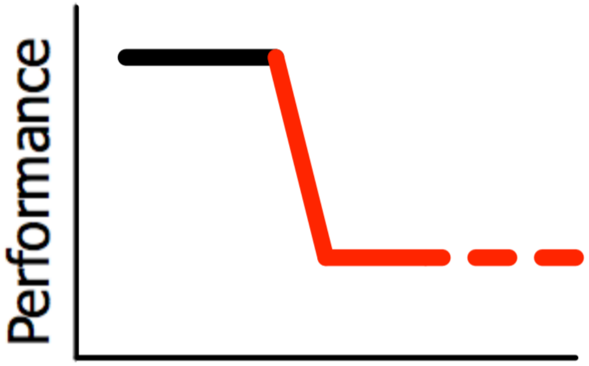
Design Tradeoffs for SSD Reliability

Bryan S. Kim, Jongmoo Choi, Sang Lyul Min

Seoul National University, Dankook University

High-level objectives

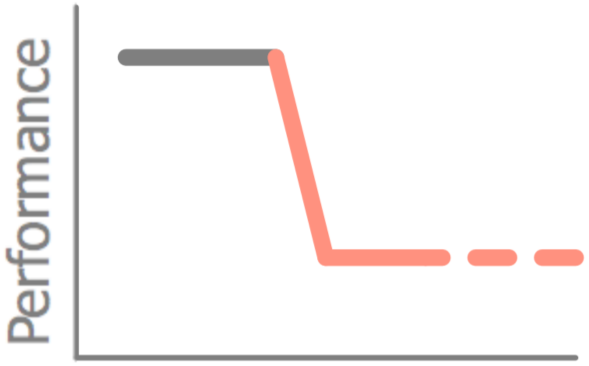
Understand the SSD-internal mechanisms
behind fail-slow symptoms



- H. Gunawi et al, "Fail-slow at scale: evidence of hardware performance faults in large production systems", FAST 2018

High-level objectives

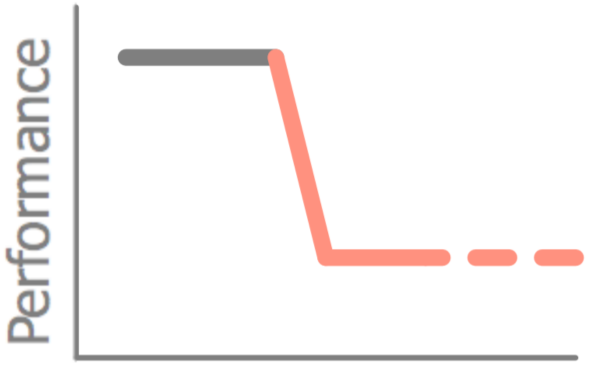
Examine SSD-internal reliability enhancement techniques



- Images from Google searches

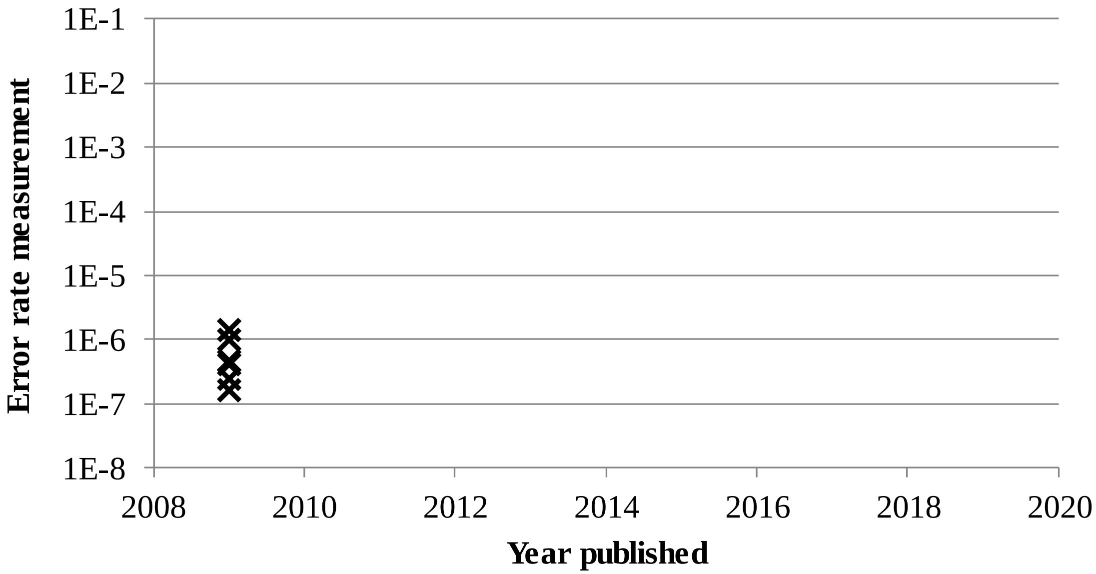
High-level objectives

Think about system- and device-level approaches for handling errors



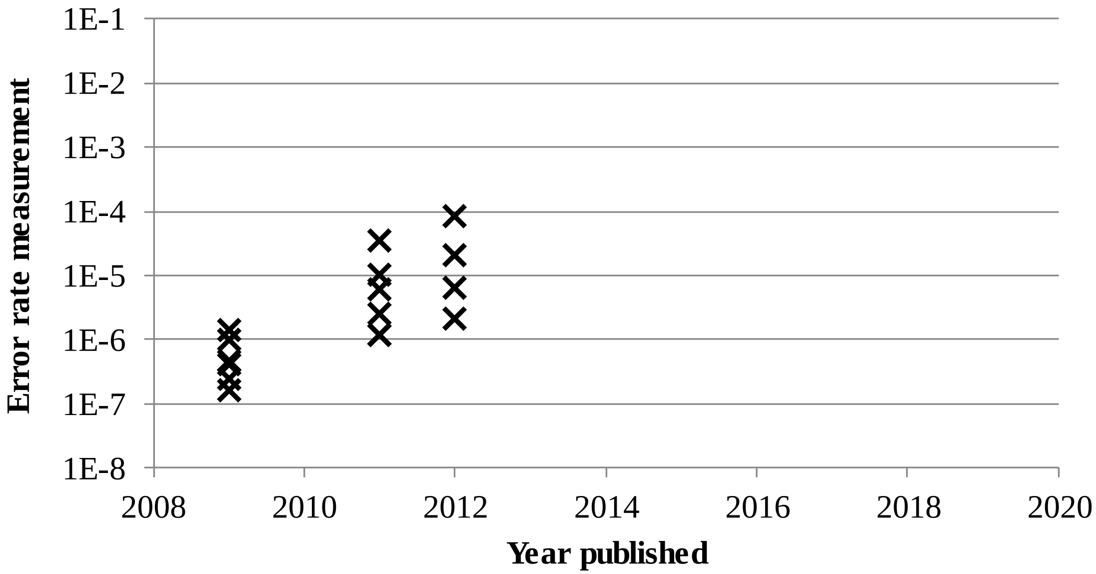
- Images from Google searches

How bad is it?



