University of Pittsburgh

DPA: A data pattern aware error prevention technique for NAND flash lifetime extension

*Jie Guo, *Zhijie Chen, **Danghui Wang, ***Zili Shao, *Yiran Chen

*University of Pittsburgh

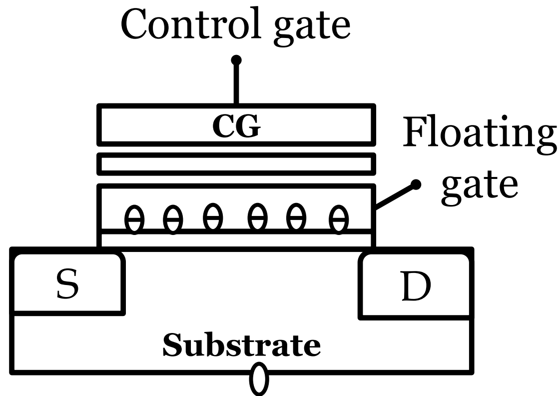
**Northwestern Polytechnical University

*** The Hong Kong Polytechnic University

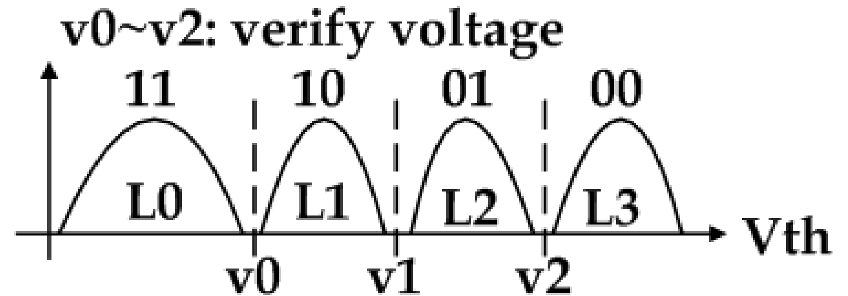
Outline

- MLC NAND flash basics
- MLC NAND flash bit error pattern
- Motivation
- Data Pattern Aware (DPA) overview
- DPA-PPU: pattern probability unbalance
- DPA-DRM: data-redundancy management
- DPA experimental results
- Conclusion

