/ Read Retry

Erased State   Programmed States
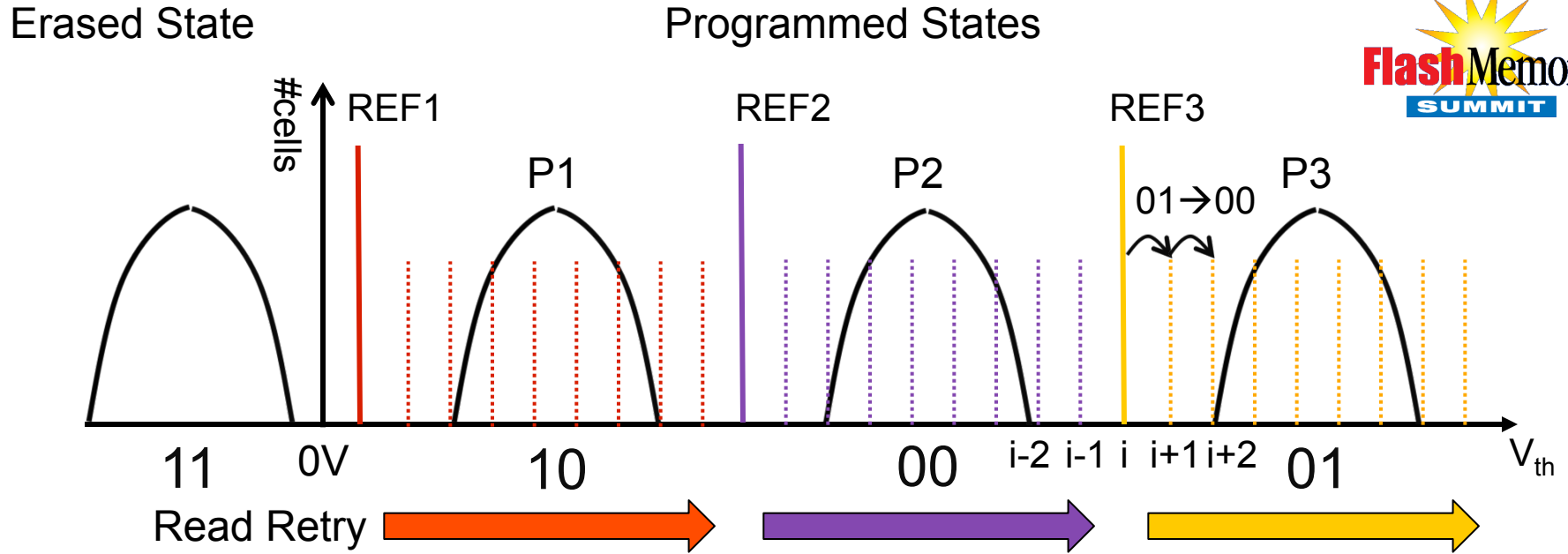
- **Read-retry feature of new NAND flash**
  - Tune read reference voltage and check which V_{th} region of cells
- **Characterize the threshold voltage distribution of flash cells in programmed states through Monte-Carlo emulation**

- **Parametric distribution**
  - Closed-form formula, only a few number of parameters to be stored
- **Exponential distribution family**

  Distribution parameter vector

  $$p(\mathbf{x}|\boldsymbol{\eta}) = h(\mathbf{x})g(\boldsymbol{\eta})\exp\left\{\boldsymbol{\eta}^{\mathrm{T}}\mathbf{u}(\mathbf{x})\right\}$$

- **Maximum likelihood estimation (MLE) to learn parameters**

  Observed testing data

  Likelihood Function
  $$p(\mathbf{X}|\boldsymbol{\eta}) = \left(\prod_{n=1}^{N} h(\mathbf{x}_n)\right) g(\boldsymbol{\eta})^N \exp\left\{\boldsymbol{\eta}^{\mathrm{T}}\sum_{n=1}^{N}\mathbf{u}(\mathbf{x}_n)\right\}$$

