



Exploring System Challenges of Ultra-Low Latency Solid State Drives

Sungjoon Koh

Changrim Lee, Miryeong Kwon, and Myoungsoo Jung

Computer Architecture and Memory systems Lab

Executive Summary



Motivation. Ultra-low latency (ULL) is emerging, but not characterized by far.

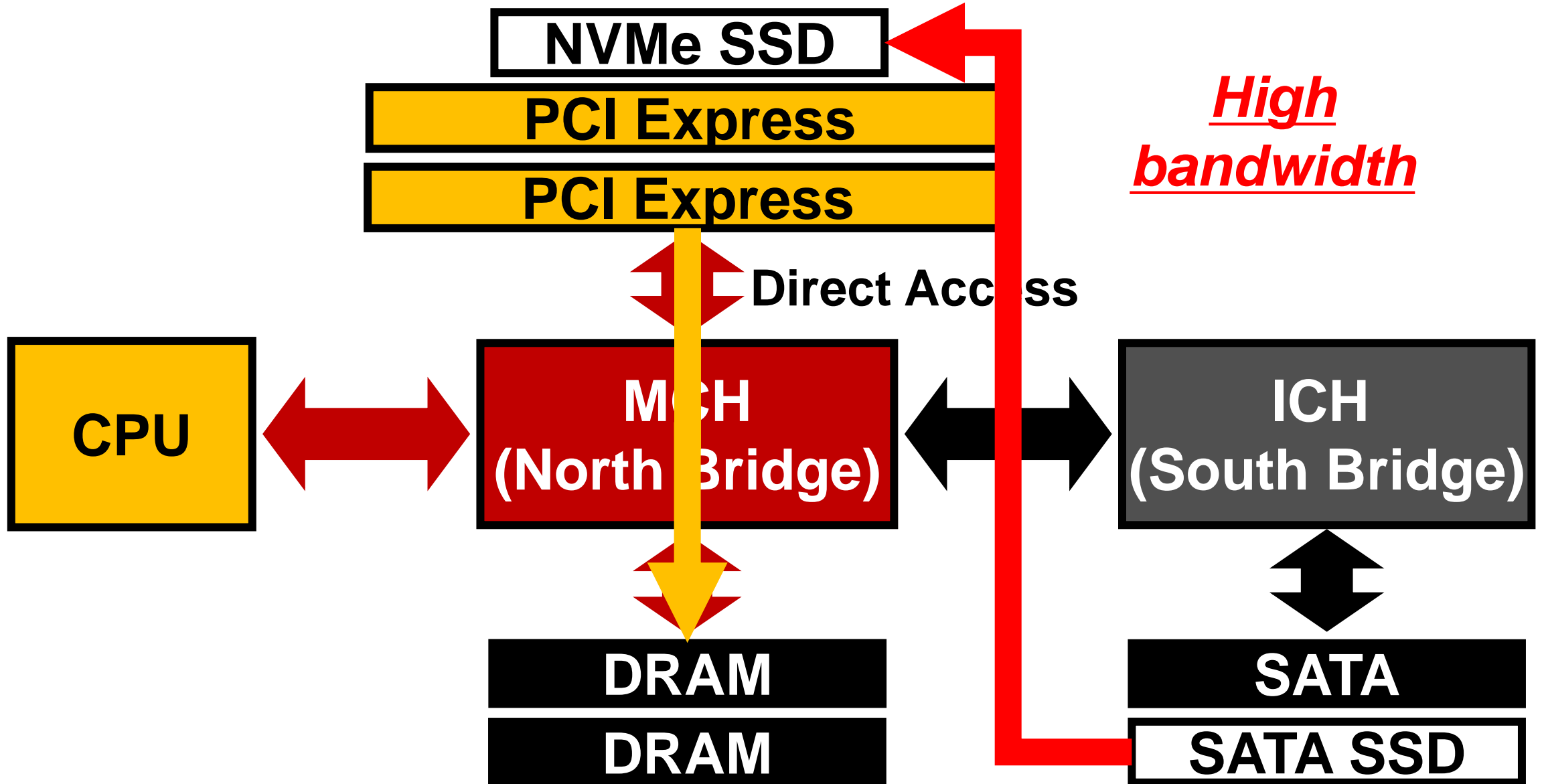
Contributions.

- Characterizing the performance behaviors of ULL SSD.
- Studying several system-level challenges of the current storage stack.

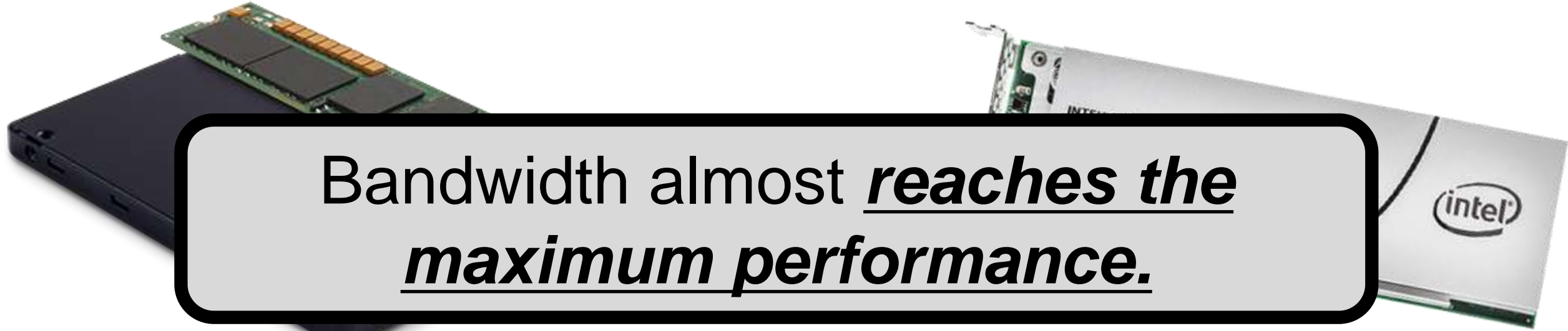
Key Observations.