MLC NAND flash basics



NAND flash cell



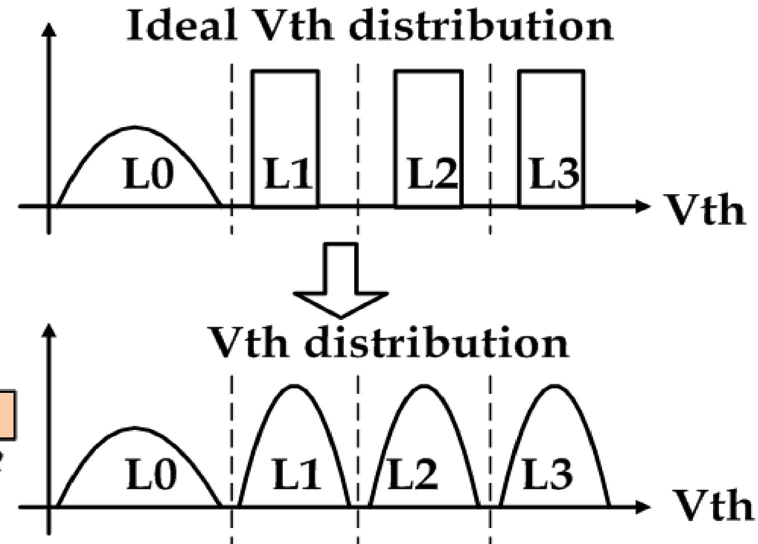
MLC NAND flash Vth

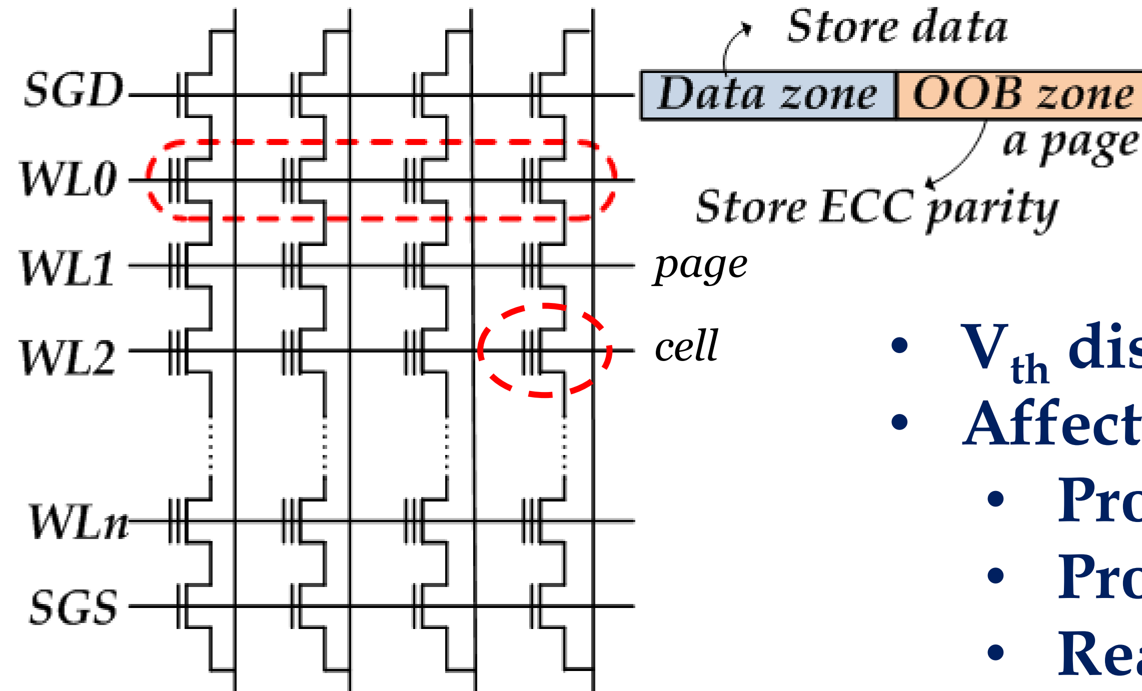
- A NAND flash cell: floating gate transistor
- Cell V_{th} is configured by electrons on floating gate
- Bits are represented by V_{th} levels
- MLC NAND flash: L0~L3 for 11, 10,01 and 00
- Program: inject electrons to floating gate
- Erase: remove electrons from floating gate
- Read: compare V_{th} with ref. voltage

MLC NAND flash basics (cont'd)

NAND flash chip

- Each chip: 8192 blocks
- Each block: 64/128 pages
- Each page: OOB + data

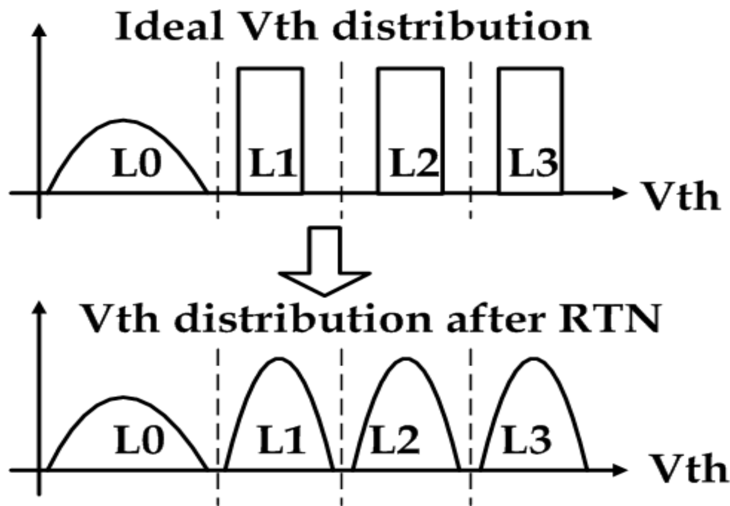


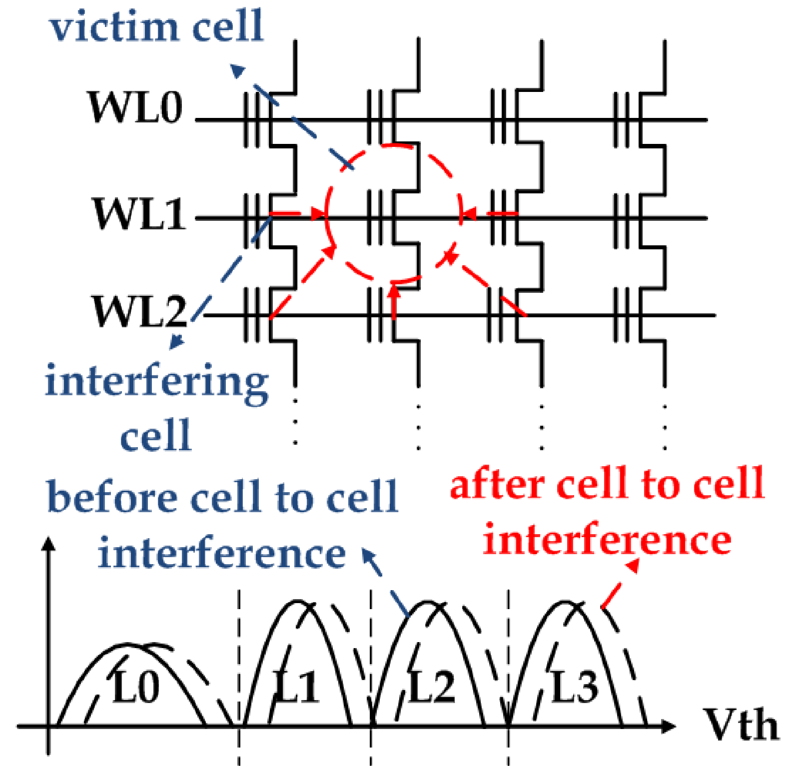


- V_{th} distortion leads to bit error
- Affecting factors
  - Program/Erase(P/E) cycling
  - Program disturb noise
  - Read disturb noise
  - Retention time noise