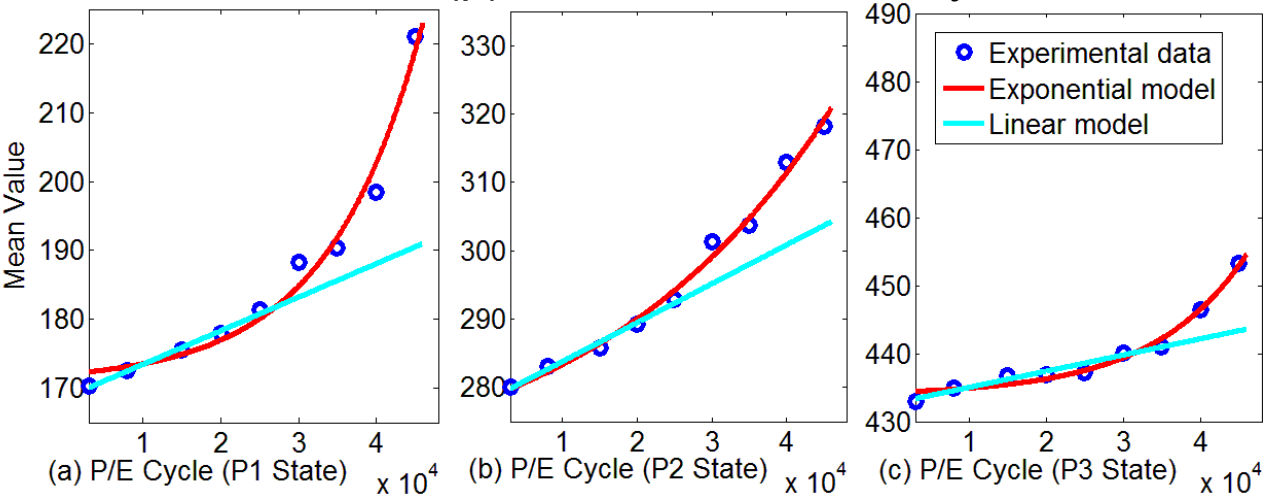**Goal of MLE**: Find distribution parameters to maximize likelihood function

Distribution $p(x\mid\eta)$		Parameters
Gaussian	$\dfrac{1}{\sigma\sqrt{2\pi}}\exp\{-\dfrac{1}{2}(\dfrac{x-\mu}{\sigma})^2\}$	Mean: $\boldsymbol{\mu}$ Var: $\boldsymbol{\sigma^2}$
Beta	$\dfrac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}x^{\alpha-1}(1-x)^{\beta-1}$	Mean: $\boldsymbol{\alpha/(\alpha+\beta)}$ Var: $\boldsymbol{\alpha\beta/((\alpha+\beta)^2(\alpha+\beta+1))}$
Gamma	$\dfrac{1}{\theta^k}\dfrac{1}{\Gamma(k)}x^{k-1}e^{-\frac{x}{\theta}}$	Mean: $\boldsymbol{k\theta}$ Var: $\boldsymbol{k\theta^2}$
Log-normal	$\dfrac{1}{x\sigma\sqrt{2\pi}}\exp\{-\dfrac{(\ln x-\mu)^2}{2\sigma^2}\}$	Mean: $\boldsymbol{\exp(\mu+\sigma^2/2)}$ Var:$\boldsymbol{(\exp(\sigma^2)-1)\exp(2\mu+\sigma^2)}$
Weibull	$\dfrac{k}{\lambda}(\dfrac{x}{\lambda})^{k-1}\exp\{-(\dfrac{x}{\lambda})^k\}$	Mean: $\boldsymbol{\lambda\Gamma(1+1/k)}$ Var: $\boldsymbol{\lambda^2\Gamma(1+2/k)-\mu^2}$
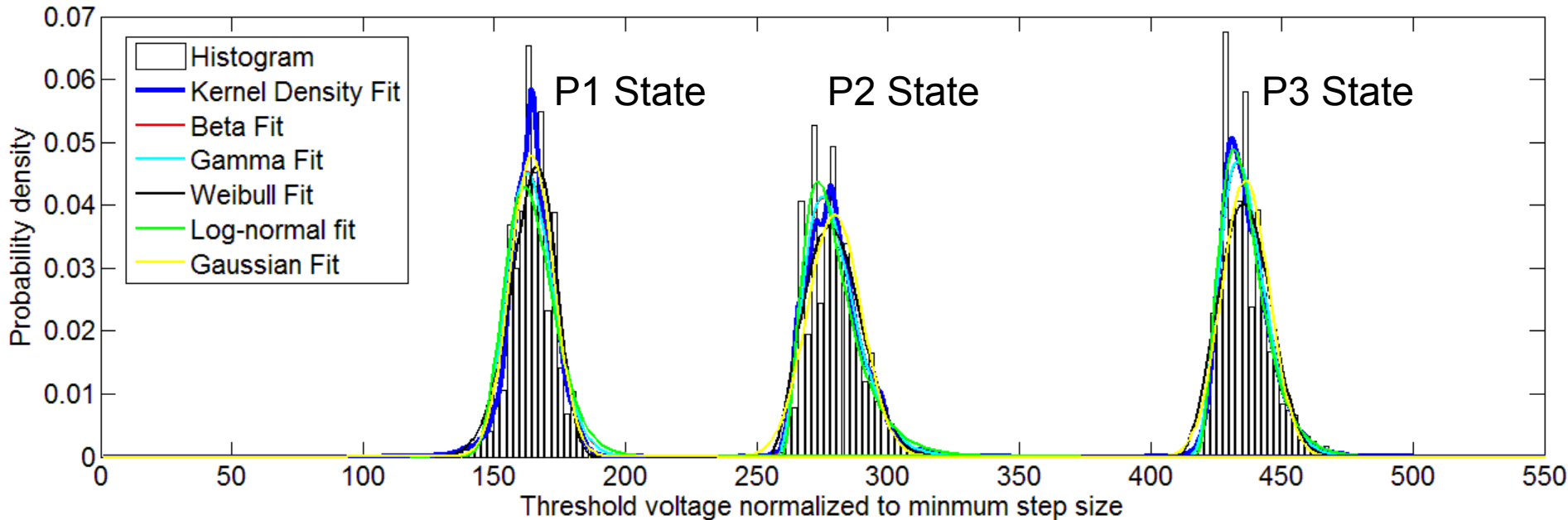
	Beta	Gamma	Gaussian	Log-normal	Weibull
RMSE	19.5%	20.3%	22.1%	24.8%	28.6%