- ULL SSD minimizes the I/O interferences (interleaving reads and writes).
- NVMe queue mechanisms are required to be optimized for ULL SSDs.
- Polling-based I/O completion routine isn't effective for current NVMe SSDs.

Architectural Change of SSD



Evolution of SSDs



Bandwidth almost reaches the maximum performance.

Still, long latency (far from DRAM)

New flash memory, called “Z-NAND”

SA
Rea
Write: 0.9 GB/s

SD
GB/s
Write: 1.2 GB/s

New Flash Memory

Existing 3D NAND

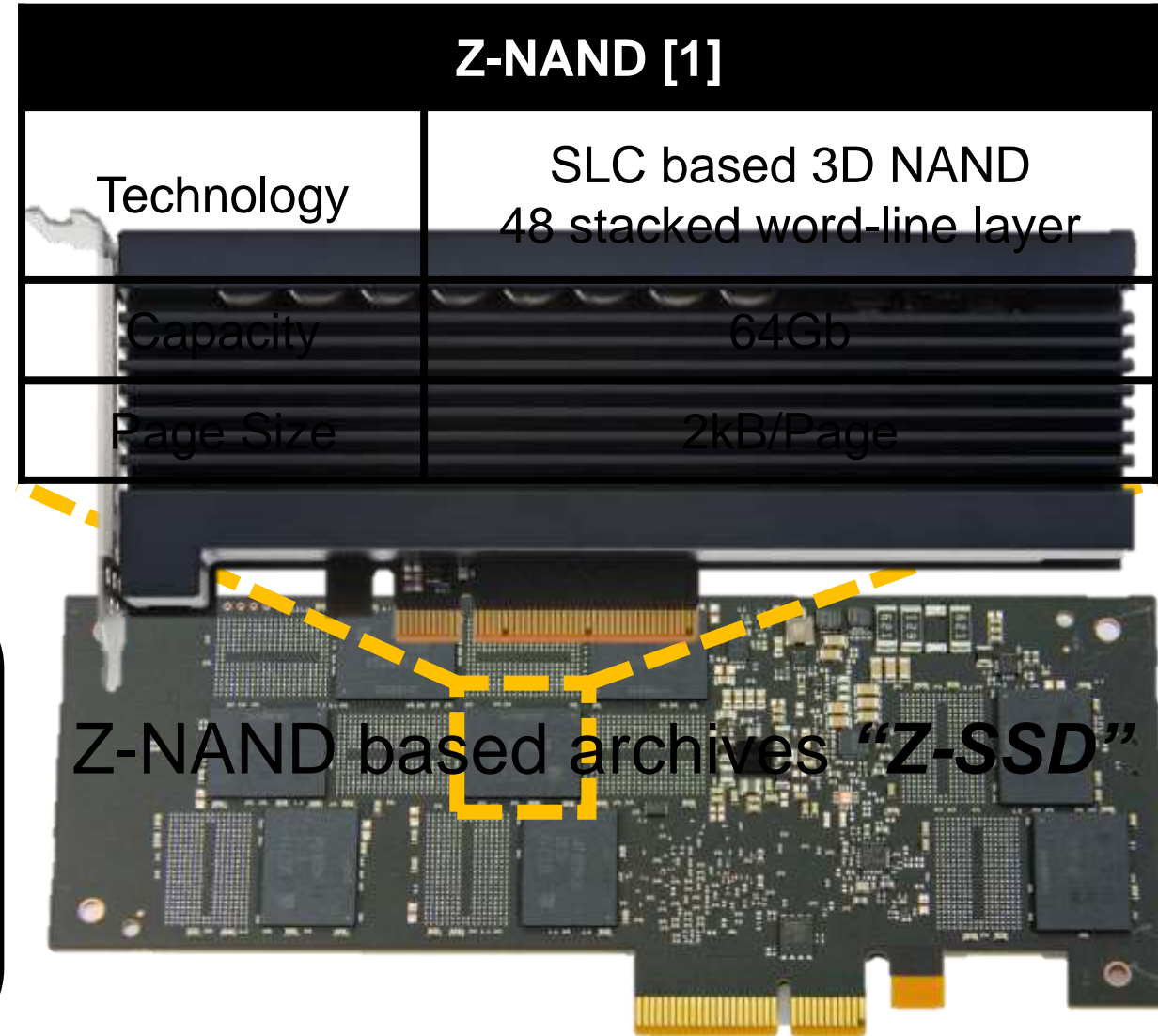
Read: 45-120 μ s

Write: 660-5000 μ s

Z-NAND [1]

Read: 3 μ s (15~20x)

Write: 100 μ s (6~7x)



Characterization Categories



Performance Analysis.

- Average latency.
- Long-tail latency.
- Bandwidth.
- I/O interference impact.

Polling vs. Interrupt

- Overall latency comparison.
- CPU utilization analysis.
- Memory requirement.
- Five-nines latency.