NAND flash block structure

MLC NAND flash error pattern

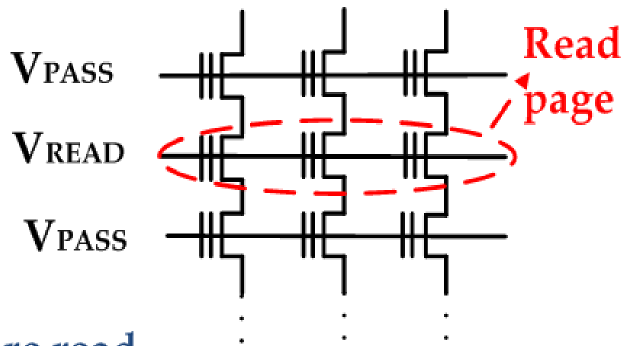


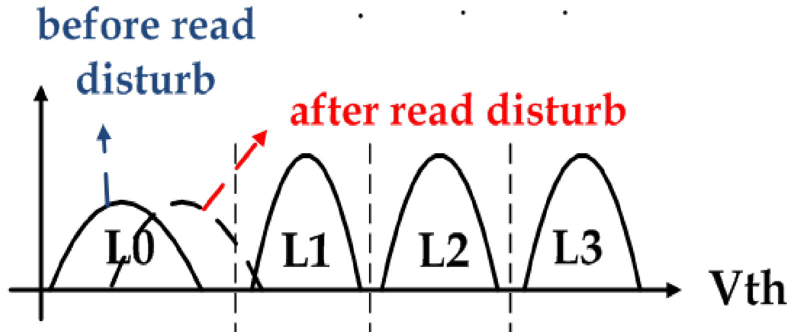


Program disturb:

- Random telegraph noise(RTN)
- Cell-to-cell interference
- Most severe program disturb
  - Program to L1/L3[1]

MLC NAND error pattern (cont'd)



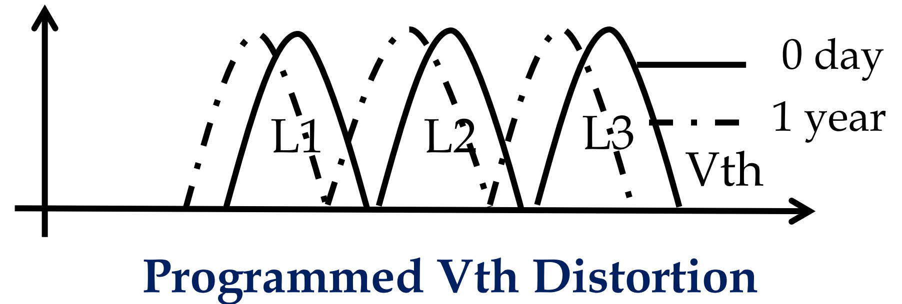


Read disturb

- Results from Fowler-Nordheim tunneling effect
- L0 is most vulnerable to read disturb[2]

Retention time noise

- Caused by charge loss on floating gate
- L3 is most vulnerable to retention time error[2]



Programmed Vth Distortion

Motivation

Due to BER, n-bit BCH ECC(m,l,n) is employ to protect data integrity. Assume that p_c is bit error rate (BER) of a single cell. Uncorrectable BER (UBER) is

$$UBER = 1 - \sum_{k=0}^{n} \binom{m}{k} {p_c}^k (1 - p_c)^{m-k}$$

Assume p_i is BER per Vth level and p_{li} denotes the probability of each Vth level. MLC NAND flash p_c is

$$p_c = \sum_{i=0}^{3} p_{li} \times p_i$$

Assume that P_{li} of each Vth level is equal.

Motivation (cont'd)

Based on the reliability model in [3][4][5], we estimate UBER

Retention time ECC failure rate
Read disturb ECC failure rate
Program Disturb ECC failure rate