Distribution can be approx. modeled as Gaussian distribution

Cycling Noise Modeling

Mean value (μ) increases with P/E cycles

(a) P/E Cycle (P1 State) x 10⁴
(b) P/E Cycle (P2 State) x 10⁴
(c) P/E Cycle (P3 State) x 10⁴

Standard deviation value (σ) increases with P/E cycles

(a) P/E cycle (P1 State) x 10⁴
(b) P/E Cycle (P2 State) x 10⁴
(c) P/E Cycle P3 State x 10⁴

Exponential model

$$V_{th}^{mean,std}(PEcycle) = A + B \times e^{C \times PEcycle}$$
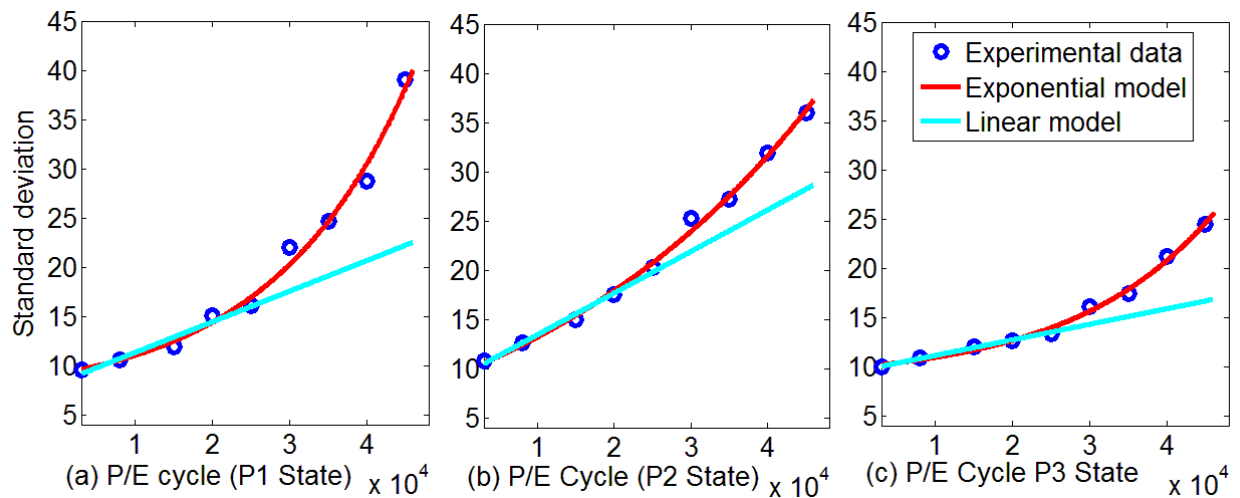
Linear model