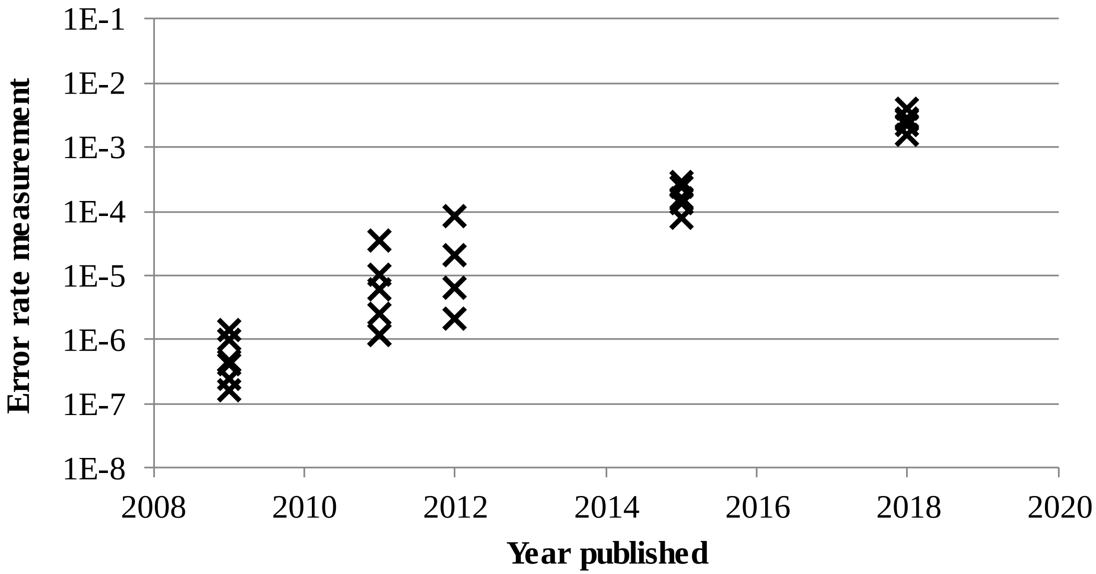
- L. Grupp et al, “Characterizing flash memory: anomalies, observations, and applications”, Micro 2009

How bad is it?



- H. Sun et al, “Quantifying reliability of solid-state storage from multiple aspects”, SNAPI 2011
- Y. Cai et al, “Error patterns in MLC NAND flash memory: measurement, characterization, and analysis”, DATE 2012

How bad is it?



- Y. Cai et al, “Data retention in MLC NAND flash memory: characterization, optimization, and recovery”, HPCA 2015
- Data from an industry partner, 2018

SSD's reliability issue

Error-prone memory

RBER: 10^{-4}~10^{-2}

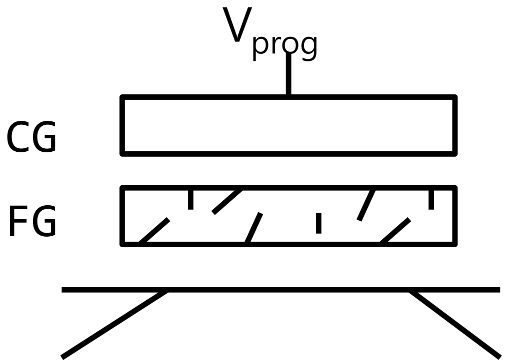
- How to make SSD reliable?
- Performance overhead?
- Across different chips and wear states?

Reliable SSD

UBER: <10^{-15}

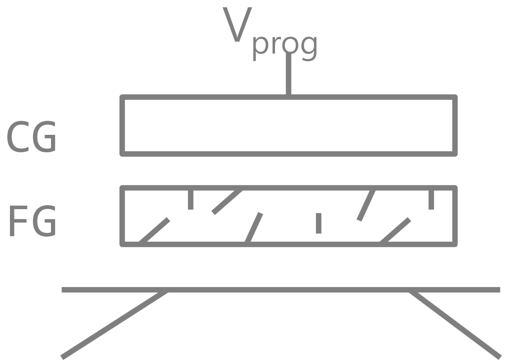
Flash memory errors

Wear-out

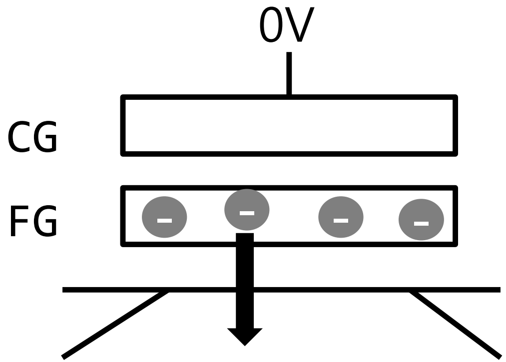


Flash memory errors

Wear-out

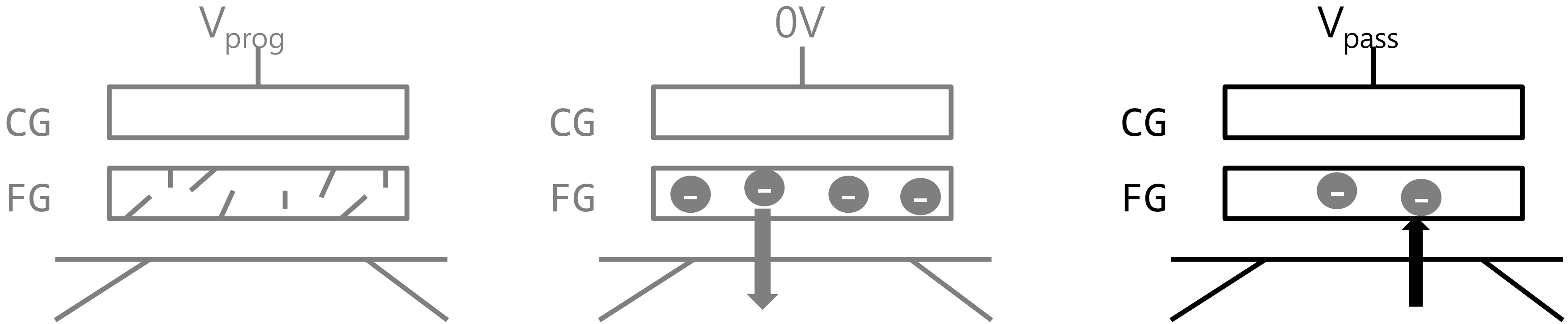


Retention loss



Flash memory errors

Wear-out Retention loss Disturbance



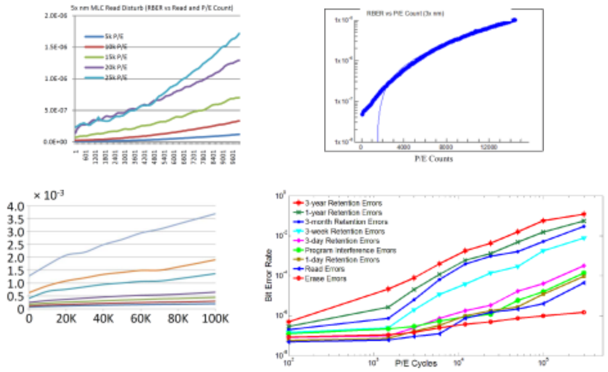
Flash memory error modeling

$$RBER\,(cycles,\ time,\ reads)$$
$$= \varepsilon$$
$$+\ \alpha \cdot cycles^{k}$$
$$+\ \beta \cdot cycles^{m} \cdot time^{n}$$
$$+\ \gamma \cdot cycles^{p} \cdot reads^{q}$$

- N. Mielke et al, "Reliability of solid-state drives based on NAND flash memory", Proceedings of the IEEE, 2017