Evaluation Settings



OS: Linux 4.14.10

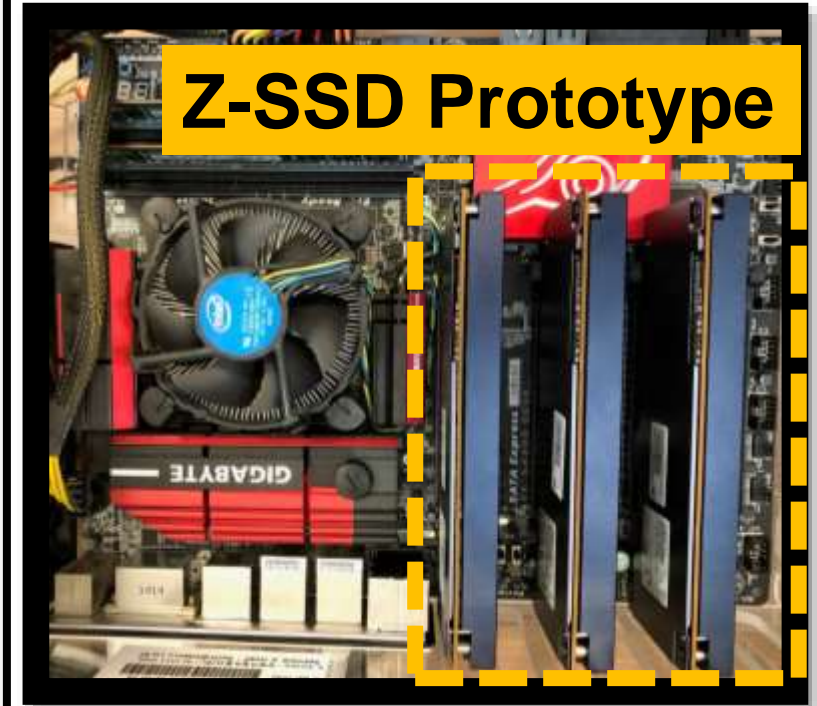
CPU: Intel® Core™ i7-4790K (4-core, 4.00GHz)

Memory: DDR4 DRAM (16GB)

SSD

- **ULL SSD**: Z-SSD Prototype (800GB)
- **NVMe SSD**: Intel® SSD 750 Series (400GB)

Benchmark: Flexible I/O Tester (FIO v2.99)