1.E-01
1.E-03
1.E-05
1.E-07
1.E-09
1.E-11
1.E-13
1.E-15
1.E-17
1.E-19

1.E-01
1.E-04
1.E-07
1.E-10
1.E-13
1.E-16
1.E-19
1.E-22
1.E-25

5K
10K
15K
20K
25K

P/E cycle count

System lifetime: only 7.5K under 10^{-13} UBER

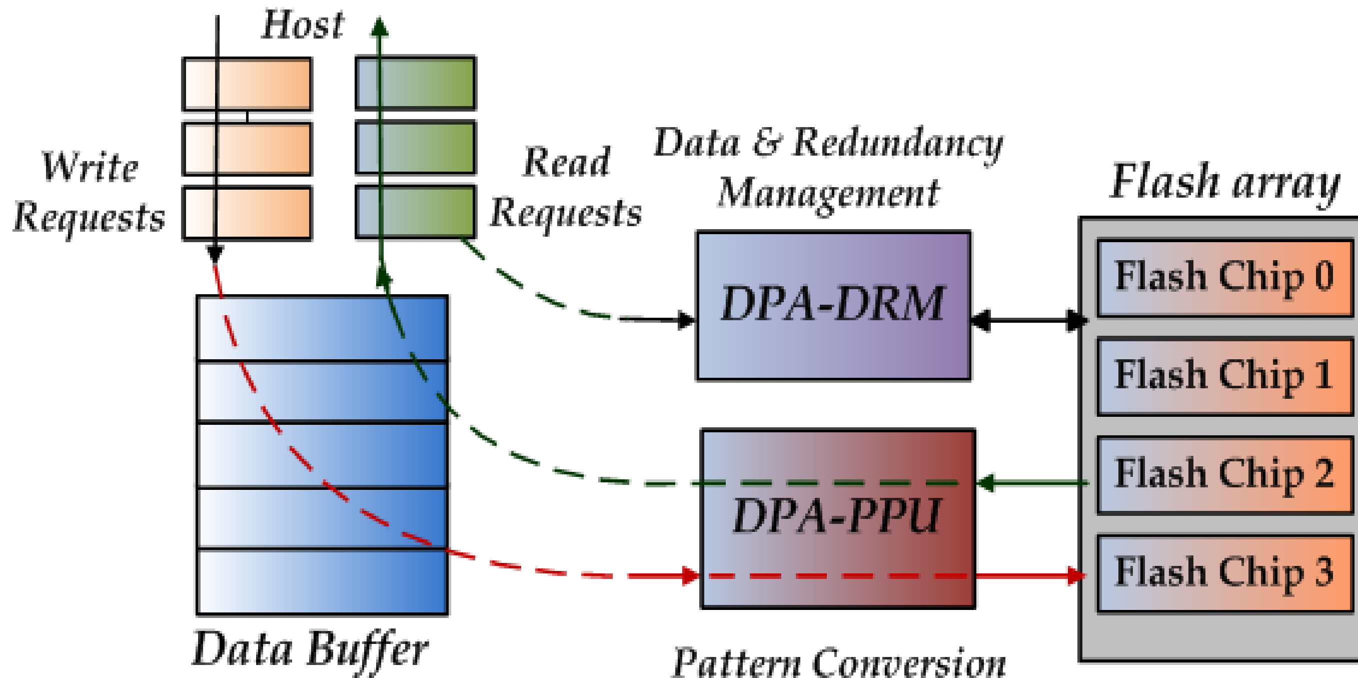
Motivation: if we can reduce p_i of the Vth level which is most vulnerable to noise, we can extend system lifetime

Data Pattern Aware (DPA) overview

- **Data Pattern Aware DPA technique**
  - Aim: reduce NAND flash BER
  - Observation: highest Program error BER occurs when the interfering cell is programmed to L1(10) or L3 (00); highest Retention time BER occurs in L3 (00)
- **DPA-PPU: pattern probability unbalance**
  - Increase the probability of programming NAND flash cells to L0 by maximizing the ratio of 1's in the stored data
- **DPA-DRM: Data-Redundancy Management**
  - DPA-PPU induces redundancy
  - DPA-DRM protects redundancy integrity and reduces redundancy induced performance degradation

DPA overview (cont'd)

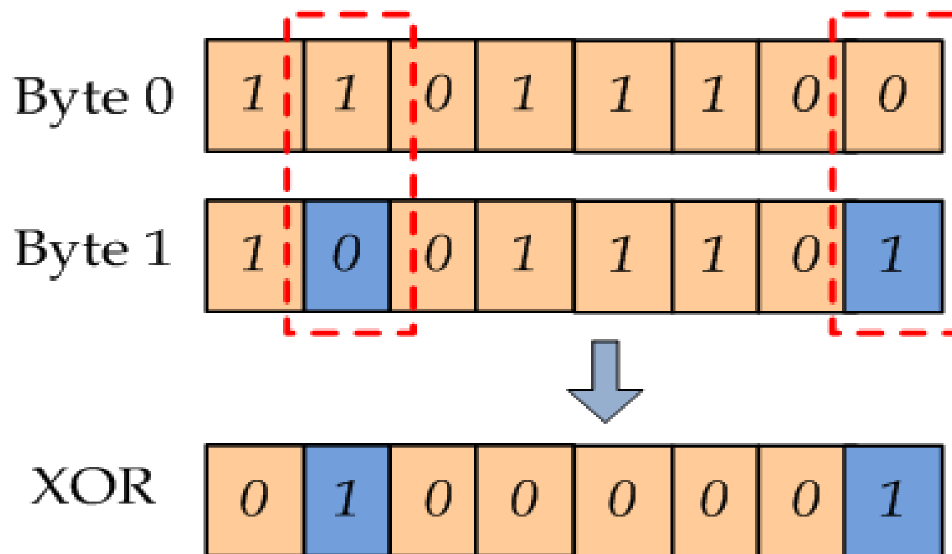
- Write: DPA-PPU converts data pattern to a favorable one
- Read: DPA-PPU recovers the inverted data pattern to the original one before sent to host
- DPA-DRM performs data management to reduce DPA-PPU incurred performance overhead