From measurements to model

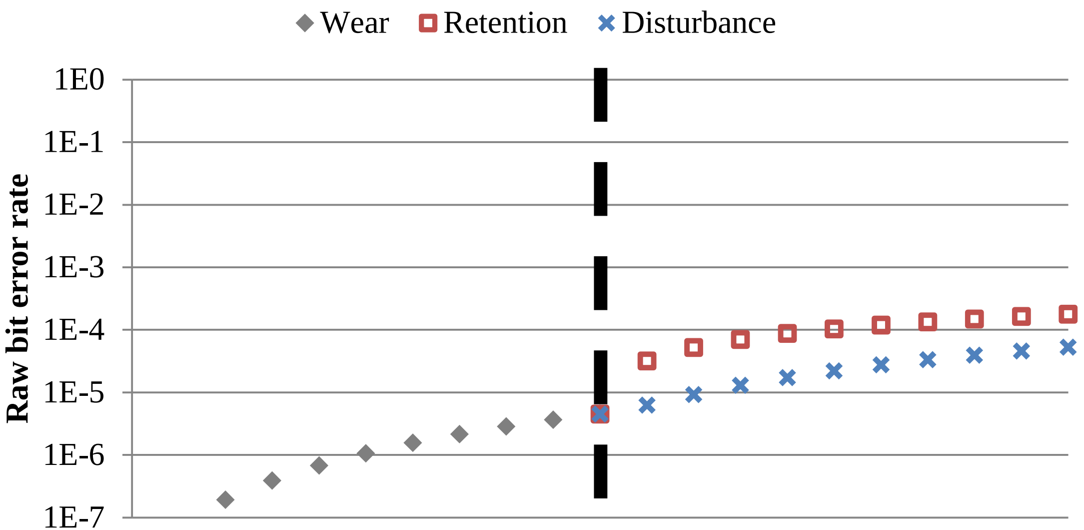
Measurement (data)	Model



- 3x-nm MLC (2011)
- 2y-nm MLC (2015)
- 3D TLC (2018)

- H. Sun et al, "Quantifying reliability of solid-state storage from multiple aspects", SNAPI 2011
- Y. Cai et al, "Data retention in MLC NAND flash memory: characterization, optimization, and recovery", HPCA 2015
- Y. Cai et al, "Read disturb errors in MLC NAND flash memory: characterization, mitigation, and recovery", DSN 2015
- Data from an industry partner, 2018

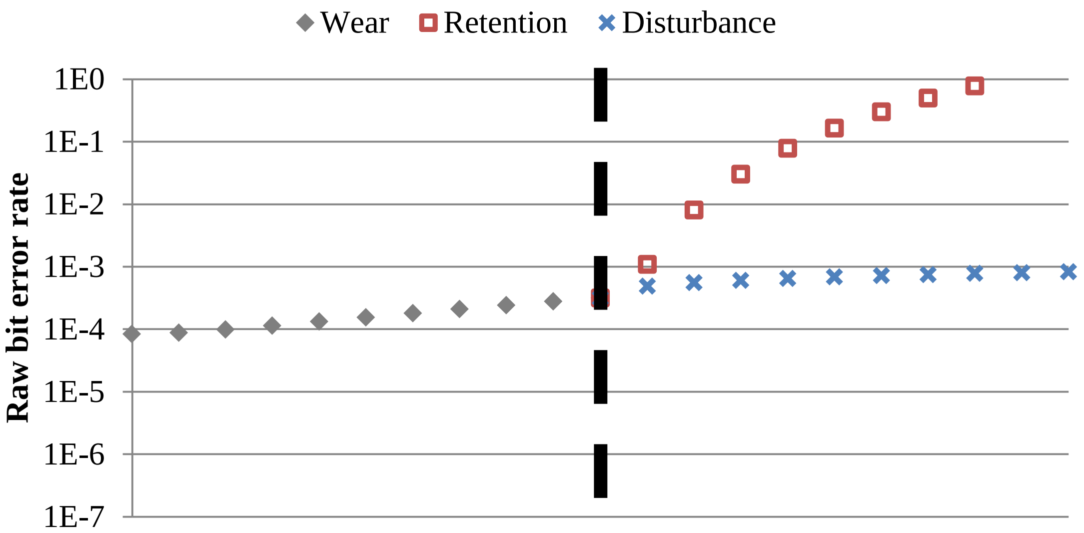
Error model: 3x-nm MLC (2011)



Wear up to
10K P/E cycles

10K P/E cycles +
up to 10K reads
or up to 1 year

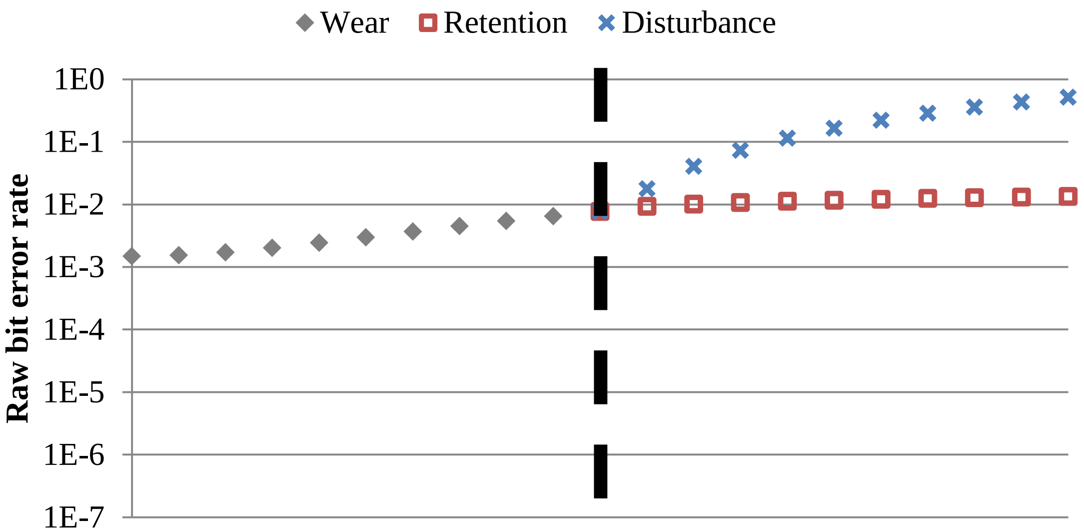
Error model: 2y-nm MLC (2015)



Wear up to
10K P/E cycles

10K P/E cycles +
up to 10K reads
or up to 1 year

Error model: 3D TLC (2018)