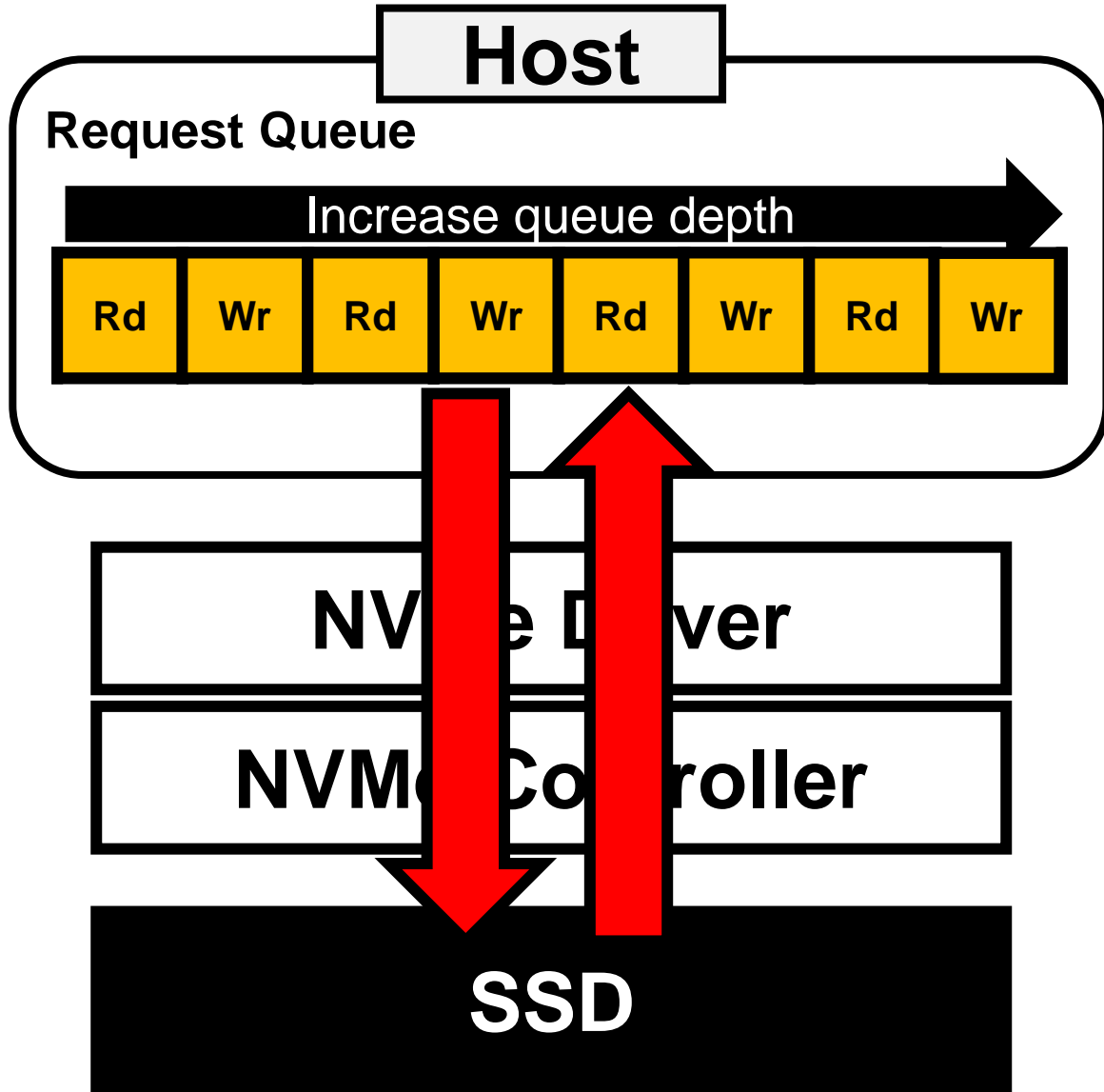


<Our testbed w/ Z-SSDs>



Performance Analysis

Overview

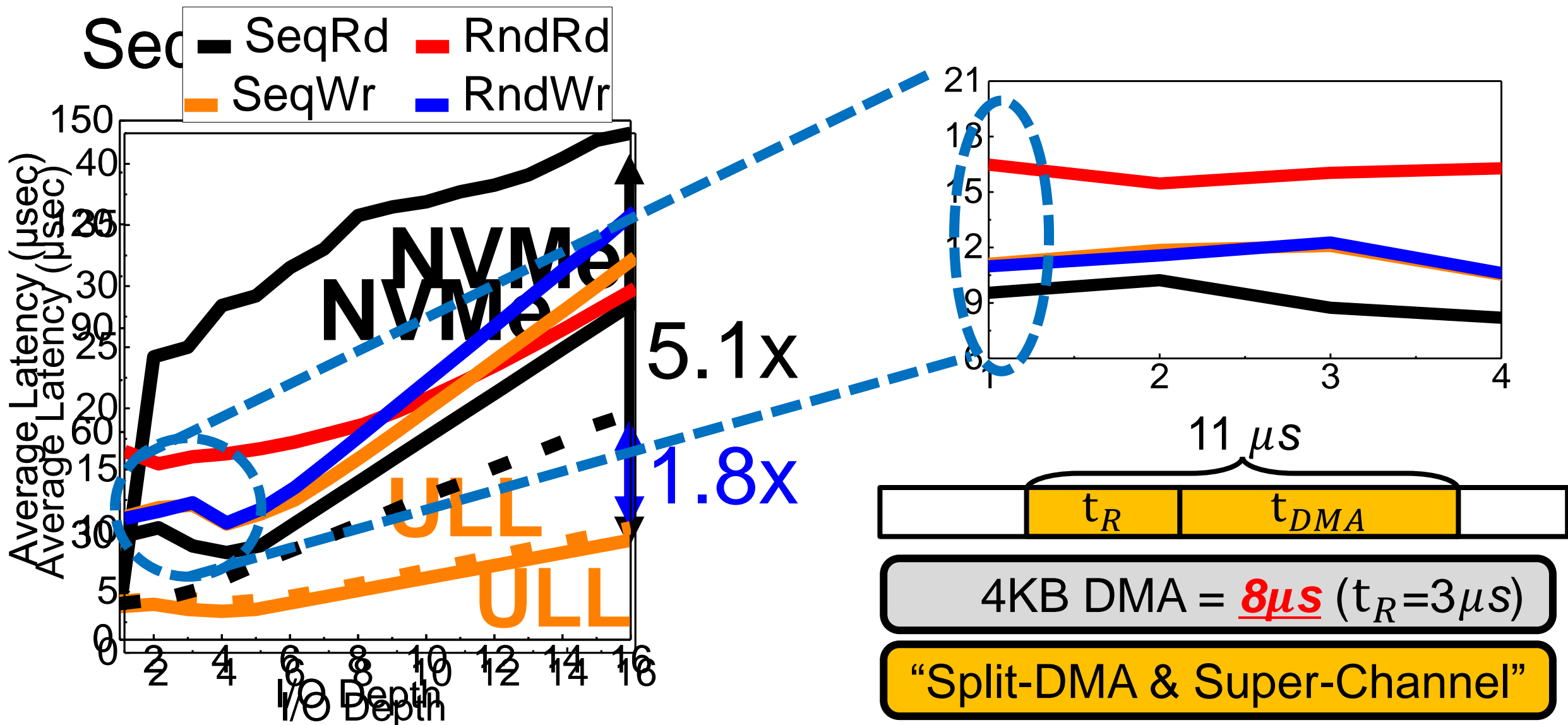


① Average latency & Long-tail latency

② Bandwidth

③ Read latency under Read & Write intermixed workload

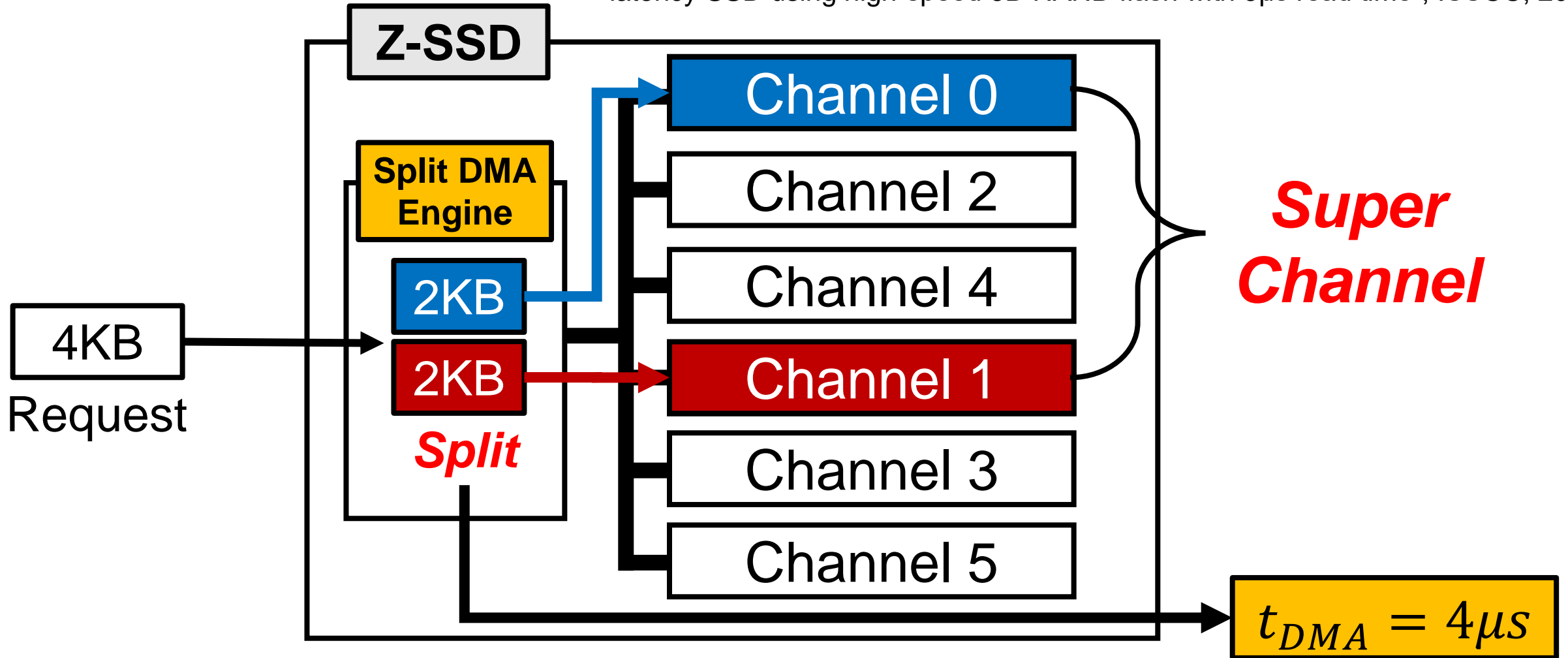