$$V_{th}^{mean,std}(PEcycle) = D + E \times PEcycle$$

79

Conclusion & Future Work

- **P/E operations modeled as signal passing thru AWGN channel**
  - Approximately Gaussian with 22% distortion
  - P/E noise is white noise

- **P/E cycling noise affects threshold voltage distributions**
  - Distribution shifts to the right and widens around the mean value
  - Statistics (mean/variance) can be modeled as exponential correlation with P/E cycles with 95% accuracy

- **Future work**
  - Characterization and models for retention noise
  - Characterization and models for program interference noise

Program Interference: Key Findings

- **Methodology:** Extensive experimentation with real 2Y-nm MLC NAND Flash chips

- Amount of program interference is dependent on
  - Location of cells (programmed and victim)
  - Data values of cells (programmed and victim)
  - Programming order of pages

- Our new model can predict the amount of program interference with 96.8% prediction accuracy

- Our new read reference voltage prediction technique can improve flash lifetime by 30%

NAC: Executive Summary

- **Problem:** Cell-to-cell Program interference causes threshold voltage of flash cells to be distorted even they are originally programmed correctly

- **Our Goal:** Develop techniques to overcome cell-to-cell program interference
  - Analyze the threshold voltage distributions of flash cells *conditionally upon the values of immediately neighboring cells*
  - Devise new error correction mechanisms that can *take advantage of the values of neighboring cells* to reduce error rates over conventional ECC

- **Observations:** Wide overall distribution can be decoupled into multiple narrower conditional distributions which can be separated easily

- **Solution:** Neighbor-cell Assisted Correction (NAC)
  - Re-read a flash memory page that initially failed ECC with a set of read reference voltages corresponding to the conditional threshold voltage distribution
  - Use the re-read values to correct the cells that have neighbors with that value
  - Prioritize reading assuming neighbor cell values that cause largest or smallest cell-to-cell interference to allow ECC correct errors with less re-reads

- **Results:** NAC improves flash memory lifetime by 39%
  - Within nominal lifetime: no performance degradation
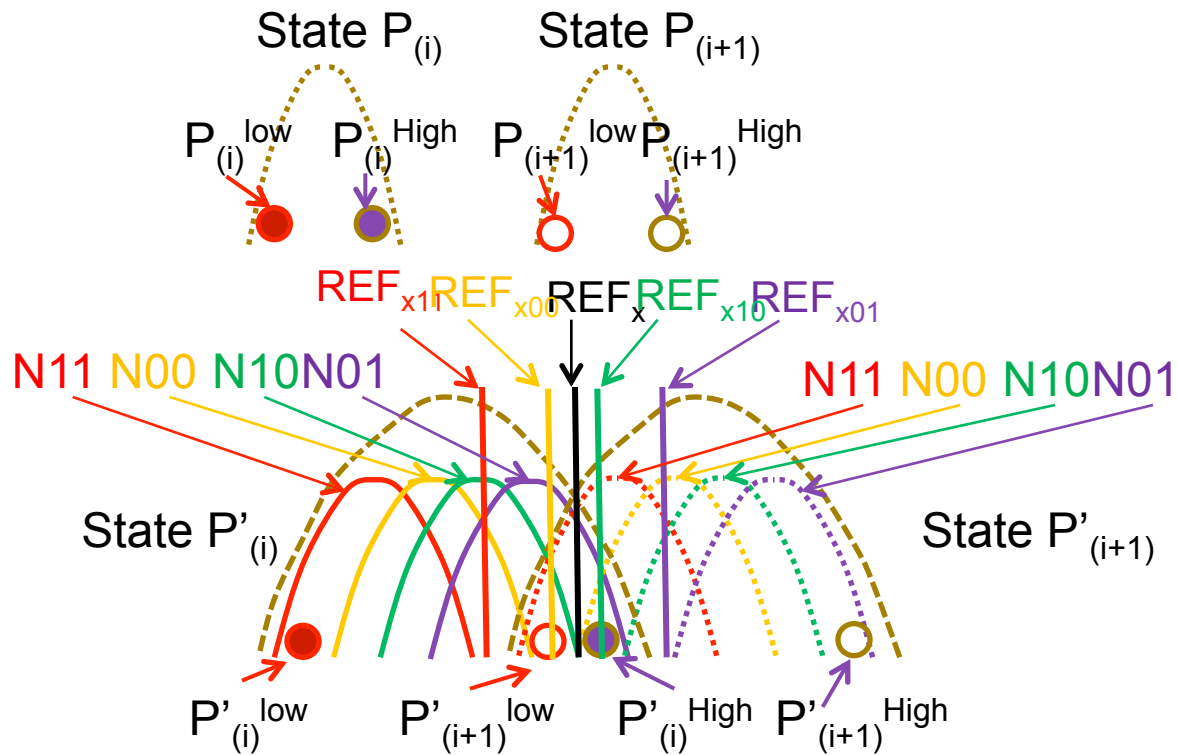 - In extended lifetime: less than 5% performance degradation