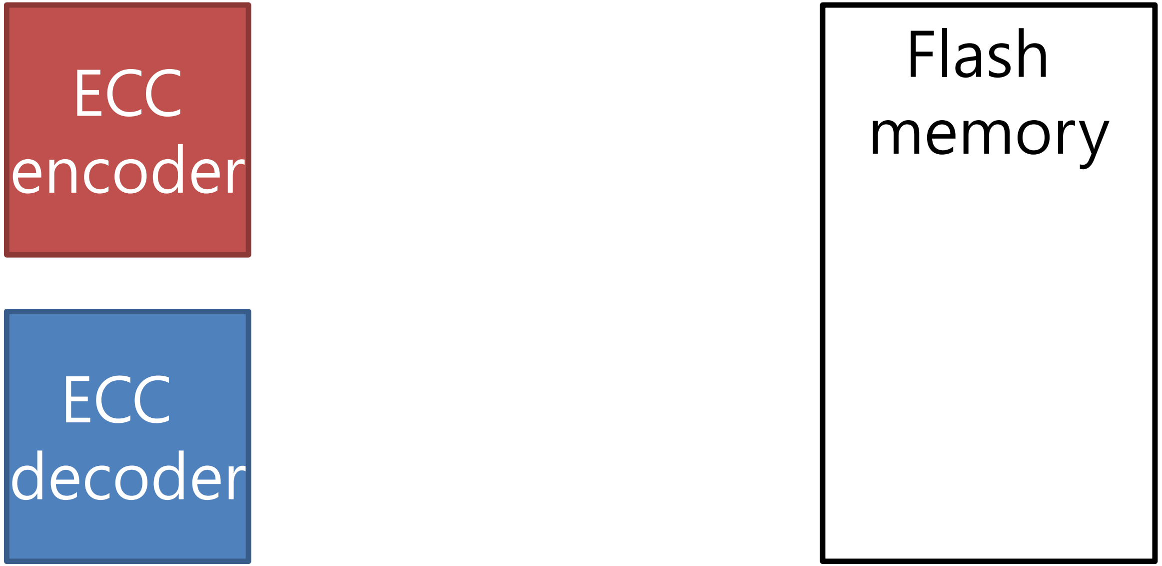
Wear up to
10K P/E cycles

10K P/E cycles +
up to 10K reads
or up to 1 year

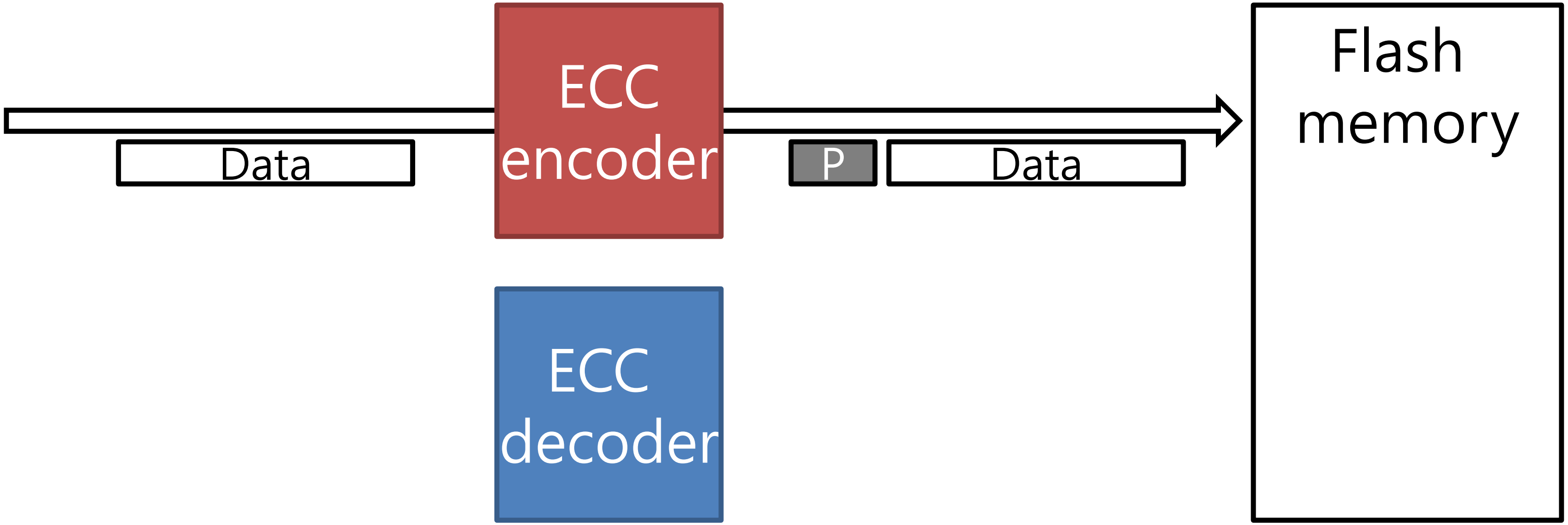
SSD reliability enhancements

- Error correction code
- Data re-reads
- Intra-SSD redundancy
- Background relocation

