

# A Study of Invalid Programming in 3D QLC NAND Flash Memories

Hongyang Dang, Xiangyu Yao, Zheng Wan, Qiao Li  
School of Informatics, Xiamen University



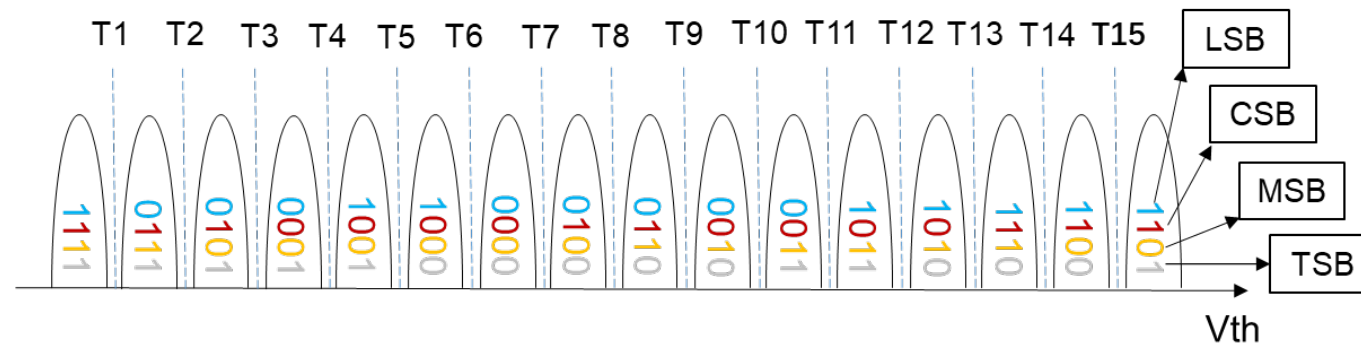
The 15<sup>th</sup> ACM Workshop on  
Hot Topics in Storage and File Systems

July 9  
Boston, MA, USA

- Two-step programming in QLC flash
- Invalid programming issue
- Characteristics of invalid programming
  - Evaluation setup
  - Time period between two programming steps
  - Severity of invalid programming
  - Breakdown of WLS
- Initial ideas for handling invalid programming issue
  - Reorganize data to alleviate invalid programming
  - Leverage invalid pages to improve reliability
- Conclusions

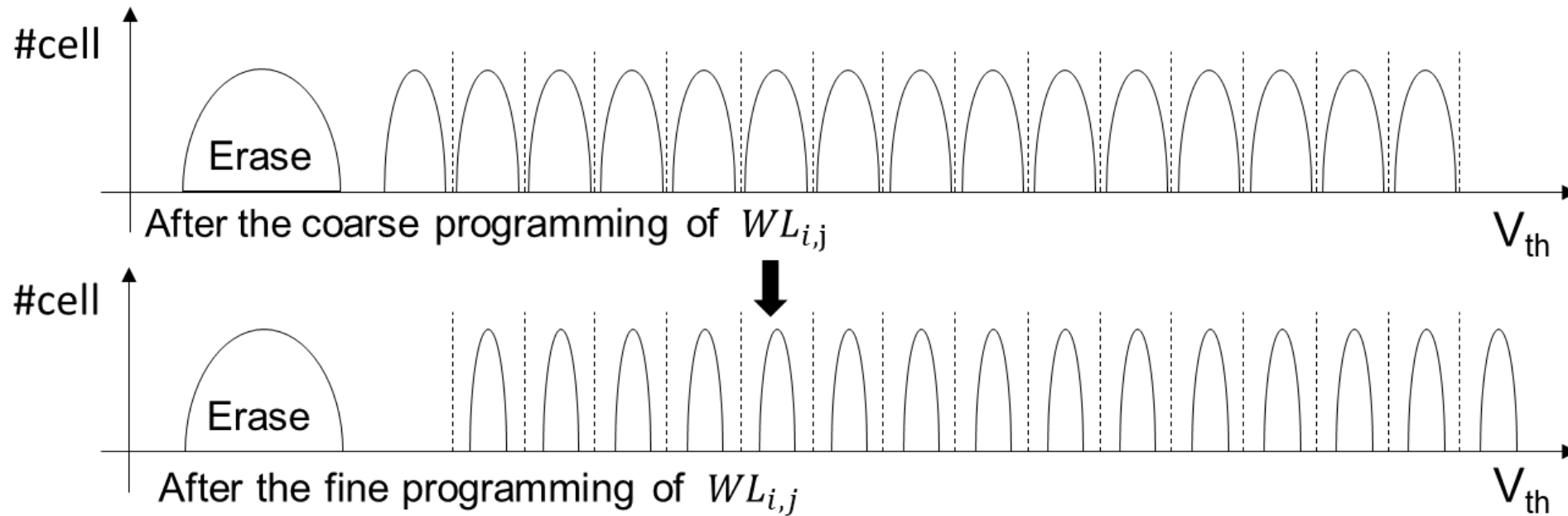
# 3D QLC flash

- 3D quad-level-cell (QLC) flash
  - High density: 4 bits per cell
  - More voltage states: 16 levels
  - Tight and narrow state distribution



# Two-step programming in QLC flash

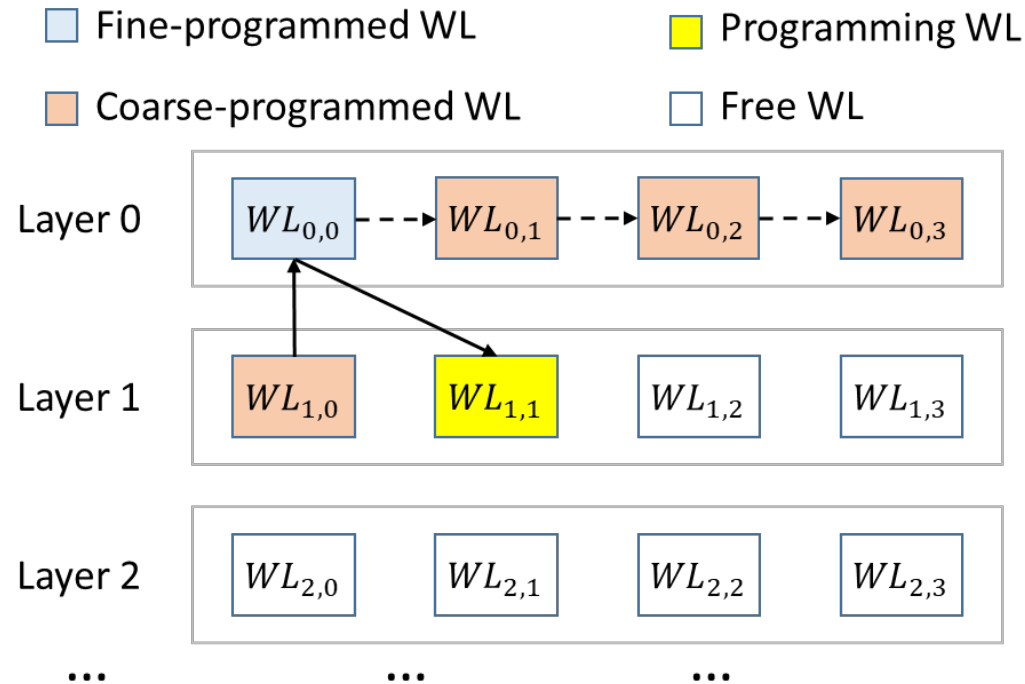
- **Coarse** programming step
  - Cells are **first** programmed to **coarse** 16 levels.
- **Fine** programming step
  - Cells are **then** programmed to the **final** 16 levels.