Average Latency of ULL SSD



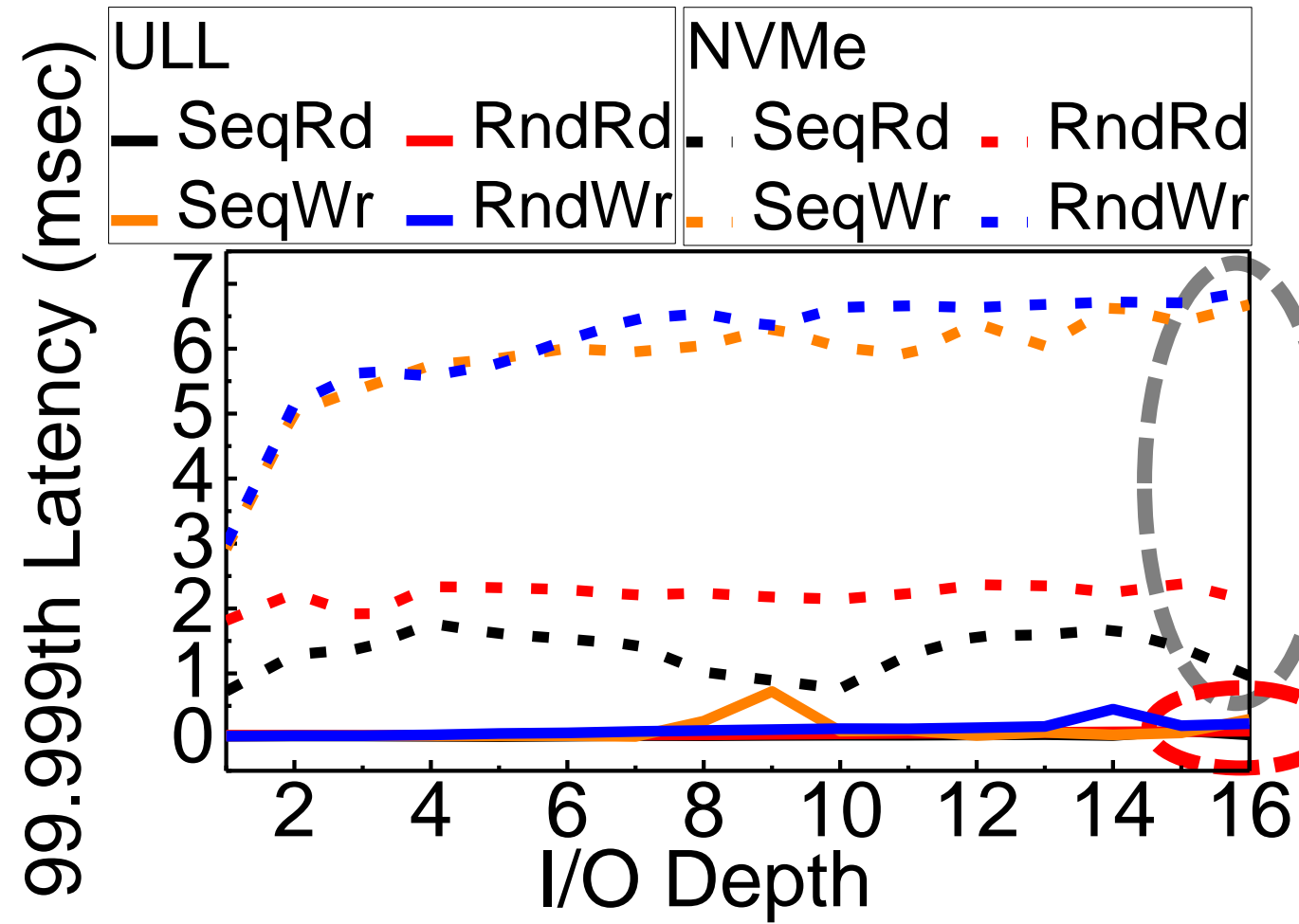
Split-DMA & Super-Channel



Reference: Cheong, Woosung *et al.*, "A flash memory controller for 15 μ s ultra-low-latency SSD using high-speed 3D NAND flash with 3 μ s read time", ISSCC, 2018