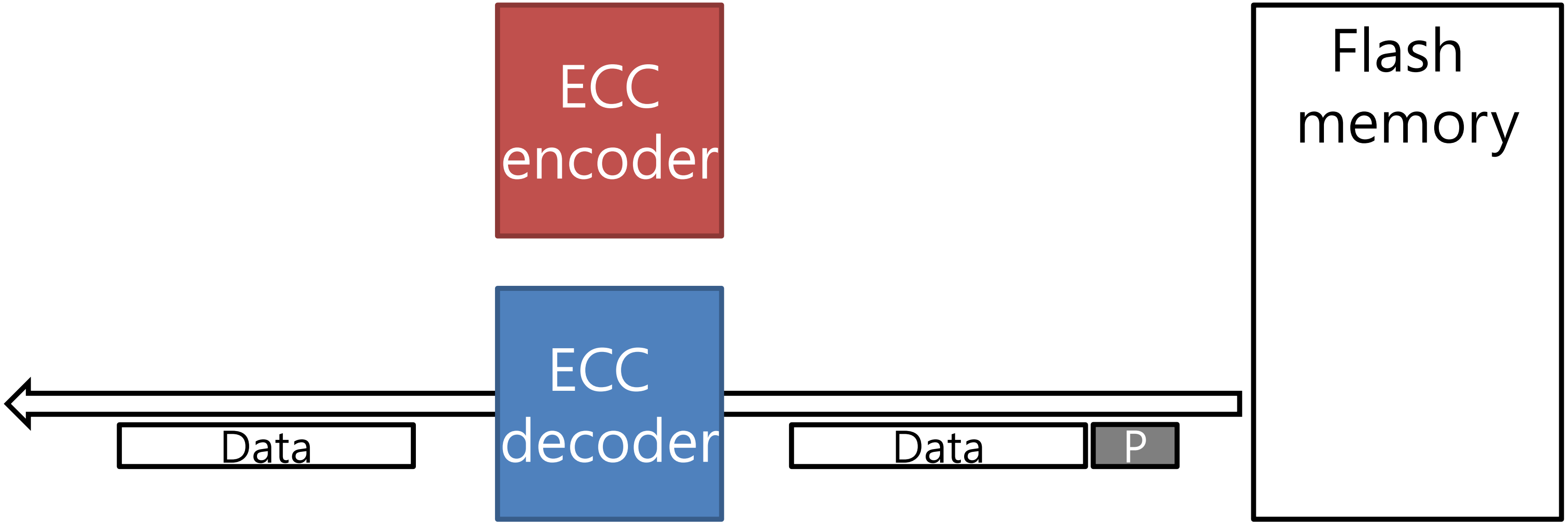
Error correction code



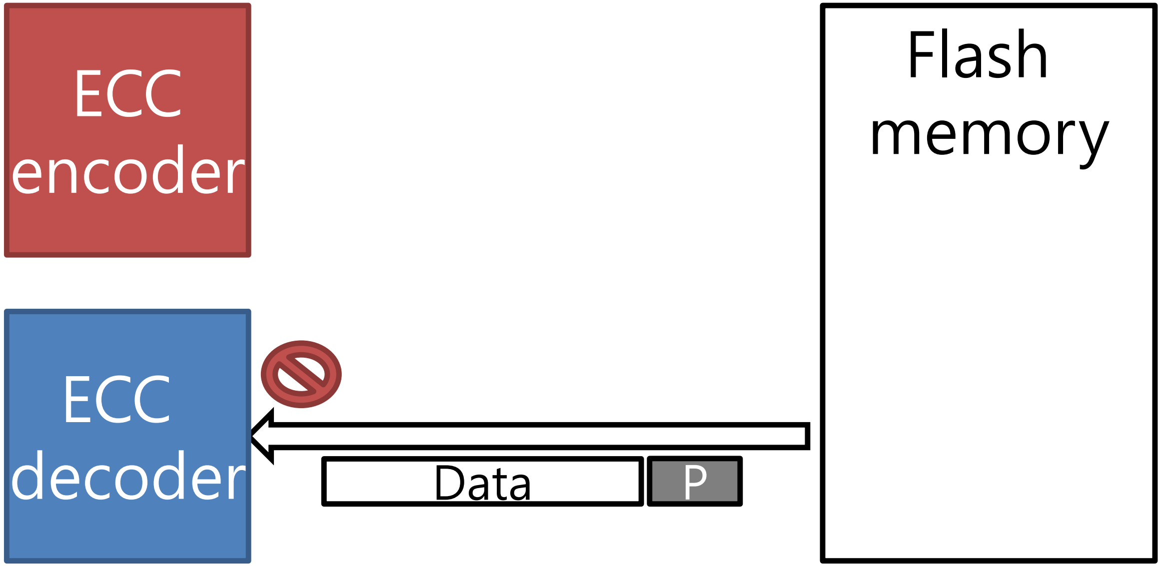
Error correction code



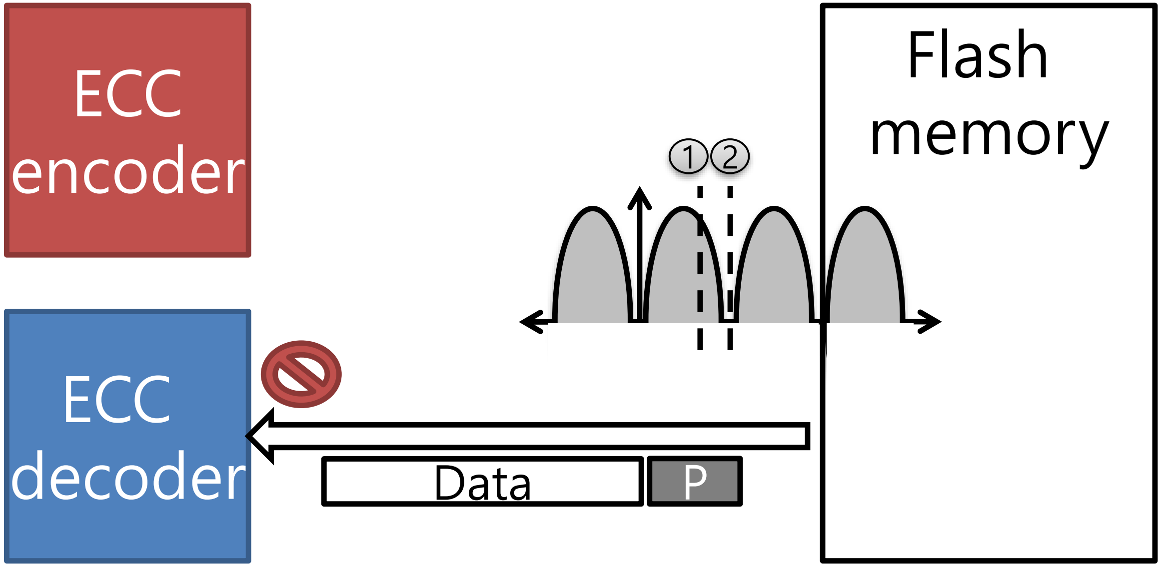
Error correction code



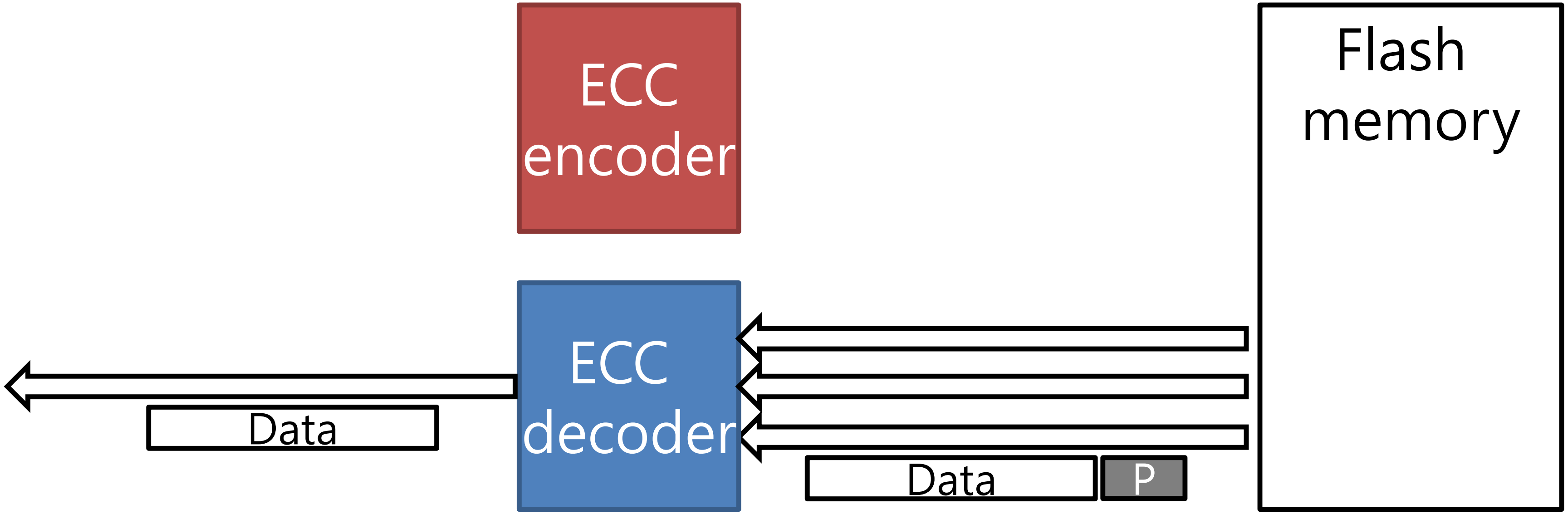
Data re-reads



Data re-reads



Data re-reads

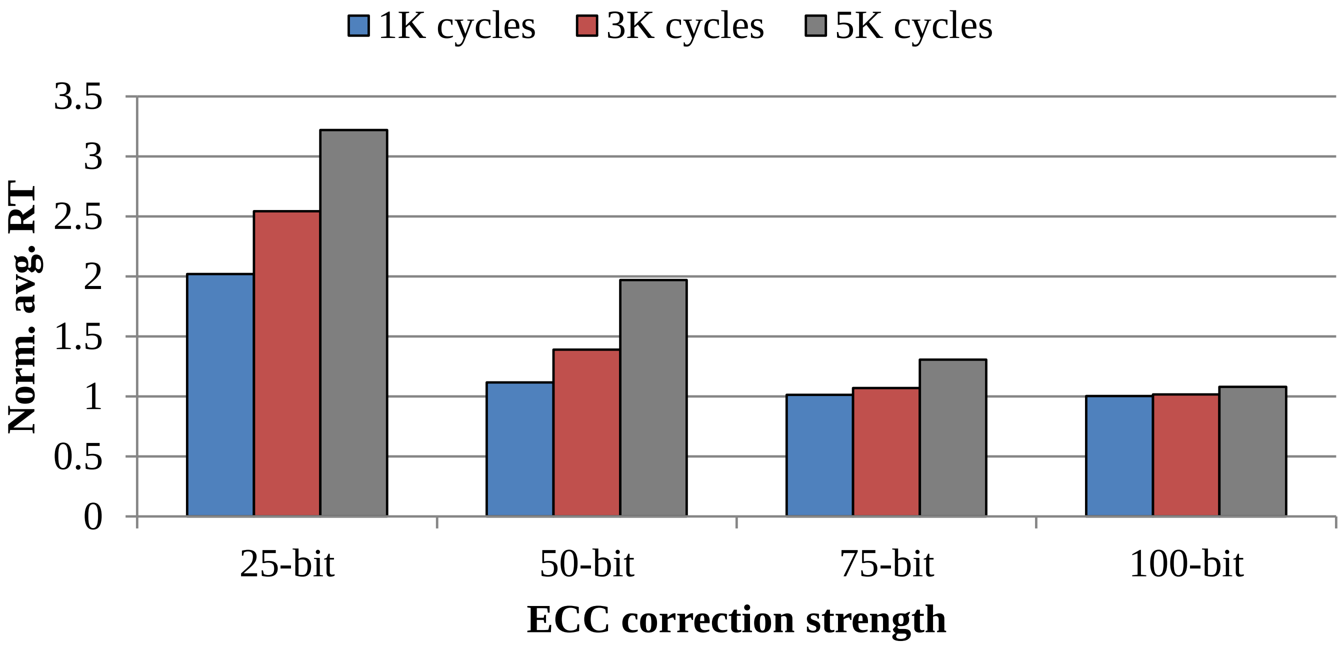


Summary: ECC and data re-reads