# Two-step programming in QLC flash

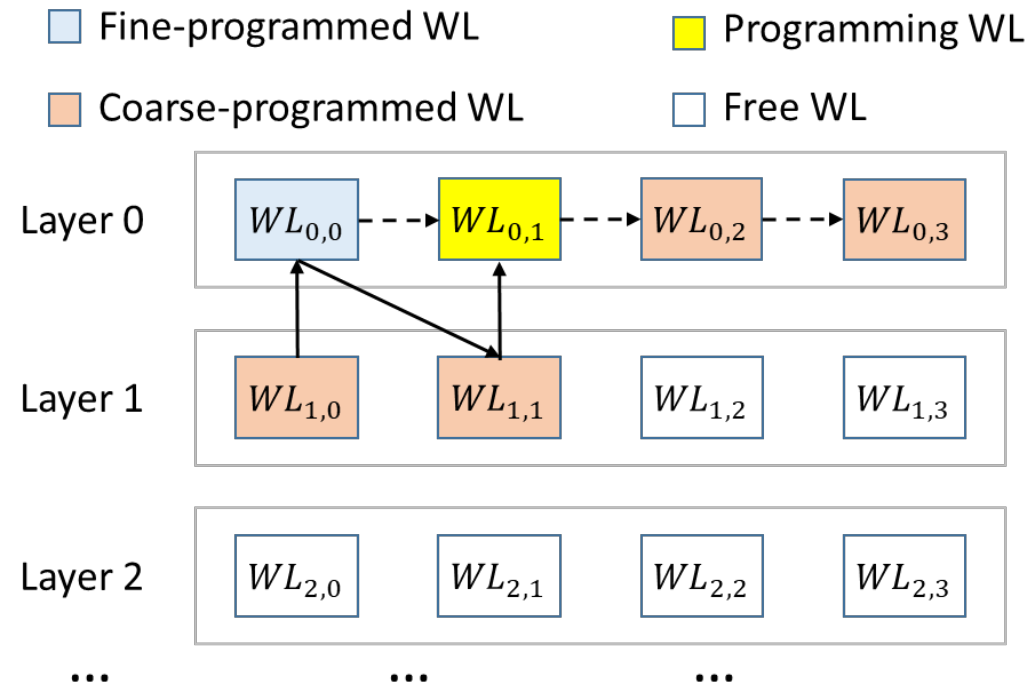
- Programming sequence:

--Alternate coarse and fine programming between the lower and upper layers.



# Two-step programming in QLC flash

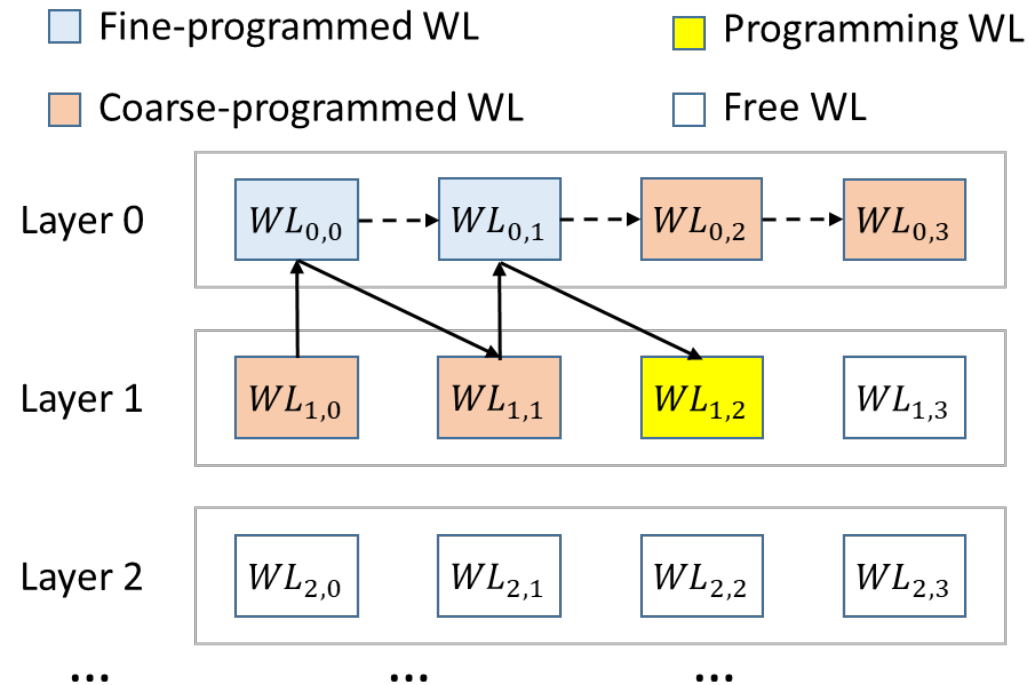
- Programming sequence:
  - Alternate coarse and fine programming between the lower and upper layers.



# Two-step programming in QLC flash

- Programming sequence:

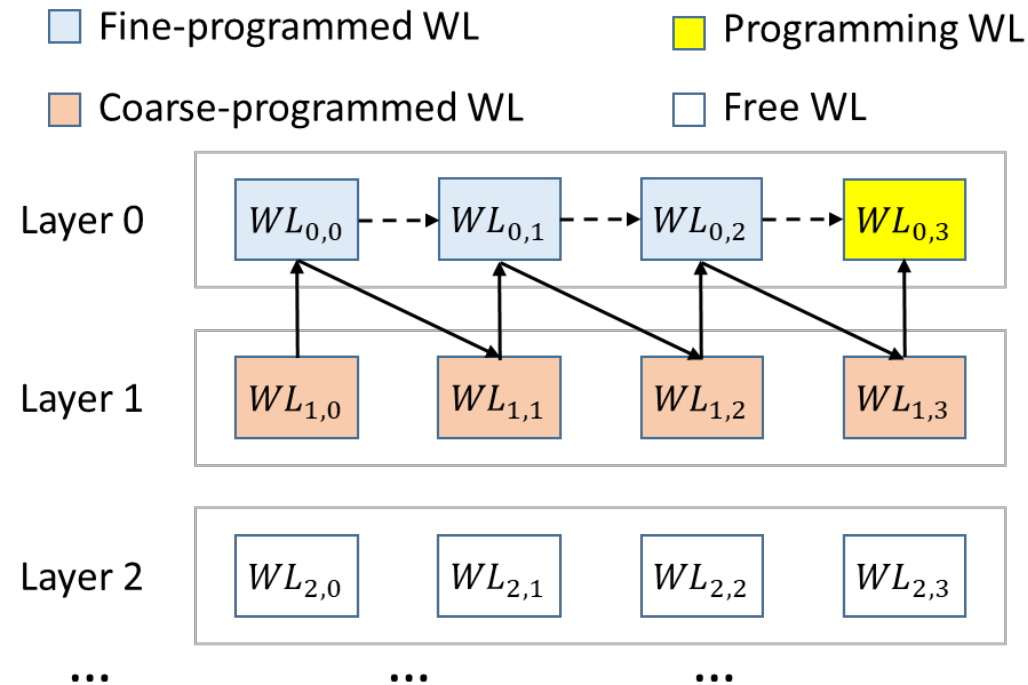
--Alternate coarse and fine programming between the lower and upper layers.