Data pattern observation

DPA-PPU is based on investigation of data patterns

- Strongly corrected data
  - Small bit difference
  - Simple XOR operation can reduce 1's number



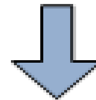
Data pattern observation (cont'd)

- **Weakly correlated data**
  - **Modulo-2 division can reduce 1′s number**
  - **Need to select the optimal polynomial to reduce number of 1′s**

010010100010010101001 01

Modulo-2 divided by polynomials

100101

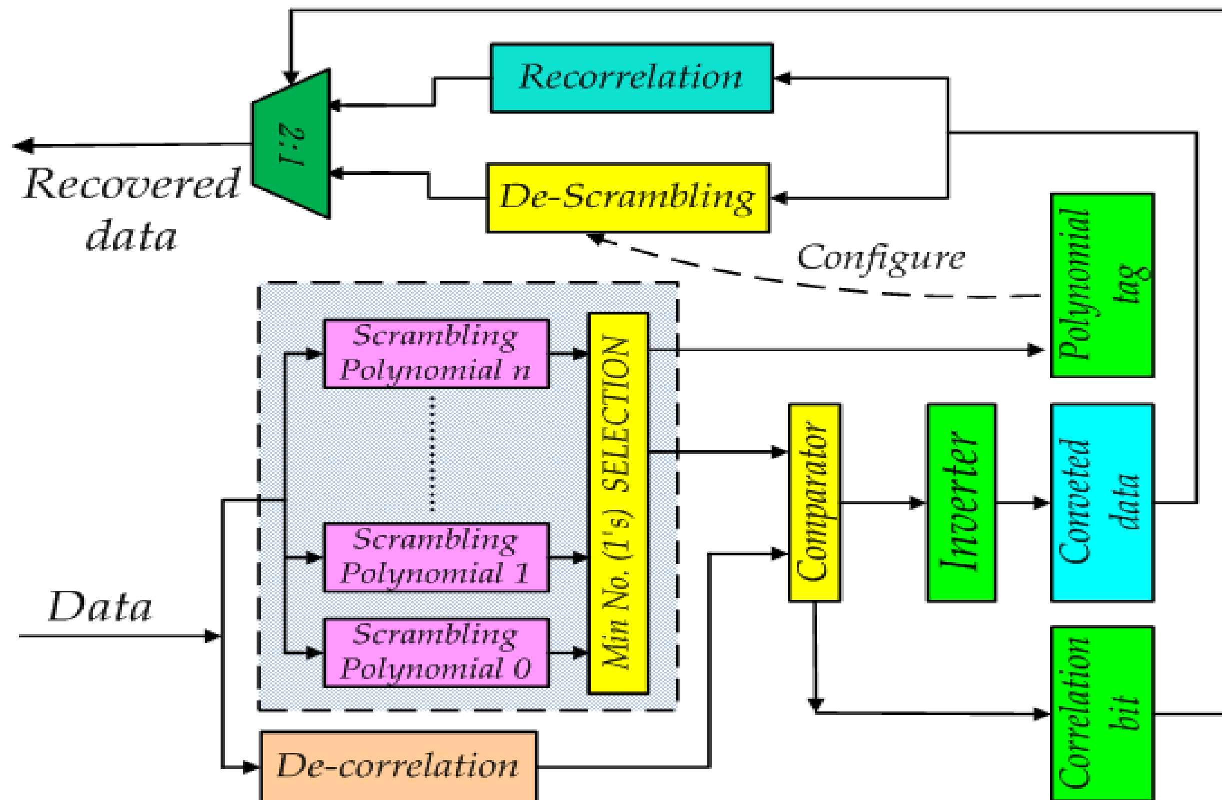


01000000001000000100000

DPA-PPU: pattern probability unbalance (cont’d)

- Step 1: decrease 0’s number (it is easier to increase the number of 0’s than 1’s)
  - Strongly correlated data
    - De-correlation (XOR) [6]
  - Weakly correlated data
    - Scrambling coding (modulo-2 division)
    - Use multiple polynomials
- Step 2: invert all bits to obtain more 1’s
- Implementation
  - Data → multiple chunks
  - Perform scrambling/de-correlation to each chunk

DPA-PPU: pattern probability unbalance (cont'd)



DPA-PPU architecture: need to store polynomial tags in NAND flash, leading to extra redundancy access

DPA-PPU: pattern probability unbalance (cont'd)

- Estimation of polynomial induced redundancy overhead:
  - Redundancy size: 16GB (6%) for 256GB storage system under 8B chunk size and 64 16-order polynomials
  - Redundancy issue
    - Reliability: more random data pattern → less reliable than converted data
    - Performance: extra access to redundancy → performance degradation