- Error correction code
  - Predictable performance
  - Is fixed at design-time
- Data re-read
  - Is much more powerful than ECC
  - Increases latency for correcting errors

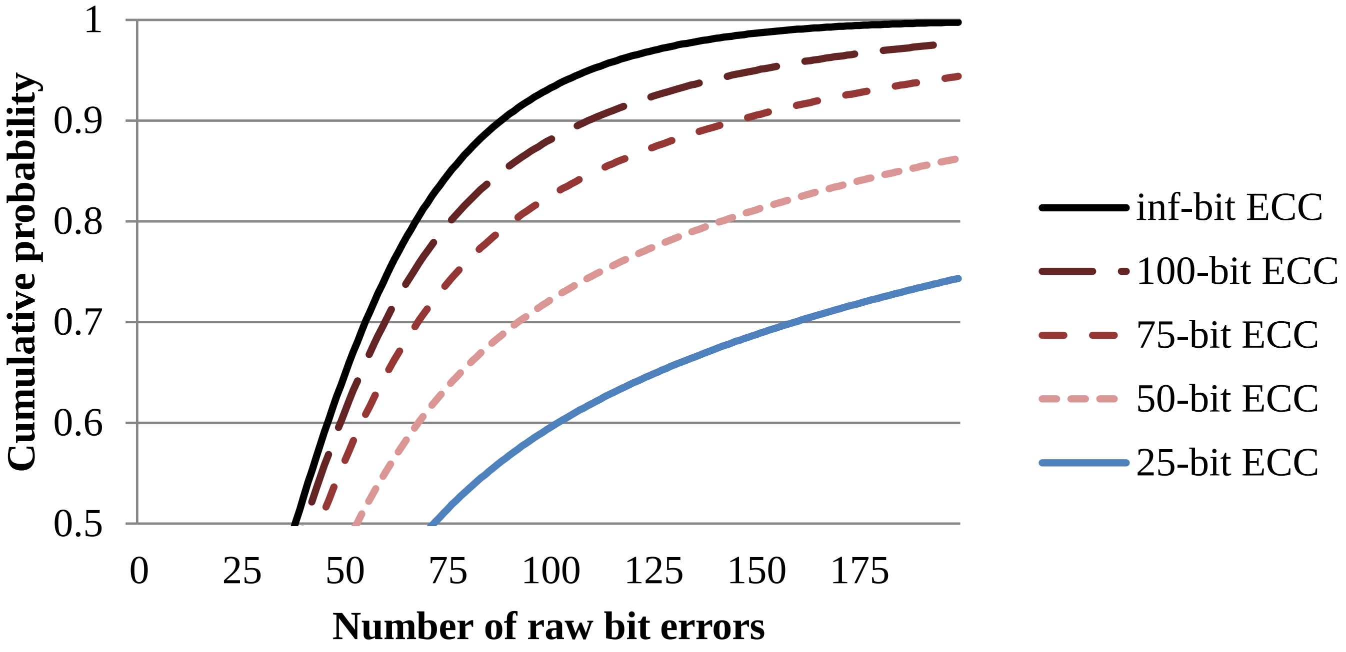
Evaluation: data re-read

For the 3D TLC (2018)



Why is data re-read bad?

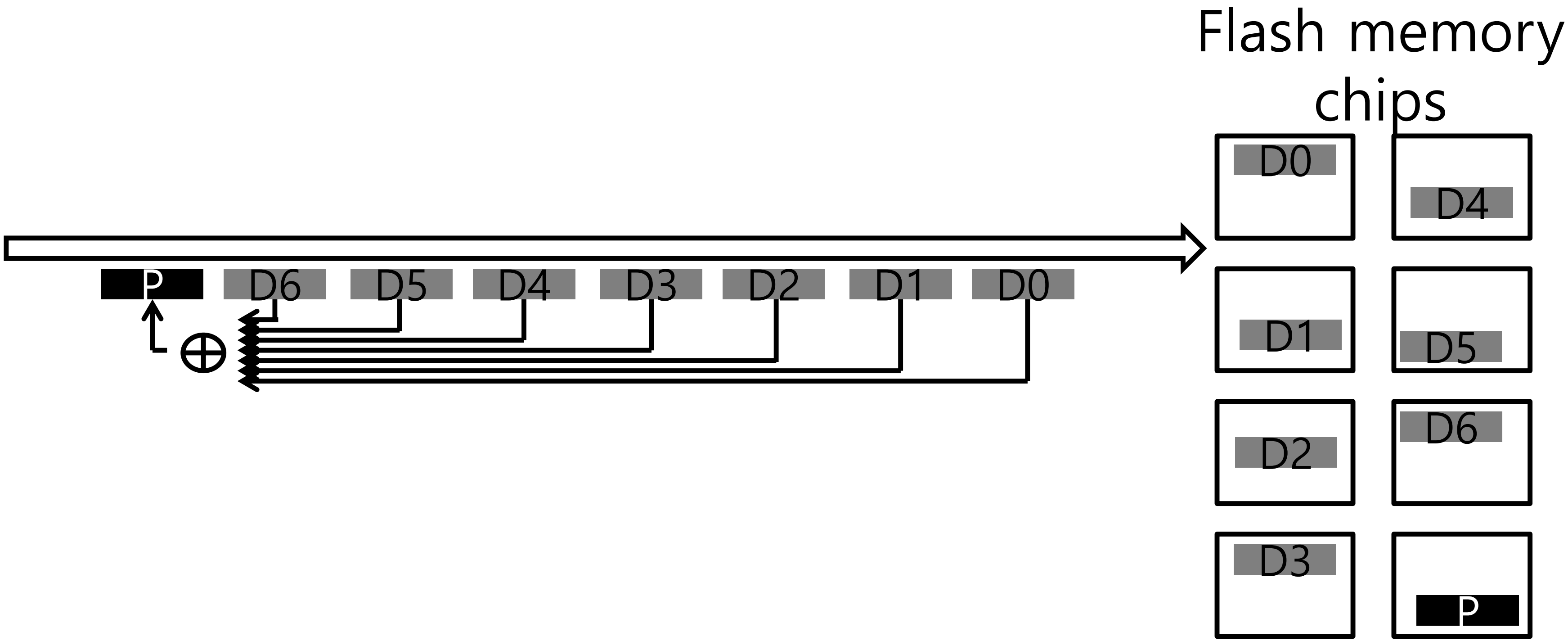
For the 3D TLC (2018)