SAFARI

Carnegie Mellon

Overall vs Conditional Vth Distributions
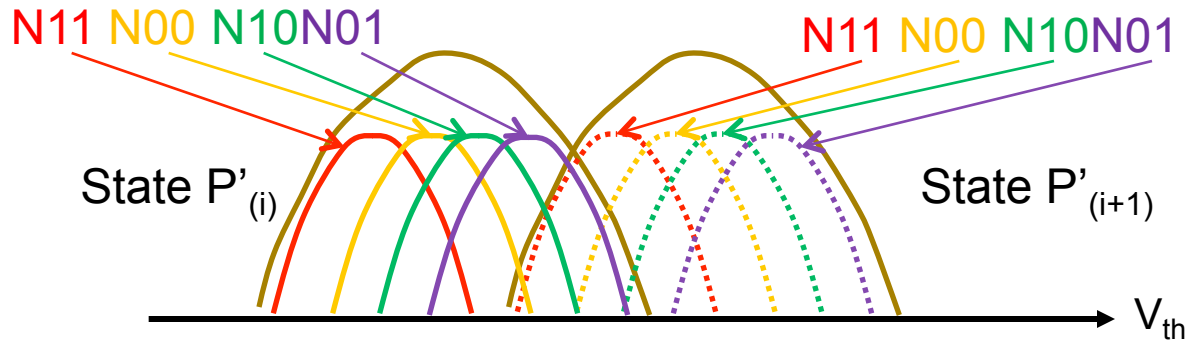
- Overall distribution: p(x)
- Conditional distribution: p(x, z=m)
  - m could be 11, 00, 10 and 01 for 2-bit MLC all-bit-line flash
- Overall distribution is the sum of all conditional distributions