Long-tail Latency of ULL SSD



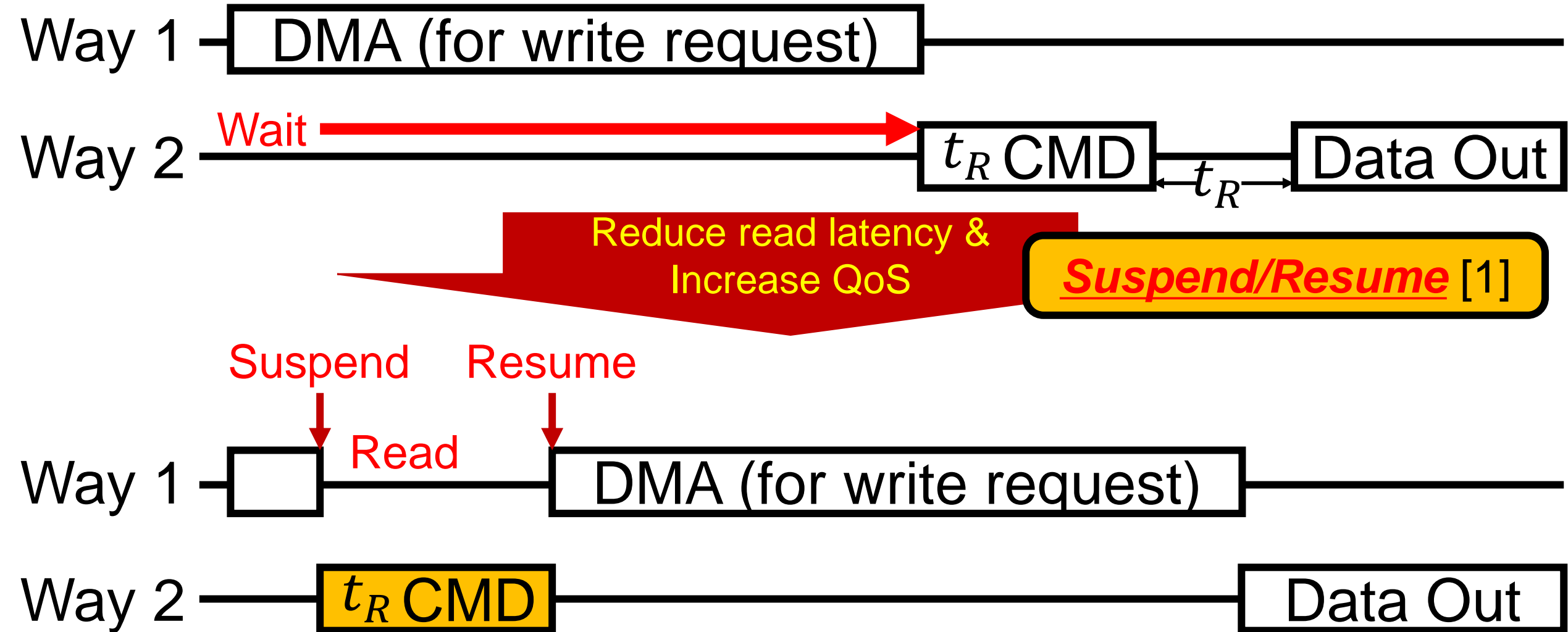
Resource conflict
Insufficient internal buffer,
Internal tasks

“Split DMA” &
“Suspend/Resume”