# Two-step programming in QLC flash

- Programming sequence:

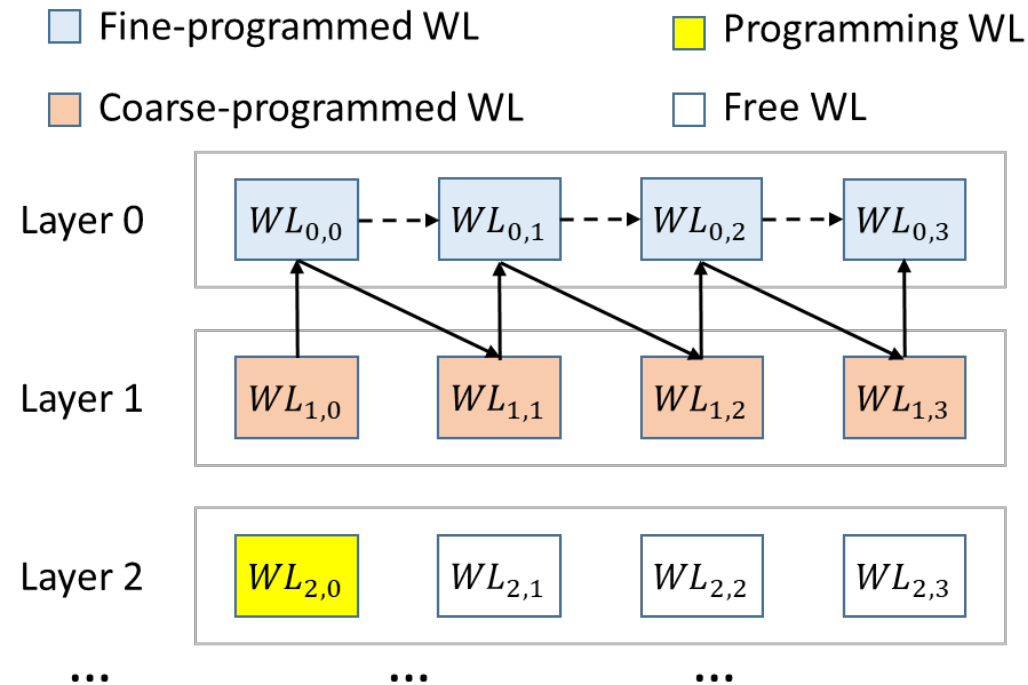
--Alternate coarse and fine programming between the lower and upper layers.





# Two-step programming in QLC flash

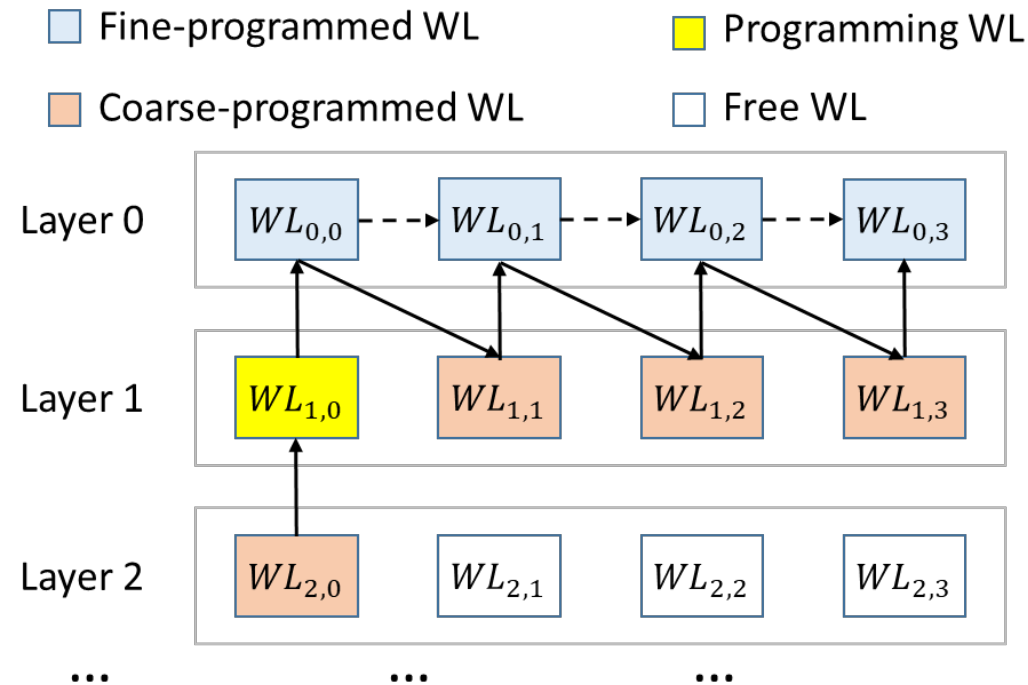
- Programming sequence:
  - Alternate coarse and fine programming between the lower and upper layers.