DPA-DRM: data-redundancy management

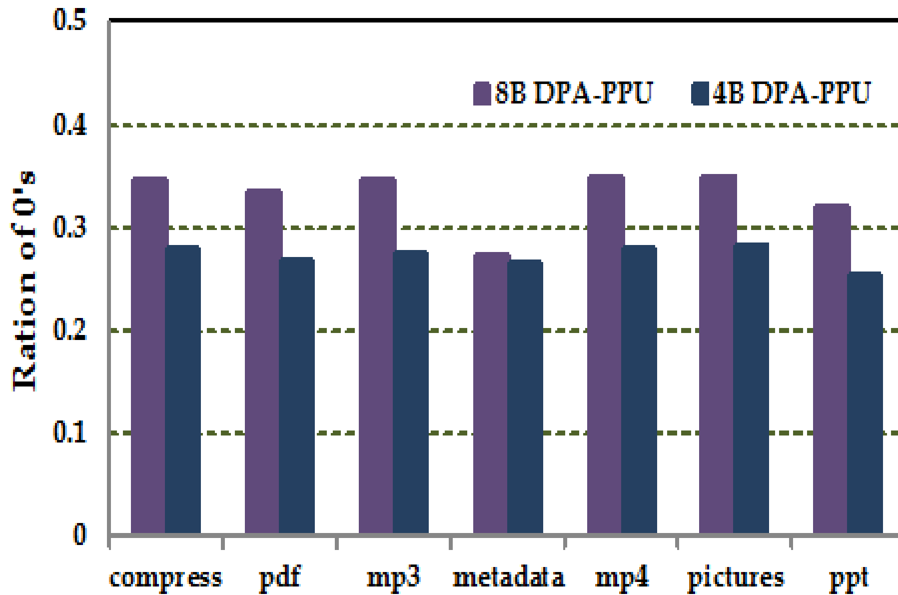
- Redundancy is vulnerable to bit error
  - Store redundancy and data in different blocks: redundancy and data blocks respectively
  - Apply BCH ECC+RAID5 to redundancy blocks
- Redundancy induces more access to NAND flash
  - Smaller data chunk → more redundancy access
  - Adopt adaptive DPA-PPU
    - Early P/E cycle stage, use large data chunk size
    - Later P/E cycle stage, use small data chunk size

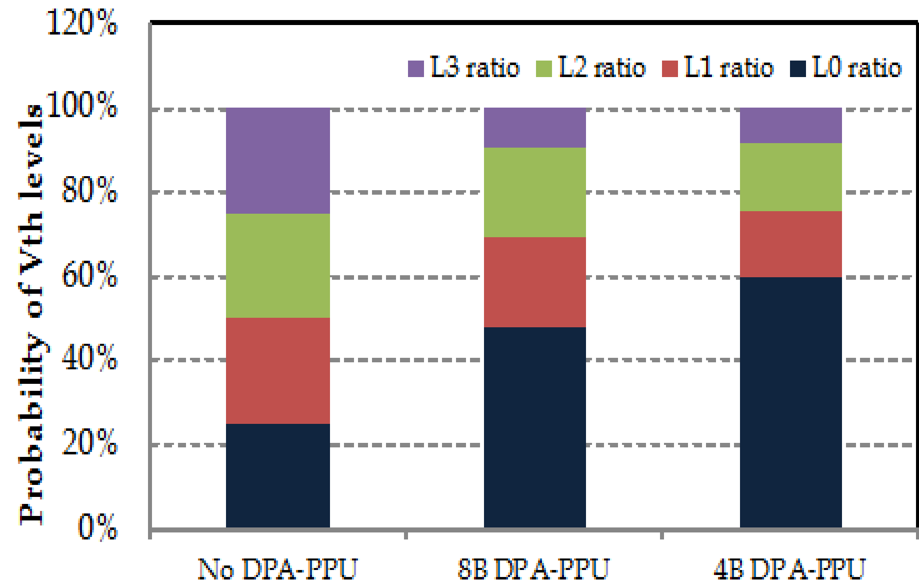
Experiment Setup

- Simulated system
  - Simulator: Flashsim [7]
  - Capacity: 256GB
- Reliability
  - 8-bit BCH-ECC(4312,4208,8)
  - Targeted UBER: 10^{-13}
- 5 Workload for simulation
- 7 data patterns

File type	File Number	File size
mp3	2	7.5MB, 6.3MB
mp4	1	101MB
compressed file (tar.gz)	2	5.3MB,4.2MB
pictures (.jpg)	2	1.42MB, 1.37MB
pdf	2	8.6MB,4.2MB
office (.ppt)	2	797KB, 1.2MB
system matadata	5	total 2MB