Suspend/Resume DMA Technique

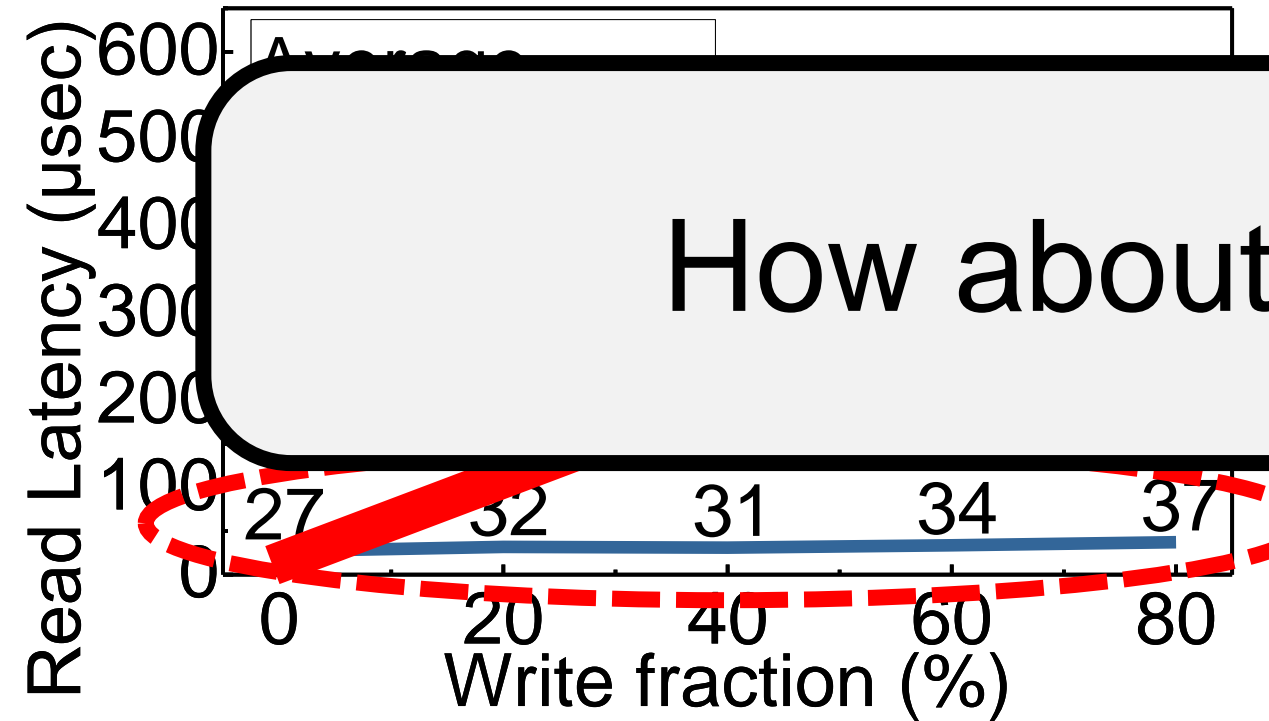


Reference: Cheong, Woosung *et al.*, "A flash memory controller for 15 μ s ultra-low-latency SSD using high-speed 3D NAND flash with 3 μ s read time", ISSCC, 2018



I/O Interference

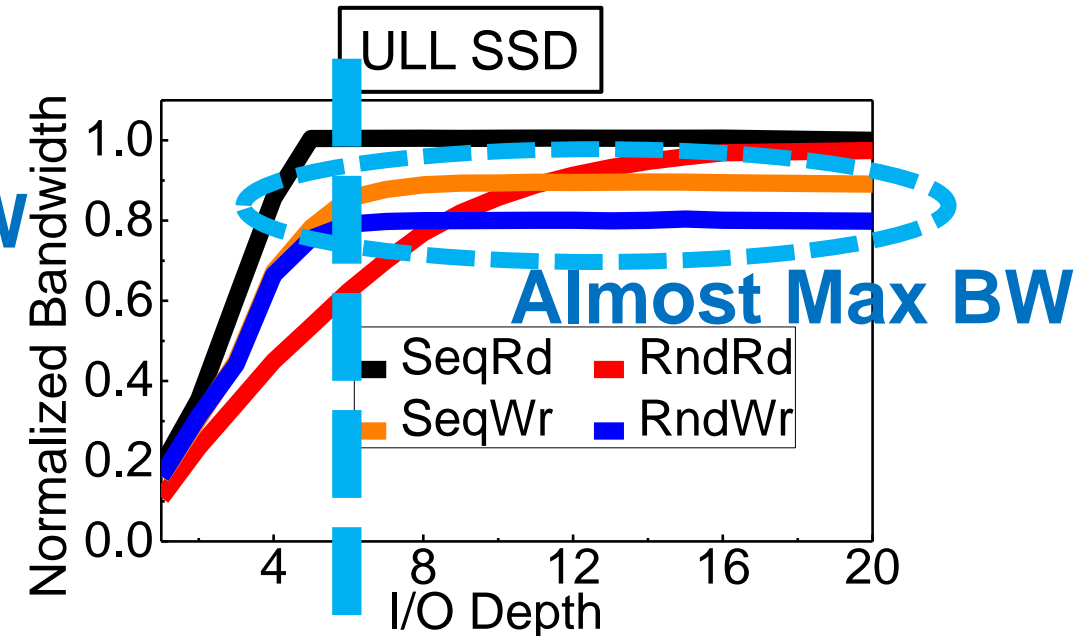
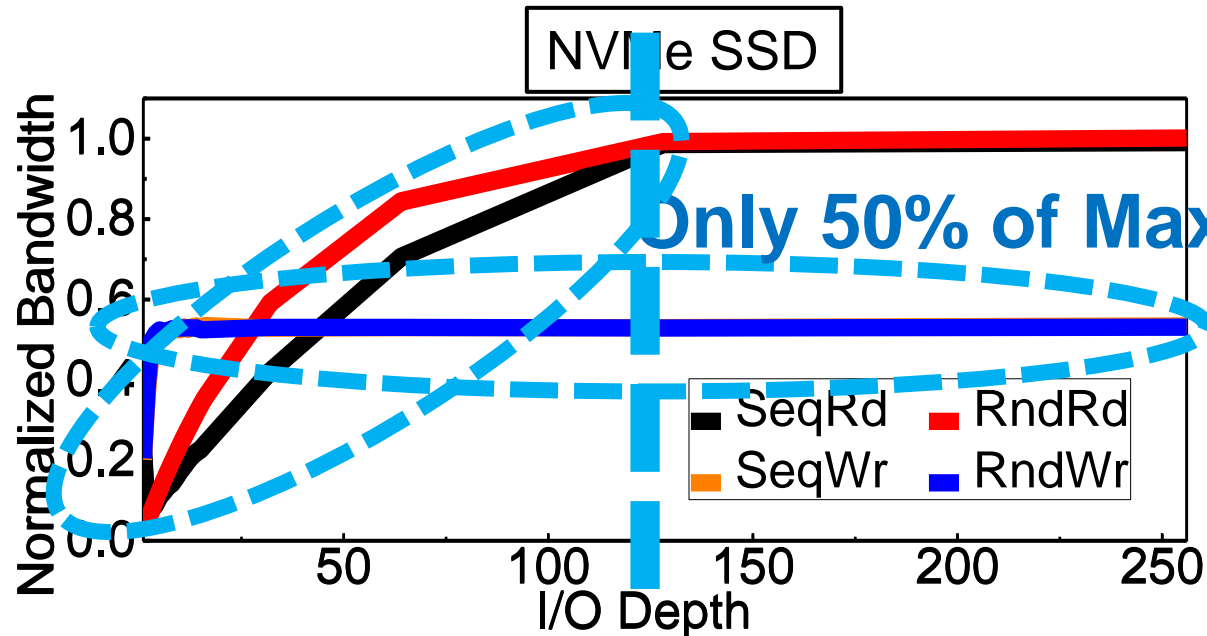
Great ***performance bottleneck*** of conventional SSDs.