# Two-step programming in QLC flash

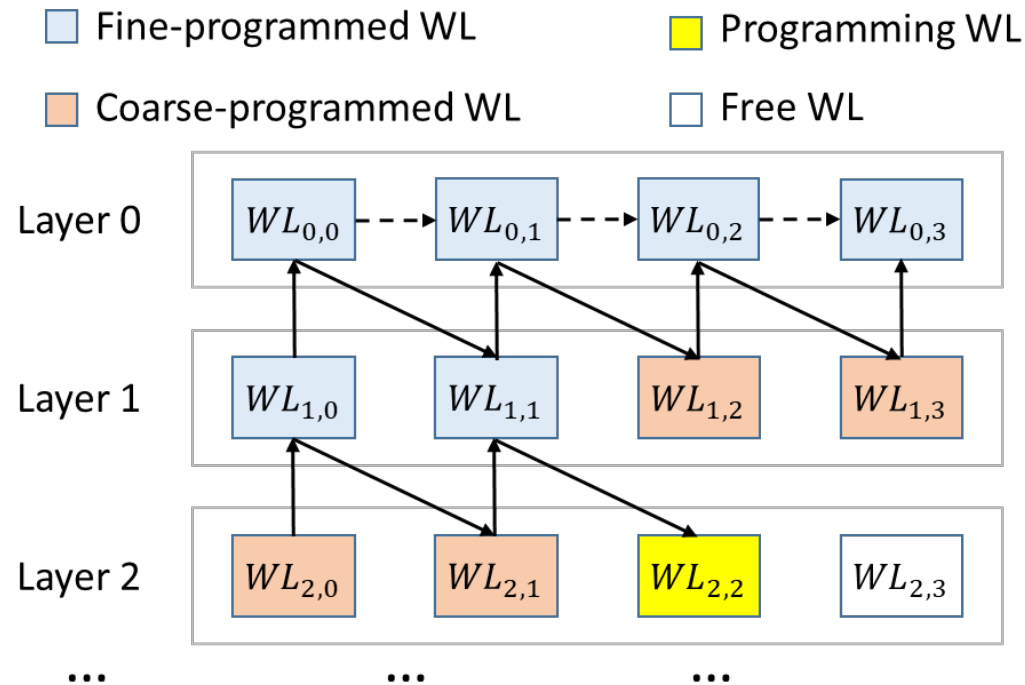
- Programming sequence:

--Alternate coarse and fine programming between the lower and upper layers.



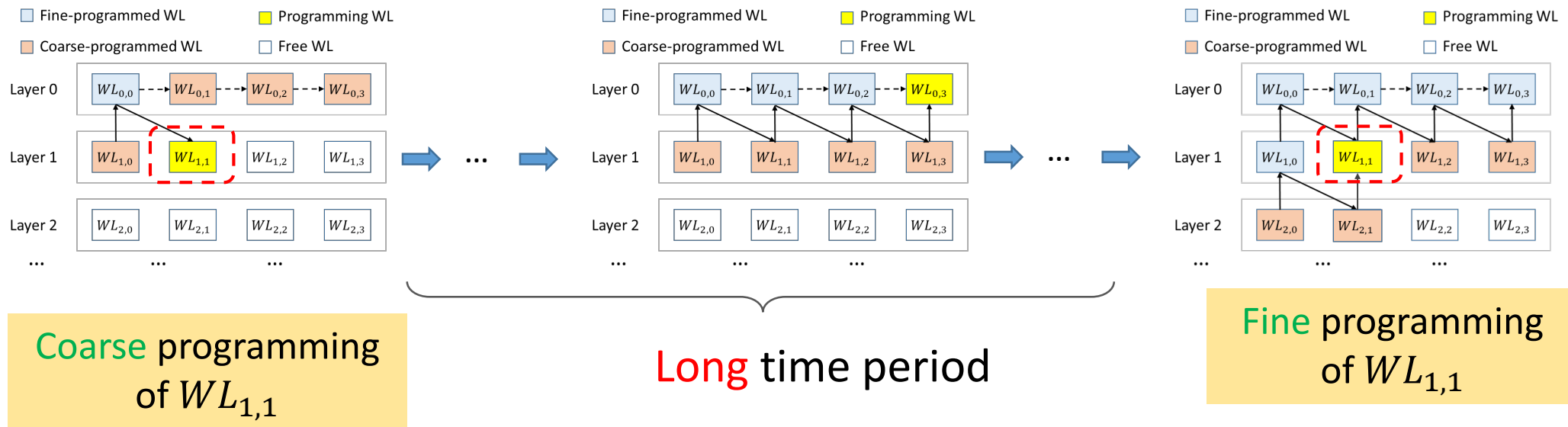
# Two-step programming in QLC flash

- Programming sequence:
  - Alternate coarse and fine programming between the lower and upper layers.



# Two-step programming in QLC flash

- Long time period between the two programming steps of  $WL_{i,j}$ 
  - Fine programming of  $WL_{i,j}$  is delayed until  $WL_{i+1,j}$  completes its coarse programming step.



# Invalid programming issue

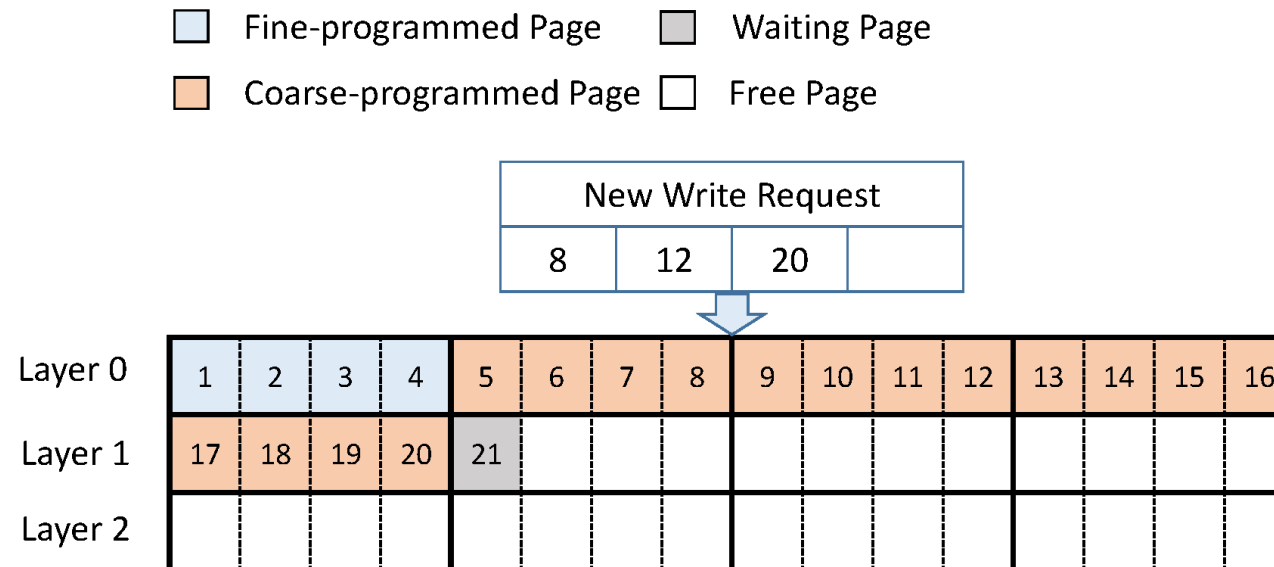
- Invalid programming: **program invalid data** in fine programming step.
- The coarse-programmed page cache may be **invalidated** during the time period between two programming steps.

□ Fine-programmed Page    □ Waiting Page  
□ Coarse-programmed Page    □ Free Page

Layer 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Layer 1	17	18	19	20	21											
Layer 2																

# Invalid programming issue

- Invalid programming: **program invalid data** in fine programming step.
- The coarse-programmed page cache may be **invalidated** during the time period between two programming steps.