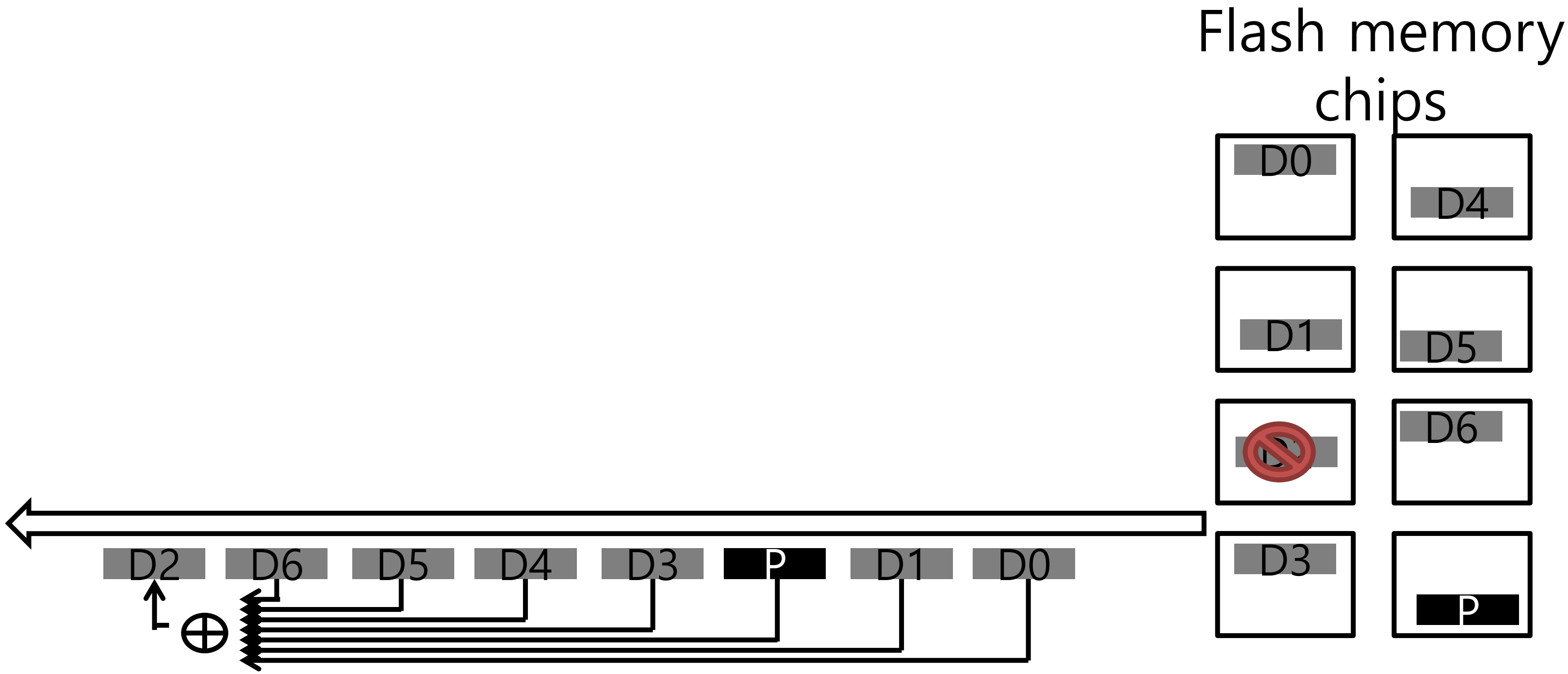
Observations

- Repeated data re-reads make it worse
  - 75-bit: ~30% increased latency at end-of-life

Intra-SSD redundancy



Intra-SSD redundancy

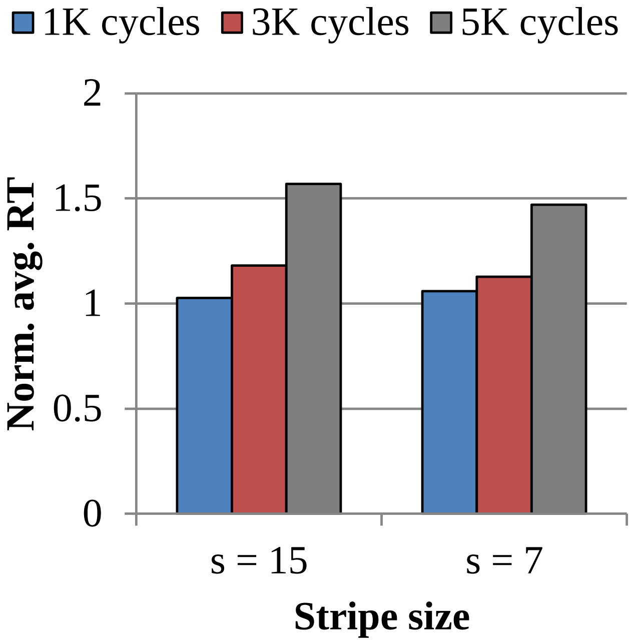


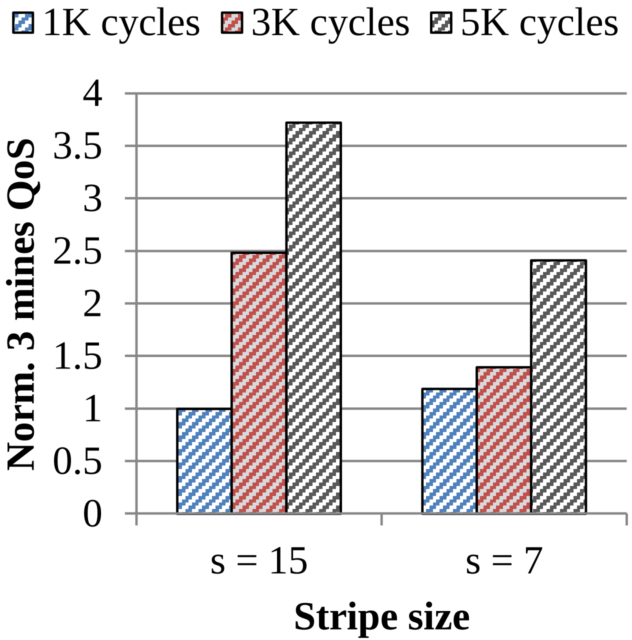
Summary: intra-SSD redundancy

- Error correction code
  - Is fixed at design-time
- Data re-read
  - Increases latency for correcting errors
- Intra-SSD redundancy
  - Protects against random and sporadic errors
  - Increases write amplification
  - Increases read amplification on errors