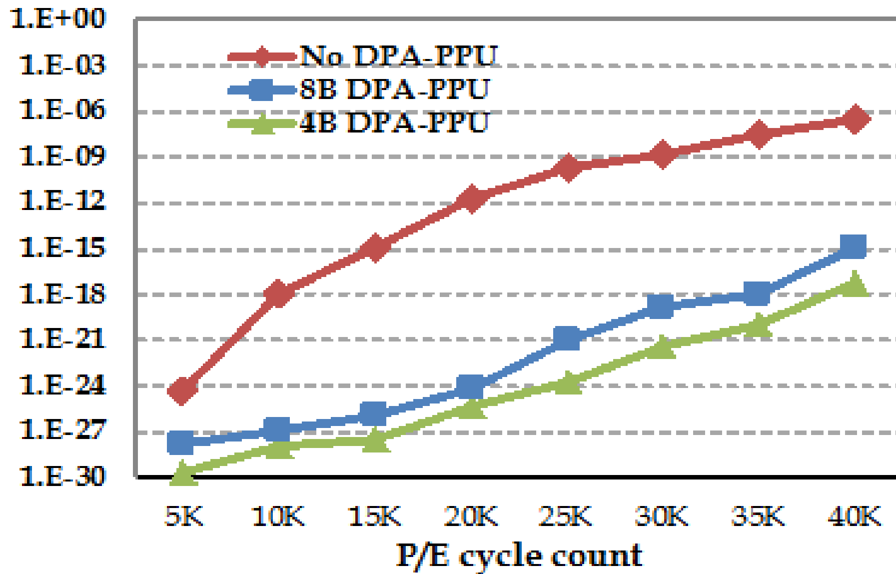
Experimental results

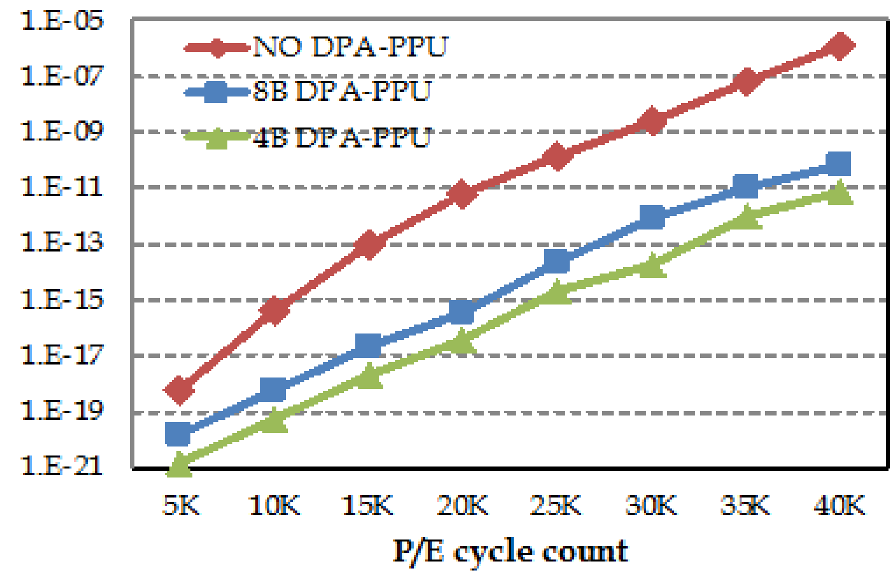


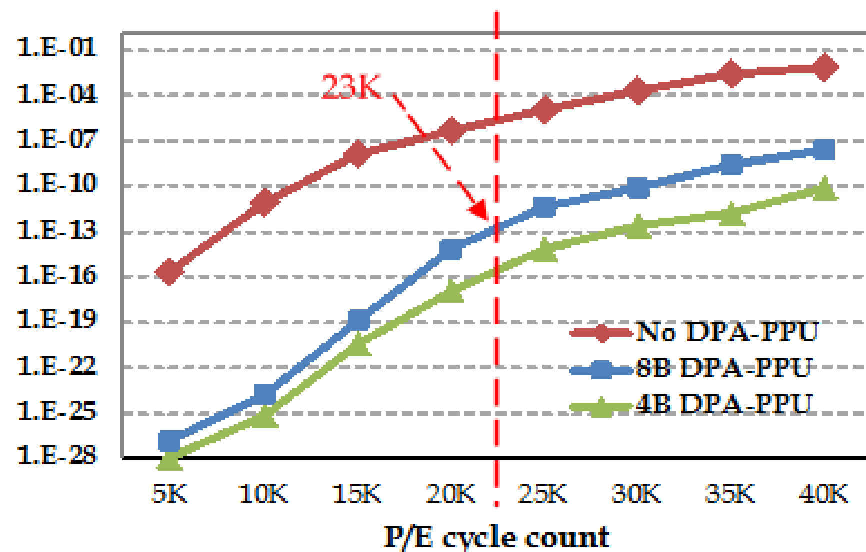


- DPA-PPU reduces 0's ratio to 27% and 33% with 4B and 8B data chuck size .
- L0 ratio increases to 48% and 59% under 8B and 4B chuck size.