ULL SSD can be applied to ***real-life storage stack*** w/o performance degradation.

Remains almost constant
→ “Suspend/resume”, ... [1]

Queue Analysis



I/O request rescheduling within queue.

Short write latency

Light queue mechanisms (ex. NCQ) are not sufficient.
 → Requires rich queue mechanism

Well-aligned with light queue mechanisms (ex. NCQ).
 NVMe needs to be lightened

Polling vs. Interrupt

Two different I/O completion methods

Interrupt / Polling

Systems with *short waiting time* adopts polling-based waiting strategy. (even though it incurs *lots of overheads*)

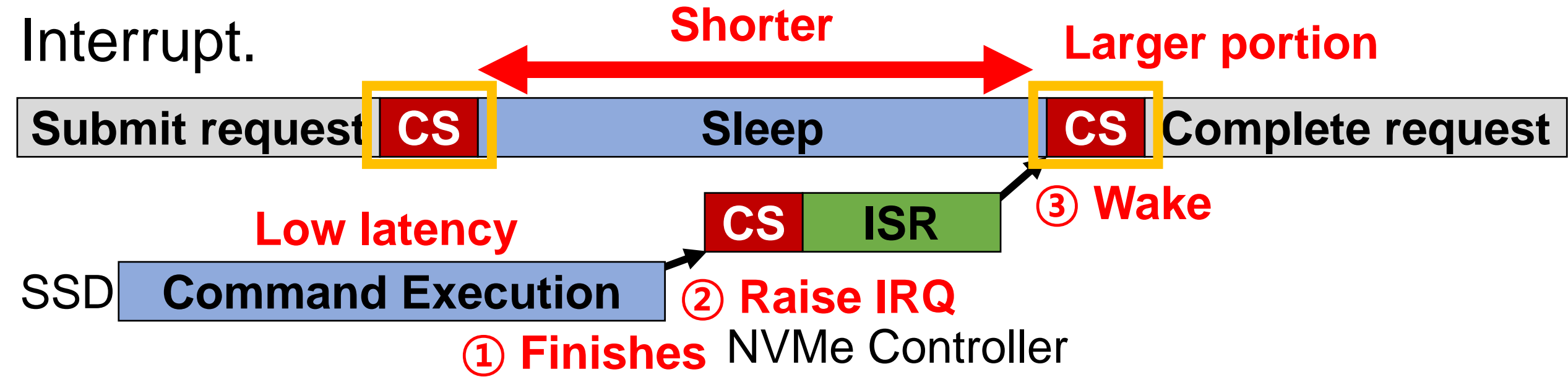
For example, *“spin lock”*, *“network message passing”* applies polling-based waiting strategy.

Polling is currently implemented to NVMe storage stack.

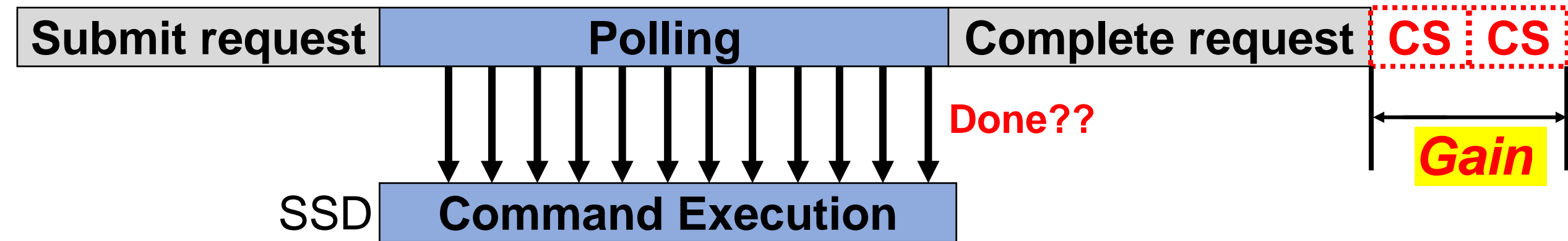
Does it really need for current NVMe SSDs?

Interrupt / Polling

Interrupt.



Polling.



Overall Performance



Future *lower latency SSD* can achieve remarkable performance improvement with polling-based I/O completion routine.

System Challenges

Polling-based
significant system
→ Needs to

Memory bound

= Fraction of slots where pipeline could be stalled due to load/store.

High memory bound

= Frequent memory access

High CPU utilization

<Memory Utilization>

Tail CQ SQ Head

Conclusion

SAMSUNG



Motivation. Ultra-low latency (ULL) is emerging, but not characterized by far.

Contributions.

- Characterizing the performance behaviors of ULL SSD.
- Studying several system-level challenges of the current storage stack.

Key Insights.

- ULL SSDs can be effectively applied to real-life storage stack. (RW mixed)
- NVMe queue mechanisms are required to be optimized for ULL SSDs.
- Polling-based I/O completion routine isn't effective for current NVMe SSDs.



Thank you

Q&A