Evaluation: redundancy

For the 3D TLC (2018) with 75-bit ECC





Observations

- Repeated data re-reads make it worse
- Overheads of redundancy outweigh its benefits
  - +56% latency at end-of-life

Observations

- Repeated data re-reads make it worse
- Overheads of redundancy outweigh its benefits
- Scrubbing reduces error-induced latency, but increases internal traffic
  - +25% latency at end-of-life
  - Highly dependent on accuracy of error prediction

Observations

- Repeated data re-reads make it worse
- Overheads of redundancy outweigh its benefits
- Scrubbing reduces error-induced latency, but increases internal traffic
- We need to consider data characteristics and compositionally combine reliability enhancements

Holistic reliability management

- Cold data
  - Need protection against retention errors
  - Least write amplification with redundancy
  - Likely to be identified by GC

Holistic reliability management

- Cold data
  - Selective redundancy for GC-ed data
- Read-hot data
  - Need protection against disturbance errors
  - # of data re-reads can be used as proxy
  - Likely to be identified by scrubber

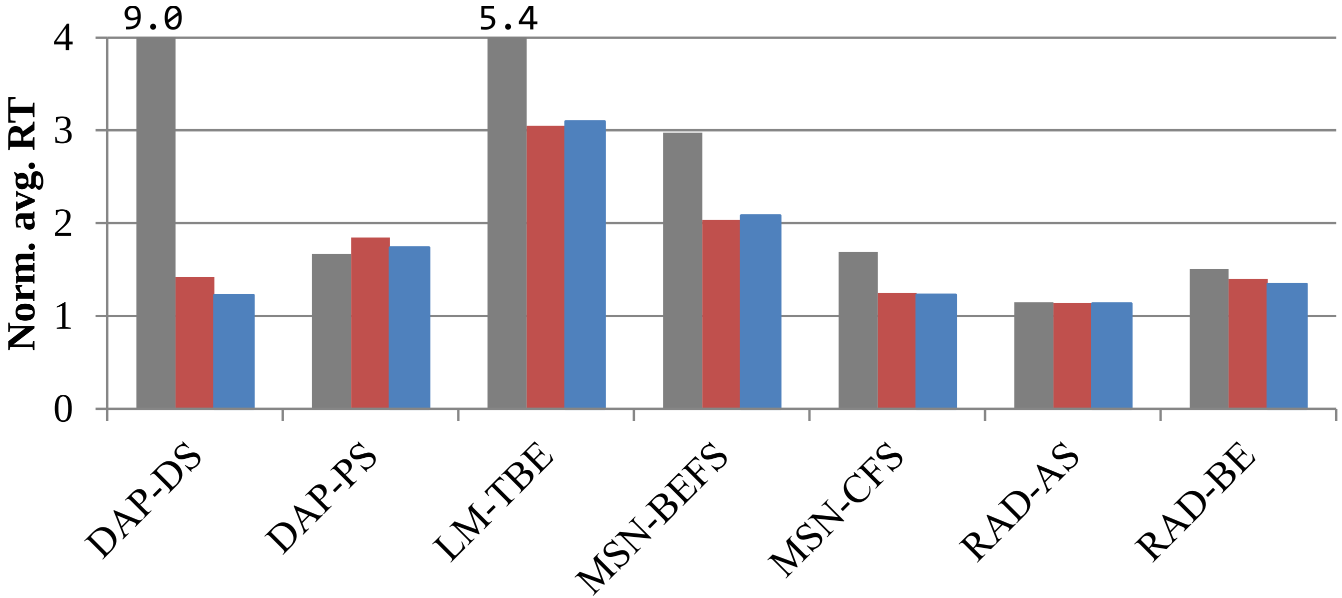
Holistic reliability management

- Cold data
  - Selective redundancy for GC-ed data
- Read-hot data
  - Cost-benefit scrubbing
- Write-hot data
  - No special attention required

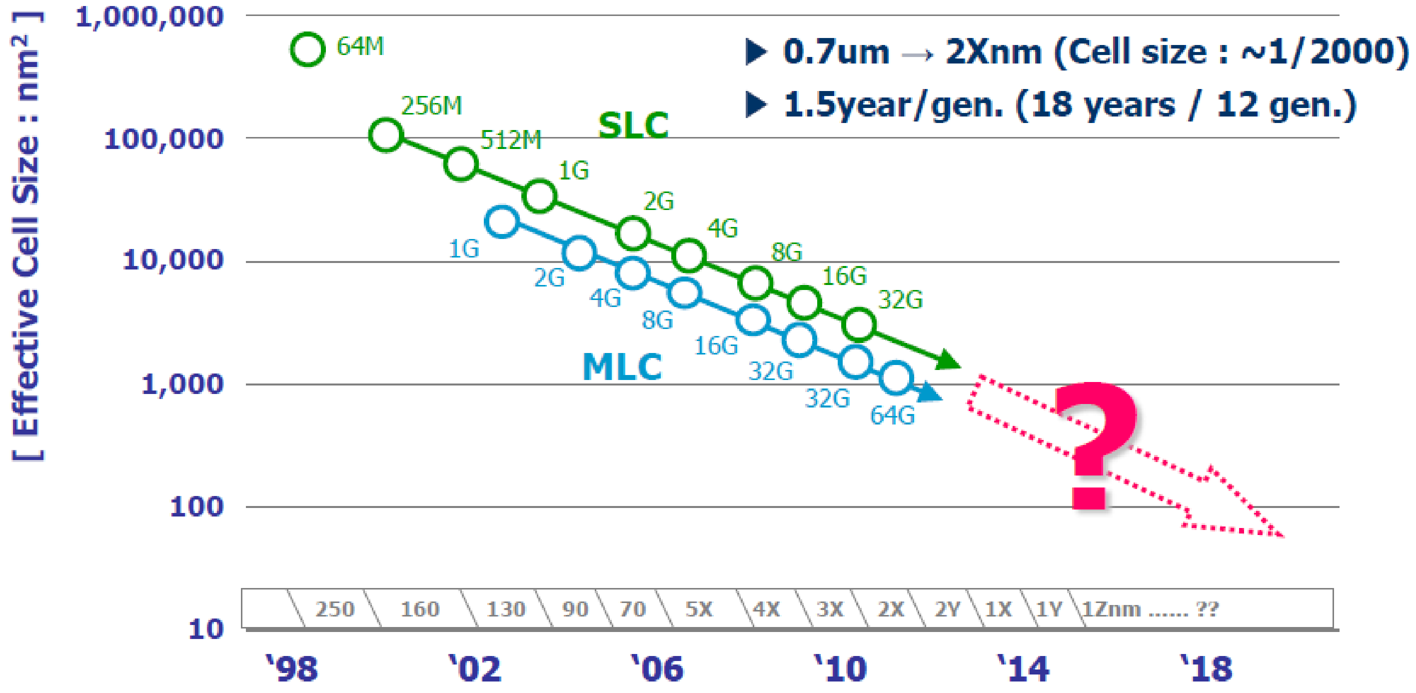
Evaluation

For the 3D TLC (2018)
with 75-bit ECC
@ end-of-life state

ECC + re-read : Rely on ECC and data re-reads
Oracle scrub : Scrub based on oracle knowledge
HRM : Holistic reliability management



The bright side of flash memory



- S. Lee, "Emerging Challenges in NAND Flash Technology", Flash Summit 2011

The dark side of flash memory