# Invalid programming issue

- Invalid programming: **program invalid data** in fine programming step.
- The coarse-programmed page cache may be **invalidated** during the time period between two programming steps.

□ Fine-programmed Page    □ Waiting Page  
□ Coarse-programmed Page    □ Free Page

Layer 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Layer 1	17	18	19	20	21	8	12	20								
Layer 2																

# Invalid programming issue

- Invalid programming: **program invalid data** in fine programming step.
- The coarse-programmed page cache may be **invalidated** during the time period between two programming steps.

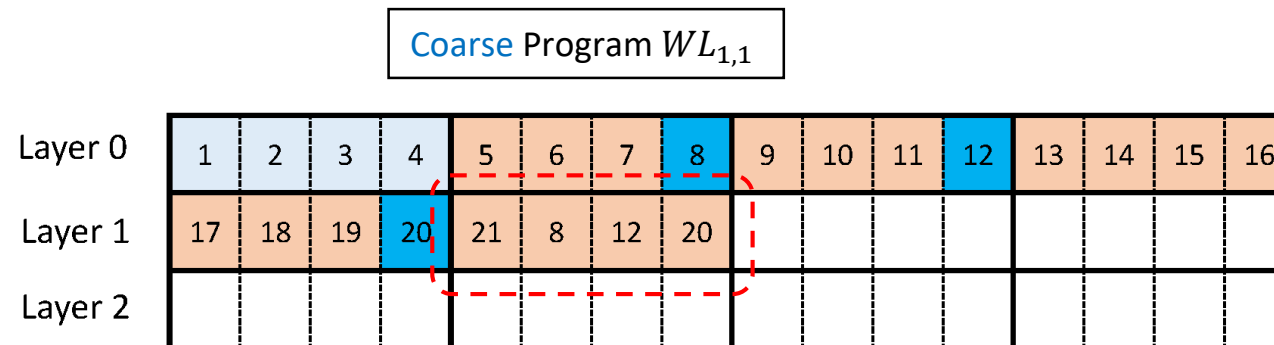
□ Fine-programmed Page   □ Waiting Page   ■ Invalidated Page  
□ Coarse-programmed Page   □ Free Page

Layer 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Layer 1	17	18	19	20	21	8	12	20								
Layer 2																



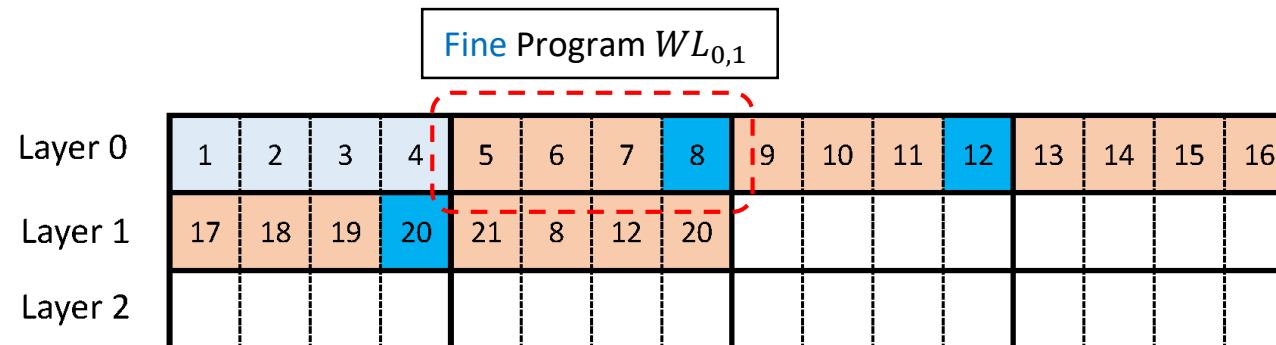
# Invalid programming issue

- Invalid programming: program invalid data in fine programming step.
- The coarse-programmed page cache may be invalidated during the time period between two programming steps.



# Invalid programming issue

- Invalid programming: program invalid data in fine programming step.
- The coarse-programmed page cache may be invalidated during the time period between two programming steps.



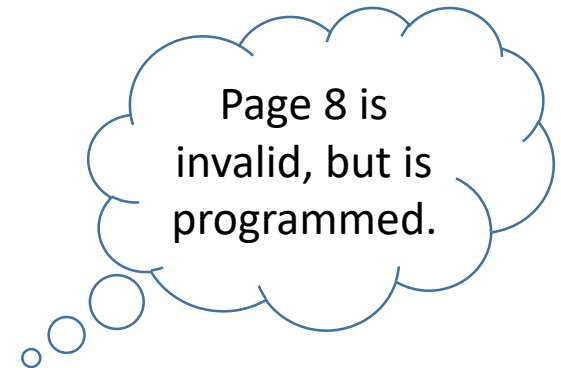
# Invalid programming issue

- Invalid programming: program invalid data in fine programming step.
- The coarse-programmed page cache may be invalidated during the time period between two programming steps.

■ Fine-programmed Page   ■ Waiting Page   ■ Invalidated Page  
■ Coarse-programmed Page   ■ Free Page

Fine Program  $WL_{0,1}$

Layer 0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Layer 1	17	18	19	20	21	8	12	20								
Layer 2																



# Evaluation setup

- Platform:

  - SimpleSSD

- Hybrid SSD:

  - SLC region: store hot data.

  - QLC region : store cold data and migrated data.

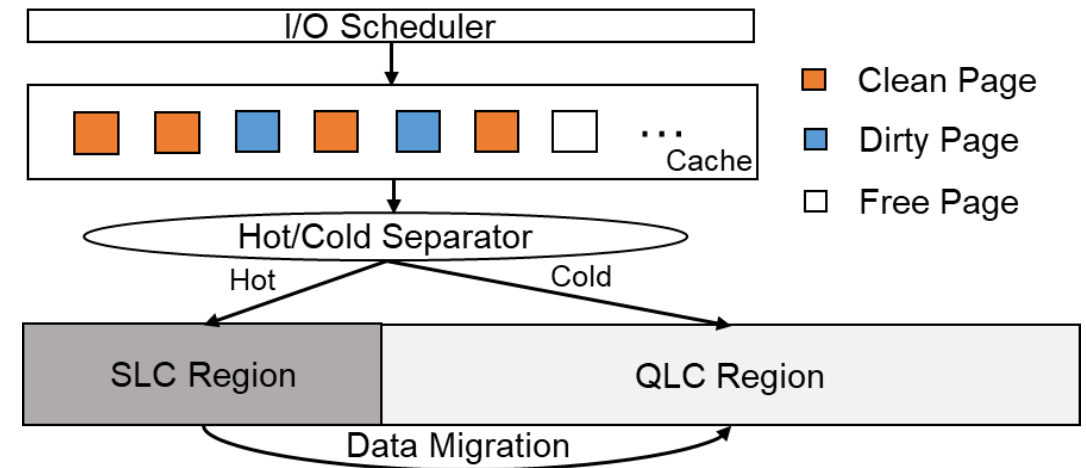
- Metrics:

  - Invalid ratio:

$$\text{Invalid Ratio} = \frac{\text{Invalid Page Count}}{\text{Page Count}} \quad (1)$$

  - Write back ratio:

$$\text{Write Back Ratio} = \frac{\text{Write Back Page Count}}{\text{Request Page Count}} \quad (2)$$

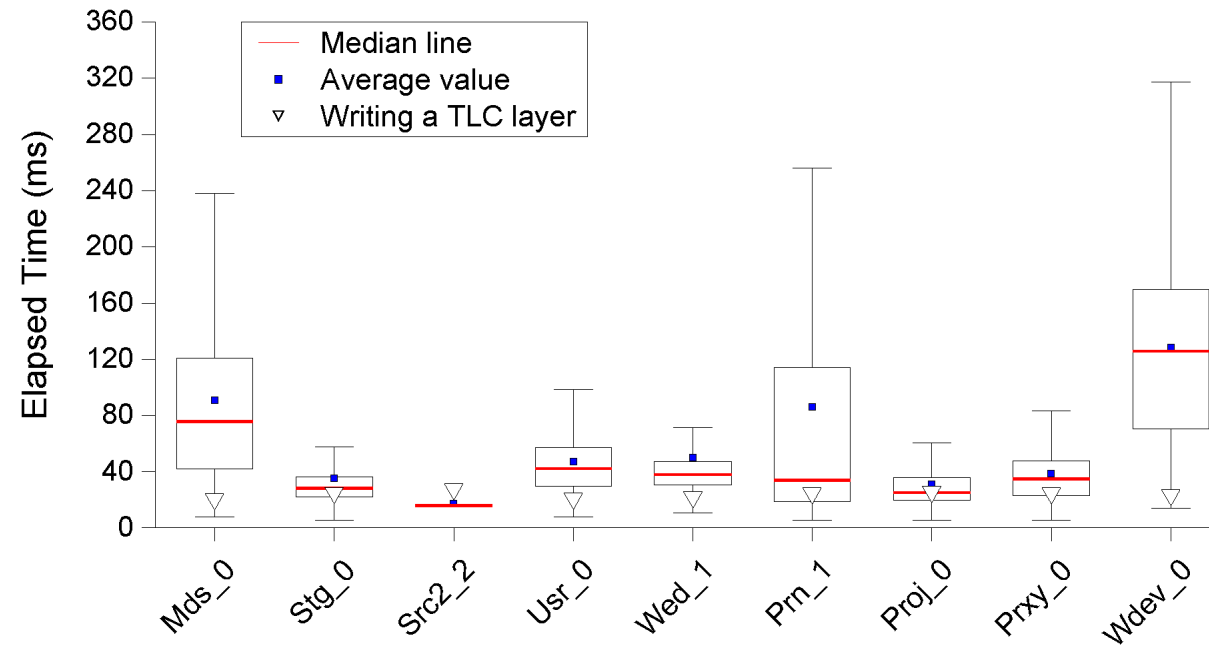


# Time period between two programming steps

- Elapsed Time:

- Much longer than the average programming time of a layer in TLC flash.

- Vary from trace to trace.



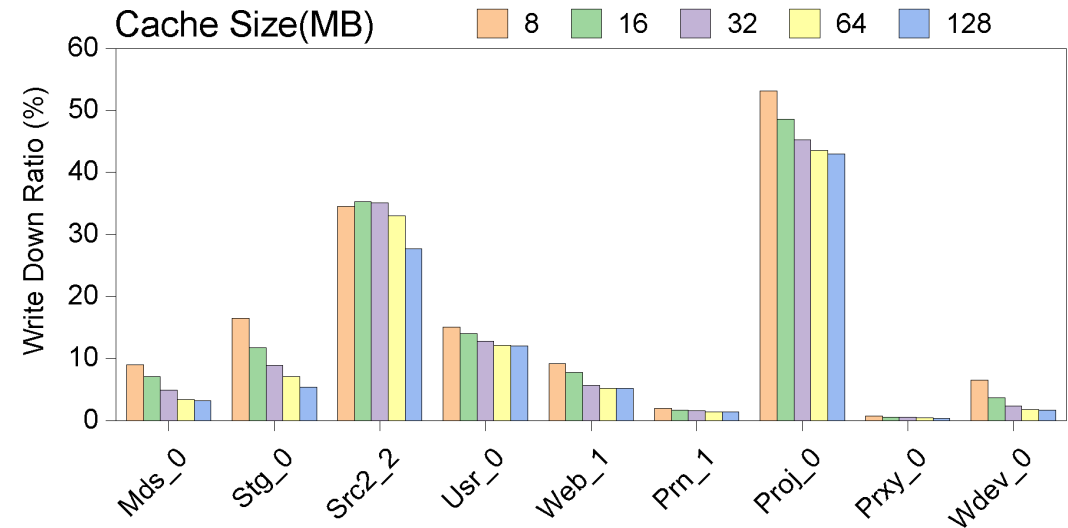
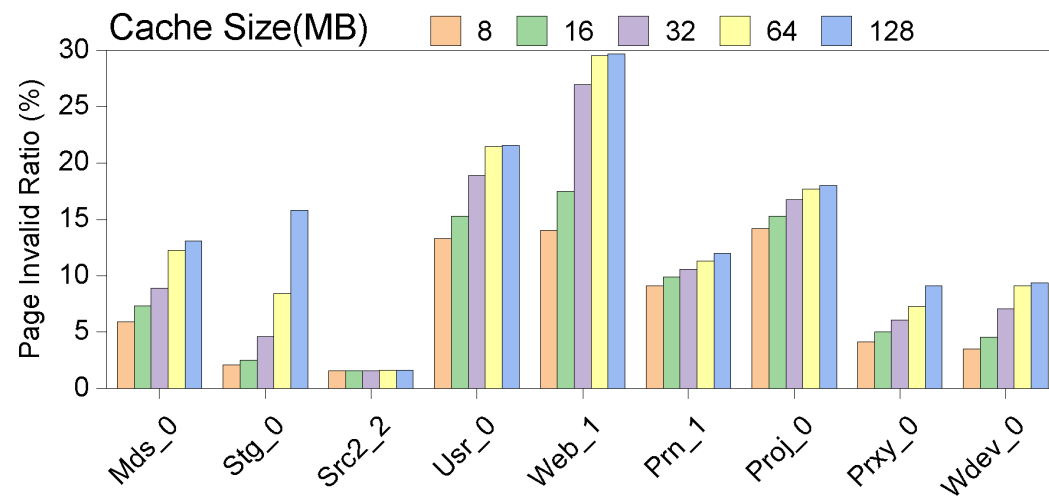
# Severity of invalid programming

- Evaluation factors:

Factor	Description	Default Value
cache size	the size of data cache	16MB
block size	the number of pages per block	512
open block count	the maximum number of blocks that can be written in parallel in the QLC region	8
LRU length	the length of the LRU list used for separating cold and hot data	64