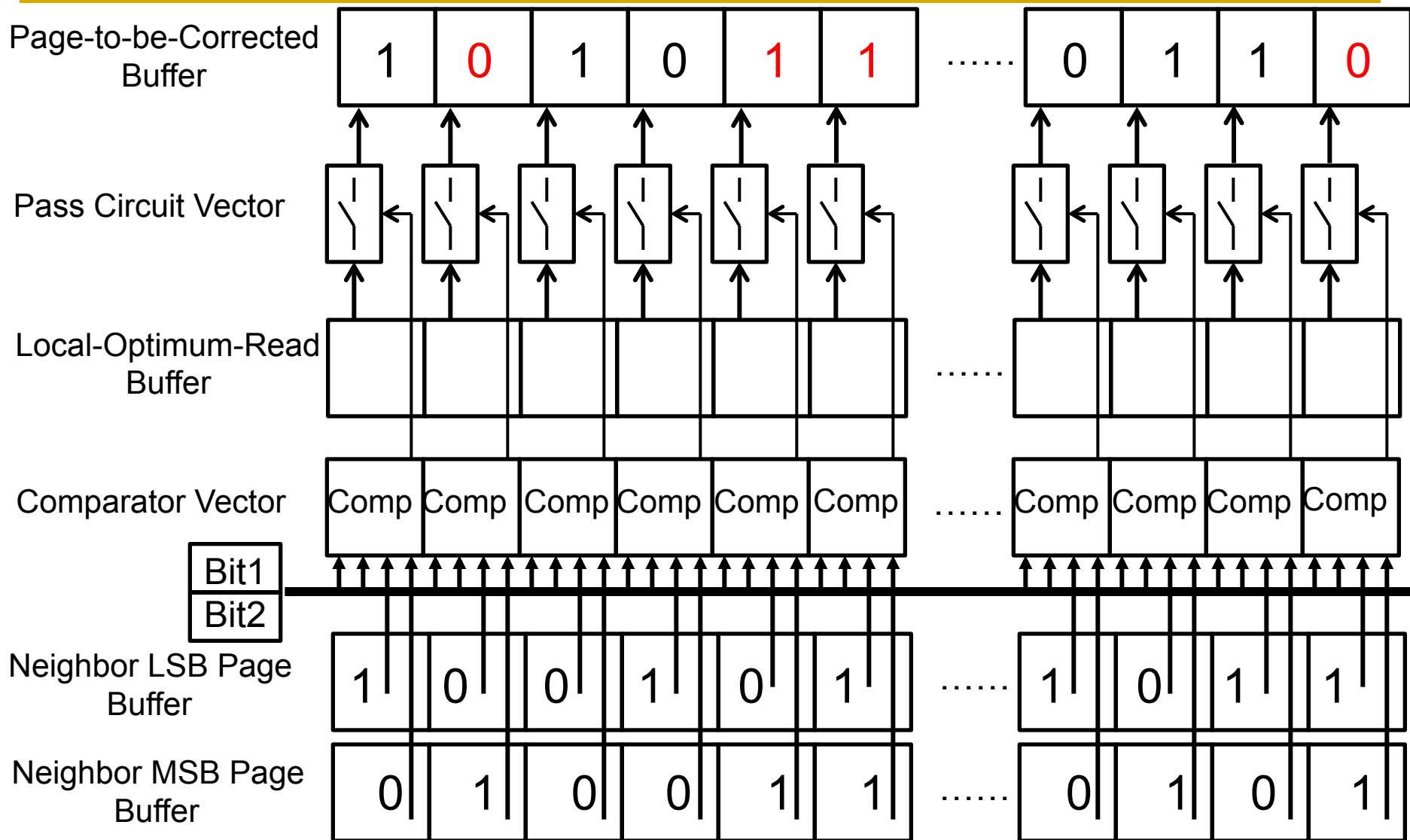$$p(x) = \sum_{m=1}^{2^n} p(x, z = m)$$

Prioritized NAC

- Dominant errors are caused by the overlap of lower state interfered by high neighbor interference and the higher state interfered by low neighbor interference

Procedure of NAC

- ## Online learning
  - Periodically (e.g., every 100 P/E cycles) measure and learn the overall and conditional threshold voltage distribution statistics (e.g. mean, standard deviation and corresponding optimum read reference voltage)

- ## NAC procedure
  - Step 1: Once ECC fails reading with overall distribution, load the failed data and corresponding neighbor LSB/MSB data into NAC
  - Step 2: Read the failed page with the local optimum read reference voltage for cells with neighbor programmed as 11
  - Step 3: Fix the value for cells with neighbor 11 in step 1
  - Step 4: Send fixed data for ECC correction. If succeed, exit. Otherwise, go to step 2 and try to read with the local optimum read reference voltage 10, 01 and 00 respectively

Microarchitecture of NAC (Initialization)

| Page-to-be-Corrected Buffer | 1 | 0 | 1 | 0 | 1 | 1 | ...... | 0 | 1 | 1 | 0 |

Pass Circuit Vector

Local-Optimum-Read Buffer

| Comparator Vector | Comp | Comp | Comp | Comp | Comp | Comp | ...... | Comp | Comp | Comp | Comp |

Bit1
Bit2

| Neighbor LSB Page Buffer | 1 | 0 | 0 | 1 | 0 | 1 | ...... | 1 | 0 | 1 | 1 |

| Neighbor MSB Page Buffer | 0 | 1 | 0 | 0 | 1 | 1 | ...... | 0 | 1 | 0 | 1 |

SAFARI

NAC (Fixing cells with neighbor 11)