Experimental results (cont’d)





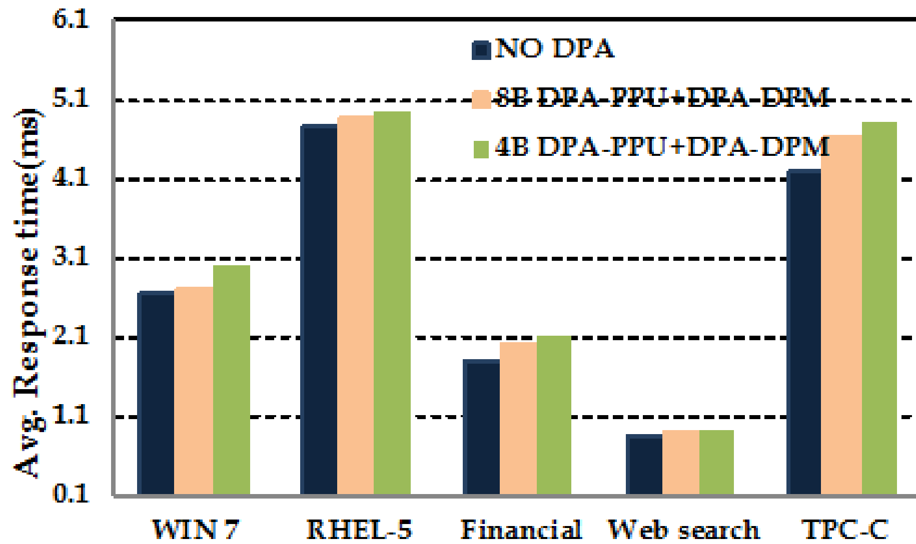


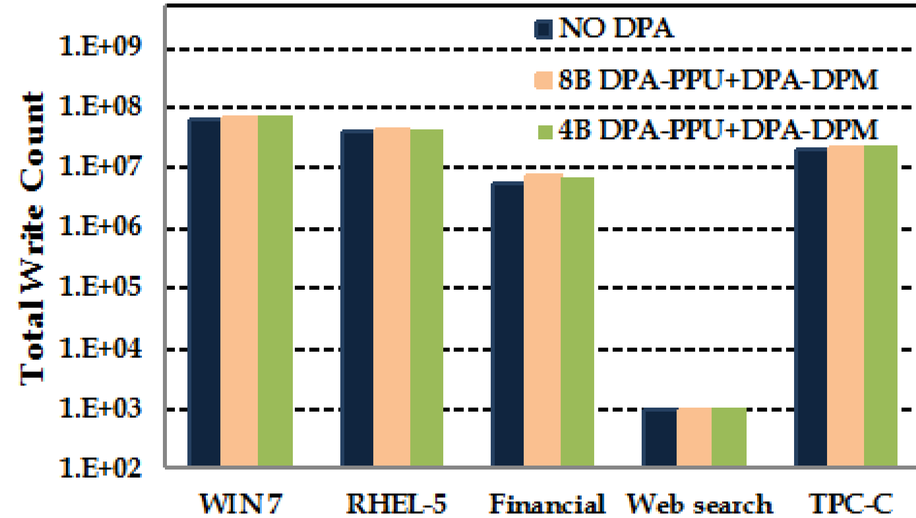
Program disturb UBER: up to 10^{-9} reduction on ave.

Read disturb UBER reduction: due to program disturb reduction

Retention time UBER: up to 10^{3} reduction on ave.

Experimental results(cont'd)





- **23K P/E cycle, ave. resp. time of DPA+DRM with 4B & 8B data chunk size degrades by 9% and 13% on ave.**
- **DPA+DRM with 4B and 8B data chuck increases write count & erase count by 8% and 4%**

DPA-Conclusion

- We quantitatively analyze the error pattern in MLC NAND flash memory. We observe that the Vth level L0 is most resilient to retention time error and cell-to-cell interference.
- We propose Pattern Probability Unbalance (DPA-PPU) scheme to skew the ratio of 1's and 0's in the stored data so as to place more cells on L0. Data Redundancy Management (DPA-DRM) is used to mitigate DPA-PPU induced performance overhead.
- Experimental results show that DPA prolongs NAND flash lifetime by 4 with 13% performance overhead.

Reference

1. J. Moon and et al, "Noise and interference characterization for mlc flash memories," in ICNC, 2012, pp. 588–592.
2. N. Mielke and et al, "Bit error rate in nand flash memories," in IRPS, 2008, pp. 9–19.
3. Y. Pan and et al, "Quasi-nonvolatile ssd: Trading flash memory nonvolatility to improve storage system performance for enterprise applications," in HPCA, 2012, pp. 1–10.
4. L. Cola and et al, "Read disturb on flash memories: Study on temperature annealing effect," Microelectronics Reliability, 2012.
5. M. Compagnoni and et al, "Analytical model for the electron-injection statistics during programming of nanoscale nand flash memories," T-ED, vol. 55, no. 11, pp. 3192–3199, 2008.
6. M. R. Stan and et al, "Low-power encodings for global communication in cmos vlsi," TVLSI, vol. 5, no. 4, pp. 444–455, 1997.
7. "A simulator for various ftl scheme," http://csl.cse.psu.edu/?q=node/322.

Thanks!