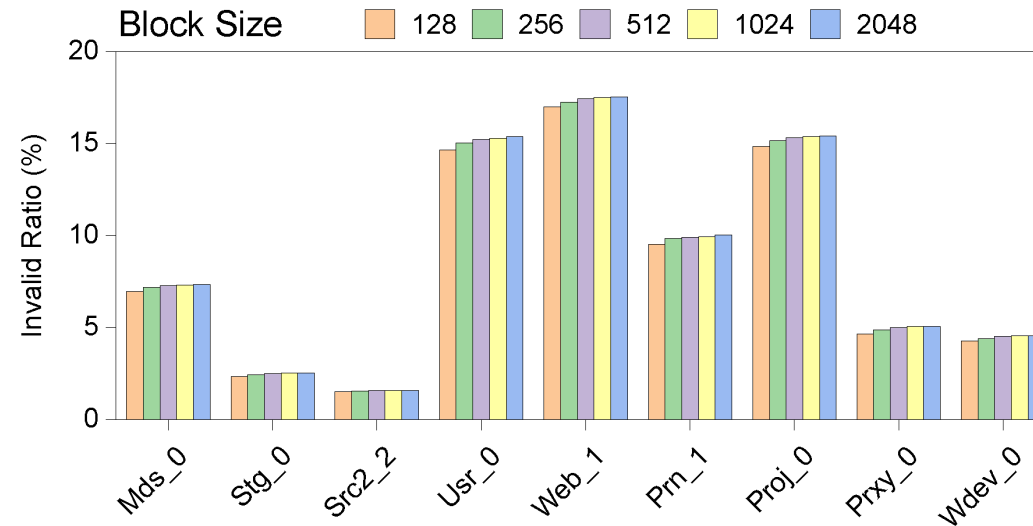
# Impacts of cache size

- The invalid ratio increases as the cache size increases.
- Cache size determines the amount of data written back to the underlying flash memory.
  - The larger cache size is, the longer elapsed time becomes.



# Impacts of block size

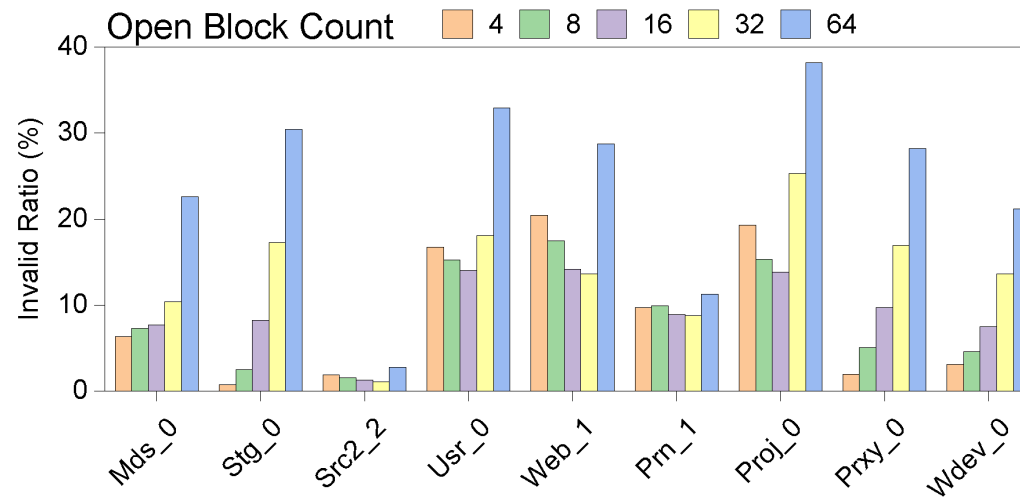
- The invalid ratio increases as the block size increases.
- The ultimate few pages of each open block can be fine-programmed immediately without any delay.
  - The larger block size is, the fewer such pages become.





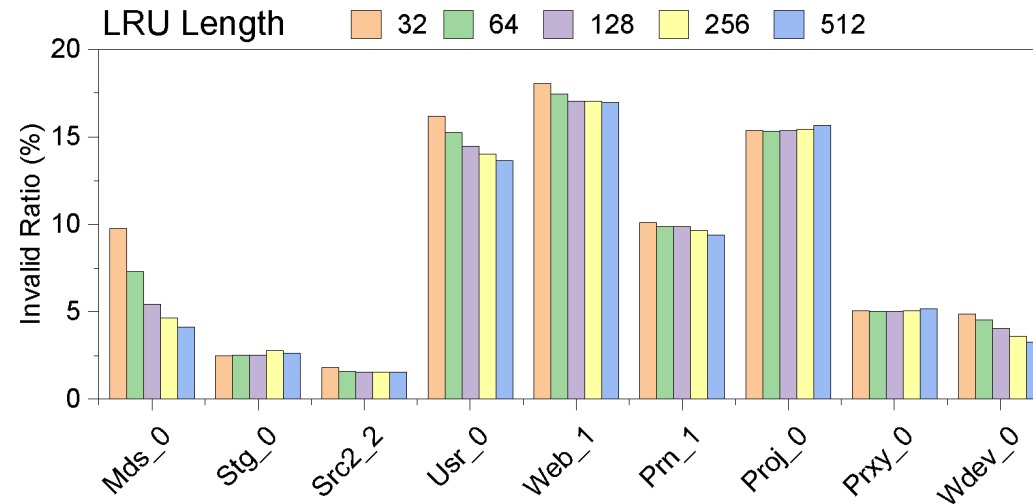
# Impacts of open block count

- The precise correlation between the number of open blocks and the invalid ratio is subject to various factors.
  - More pages awaiting fine programming simultaneously.
  - Pages with the same LBA are more likely to become invalid before performing coarse programming.



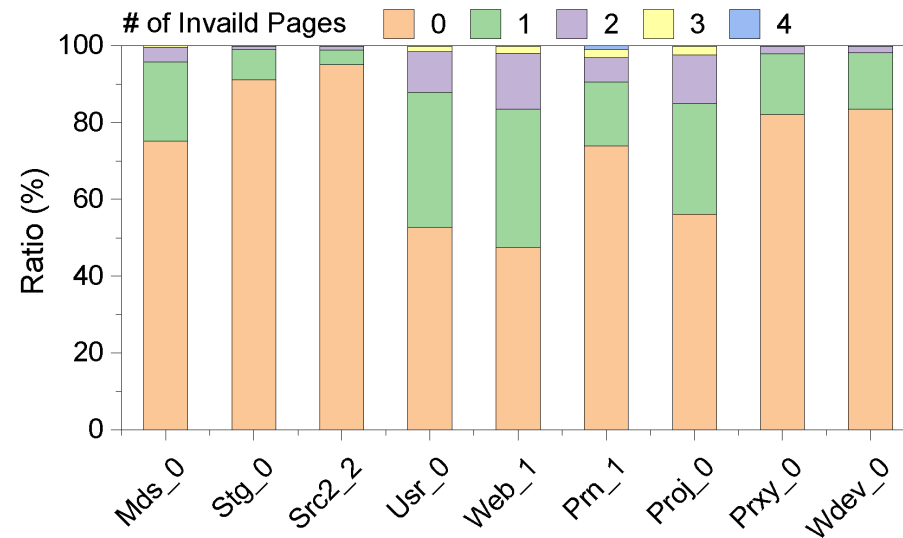
# Impacts of LRU length

- The invalid ratio decreases as the length of the LRU list increases.
- Cold data is less likely to occur invalid programming issue.
  - The larger LRU length is ,the more data is categorized as hot data and subsequently written to SLC region.



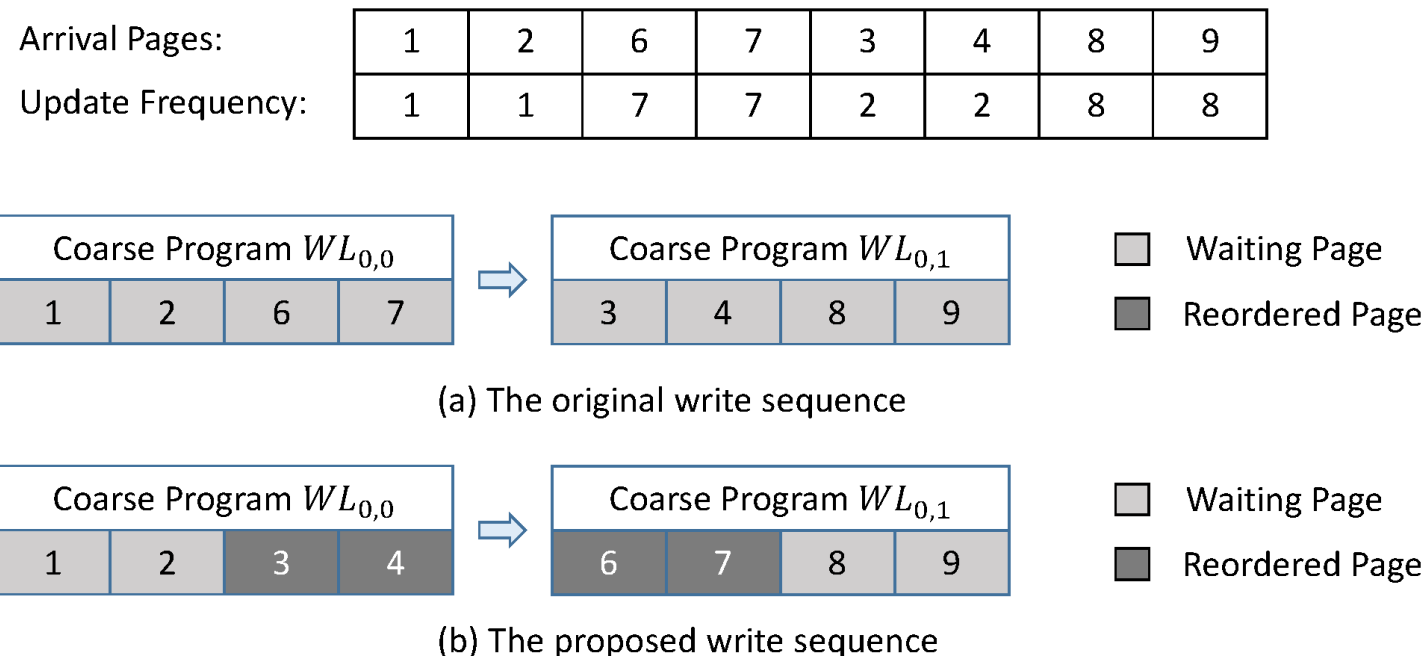
# Breakdown of WLs

- The proportion of WLs with invalid programming may be high, such as *Web\_1*.
- Among WLs with invalid programming, the main proportion is WLs with 1 and 2 invalid pages.



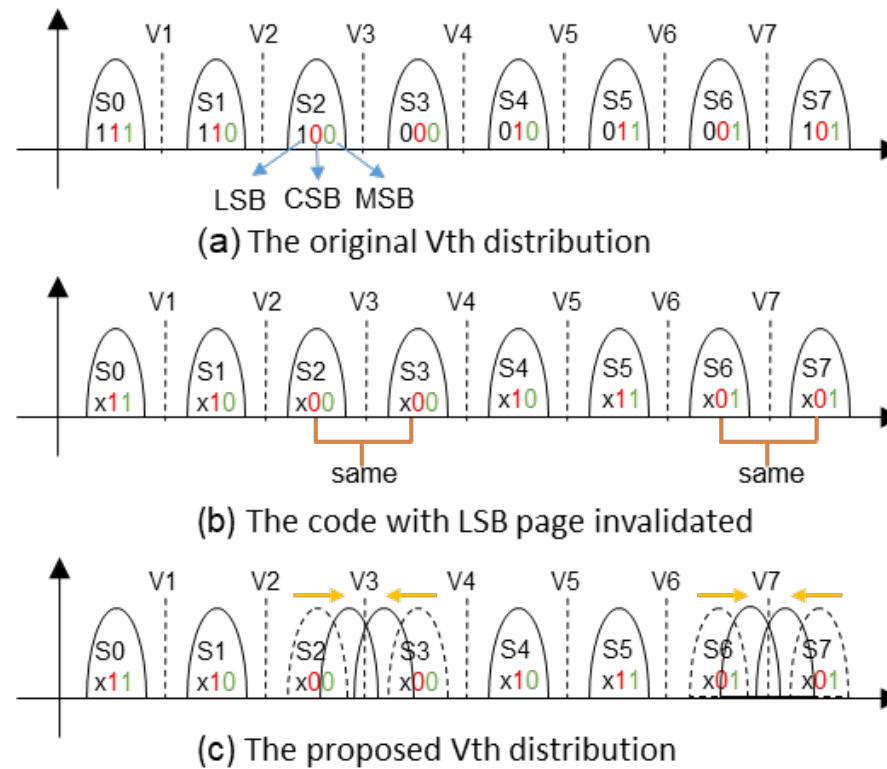
# Reorganize data to alleviate invalid programming

- Reorganize data with similar update frequencies together.



# Leverage invalid pages to improve reliability

- Exploiting the Gray code's characteristics for optimization.



- Two-step programming is widely used in 3D QLC flash.
- Invalid programming issue arises from the long period between the two programming steps.
- The characteristics of invalid programming problems under different factors are presented in detail.
- Two preliminary ideas to deal with invalid programming issue are proposed.

# A Study of Invalid Programming in 3D QLC NAND Flash Memories

Hongyang Dang, Xiangyu Yao, Zheng Wan, Qiao Li  
School of Informatics, Xiamen University



The 15<sup>th</sup> ACM Workshop on  
Hot Topics in Storage and File Systems

July 9  
Boston, MA, USA

Thanks