SAMSUNG

White Paper

Evaluating GPU-Driven Storage Performance with Samsung PM1763 NVMe SSD

Storage performance in GPU-driven data paths

May 2026

Legal Disclaimer

This document has been prepared by Samsung Electronics Co., Ltd. (“Samsung”). The contents of this document are the property of Samsung, protected by applicable laws and non-disclosure agreements between Samsung and you or your employer, as applicable. You are strictly prohibited from, including, but not limited to disclosing, copying, reproducing, distributing, transmitting, modifying, rebroadcasting, re-encoding, re-presenting, exploiting, or creating derivative works of this document or any parts of this document without the prior written permission from Samsung.

The contents of this document are provided for informational purposes only. No representation or warranty (whether express or implied) is made by Samsung or any of its officers, advisers, agents, or employees as to the accuracy, reasonableness or completeness of the information, statements, opinions, or matters contained in this document, and they are provided on an "AS-IS" basis. Samsung will not be responsible for any damages arising out of the use of, or otherwise relating to, this document. Nothing in this document grants you any license or rights in or to information, materials, or contents provided in this document, or any other intellectual property.

The contents of this document may also include forward-looking statements. These forward-looking statements include all matters that are not historical facts, as well as statements regarding Samsung's intentions, beliefs and current expectations concerning, among other things, market prospects, growth, strategies, and the industry in which Samsung operates. By definition, forward-looking statements involve risks and uncertainties because they relate to events and depend on circumstances that may or may not occur in the future. Samsung hereby reminds you that forward-looking statements are not guarantees of future performance and that the actual developments of Samsung, the market, or the industry in which Samsung operates may differ materially from those made or suggested by the forward-looking statements contained in this document or in the accompanying oral statements. In addition, even if the information contained herein or the accompanying oral statements are shown to be accurate, those developments statements may not be indicative of future developments.

All contents in this document may be subject to change without notice. Without limiting the generality of the foregoing,

1. All design, features and specifications represented herein may change without notice;
2. Images shown here have been adjusted for demonstration purposes and may appear differently on the actual products;
3. All data on products herein, including their performances, are based on internal testing using standard Samsung benchmarks under laboratory conditions. Test results do not guarantee future performance under such test conditions, and the actual throughput or performance that any user will experience may vary depending upon many factors; and
4. All images on screen are simulated, except where otherwise noted.

BY CONTINUING TO ACCESS THIS DOCUMENT, YOU ARE DEEMED TO HAVE READ, UNDERSTOOD, AND AGREED WITH THE FOREGOING TERMS AND CONDITIONS.

Introduction

GPU-Initiated I/O and the Shift in Storage Architecture

Recent advances in artificial intelligence, large language models, vector databases, and large-scale data analytics have significantly increased the demand for high-performance storage systems. In many modern workloads, storage traffic is no longer dominated by large sequential I/O. Instead, these environments generate large numbers of small, random, and highly parallel I/O operations, making storage a growing performance constraint in GPU-accelerated systems.

In conventional system architectures, storage devices are attached to the CPU, and I/O requests must pass through the CPU, operating system kernel, and storage drivers before data can be consumed by the GPU. This CPU-centric path introduces additional latency, limits I/O parallelism, and consumes host compute resources for storage processing rather than application workloads. As GPU performance continues to scale and datasets increasingly exceed GPU memory capacity, data movement between storage and GPUs has become a critical factor in overall system efficiency.

To address this limitation, a new storage architecture model—commonly described as GPU-initiated I/O or GPU-driven storage—has emerged. In this model, GPUs participate more directly in initiating and managing storage access, reducing dependence on the CPU-centric I/O path. By shortening the software path and enabling more direct interaction between GPUs and NVMe® SSDs through the PCIe® fabrics, this approach can reduce latency, improves I/O parallelism, and allows storage performance to scale more effectively with compute and device count.

This shift represents a meaningful change in system design. Rather than being treated solely as a CPU-attached resource, storage increasingly becomes part of the GPU-accessible data path in accelerated computing systems.

SCADA Software Stack for GPU-Driven Storage

While GPU-initiated I/O architectures provide the hardware foundation for direct interaction between GPUs and storage devices, an appropriate software framework is required to enable efficient storage command generation and submission from the GPU side. SCADA (Scaled Accelerated Data Access) is a software framework intended for GPU-initiated storage operations, providing a programming model in which GPUs can issue storage I/O requests to NVMe SSDs without relying on the traditional CPU-centric I/O path.

SCADA enables GPU threads to generate and submit I/O commands directly to NVMe submission queues, allowing storage operations to be managed to the GPU rather than through the CPU and operating system kernel. This model reduces software overhead, minimizes CPU involvement in storage handling, and increases the level of parallelism available for storage-intensive workloads.

Feature	CPU-centric Workloads	SCADA Workloads
I/O Path	CPU → DRAM → SSD	GPU → SSD
Parallelism	Limited (CPU threads 10^2)	Massive (GPU threads ~10^5)
Max IOPS Issuance Capacity	Moderate (~45 M IOPS*)	Very High (~95 MIOPS)

Table 1. Comparison of CPU-Centric and SCADA Workloads

* CPU IOPS Derived from Microbenchmark Data1

By enabling GPU-driven storage operations, SCADA allows storage behavior to be evaluated in environments where performance scales with both GPU compute resources and attached SSDs. This approach is particularly relevant to modern workloads such as vector databases, graph analytics, large-scale inference pipelines, and data preprocessing flows, where large datasets must be moved efficiently between storage and GPU memory.

1 NVIDIA GTC 2026 Session S81840, Intel Xeon platform

In this paper, we evaluate the performance of a GPU-driven storage system using production-grade hardware and the SCADA software stack. We focus on system architecture, workload characteristics, and performance scaling, and the broader role of storage in modern data-intensive computing environments.

Evaluation Platform

System Configuration

To evaluate storage behavior and system-level performance under the SCADA software stack, we configured a controlled test platform based on the H3 Falcon 6048 system with a PCIe Gen6 switch fabric. The system was powered by an Intel 6th-generation Xeon CPU and equipped with one H100 GPU and two H200 GPUs, and multiple Samsung PM1763 NVMe SSDs connected through the PCIe fabric.

This configuration was designed to support direct GPU-initiated I/O to the SSDs to enable observation of peer-to-peer data transfer behavior between GPUs and storage devices. It was also intended to utilize high degree of I/O concurrency generated by GPU threads, making it suitable for analyzing fine-grained, highly parallel access patterns representative of SCADA workloads.

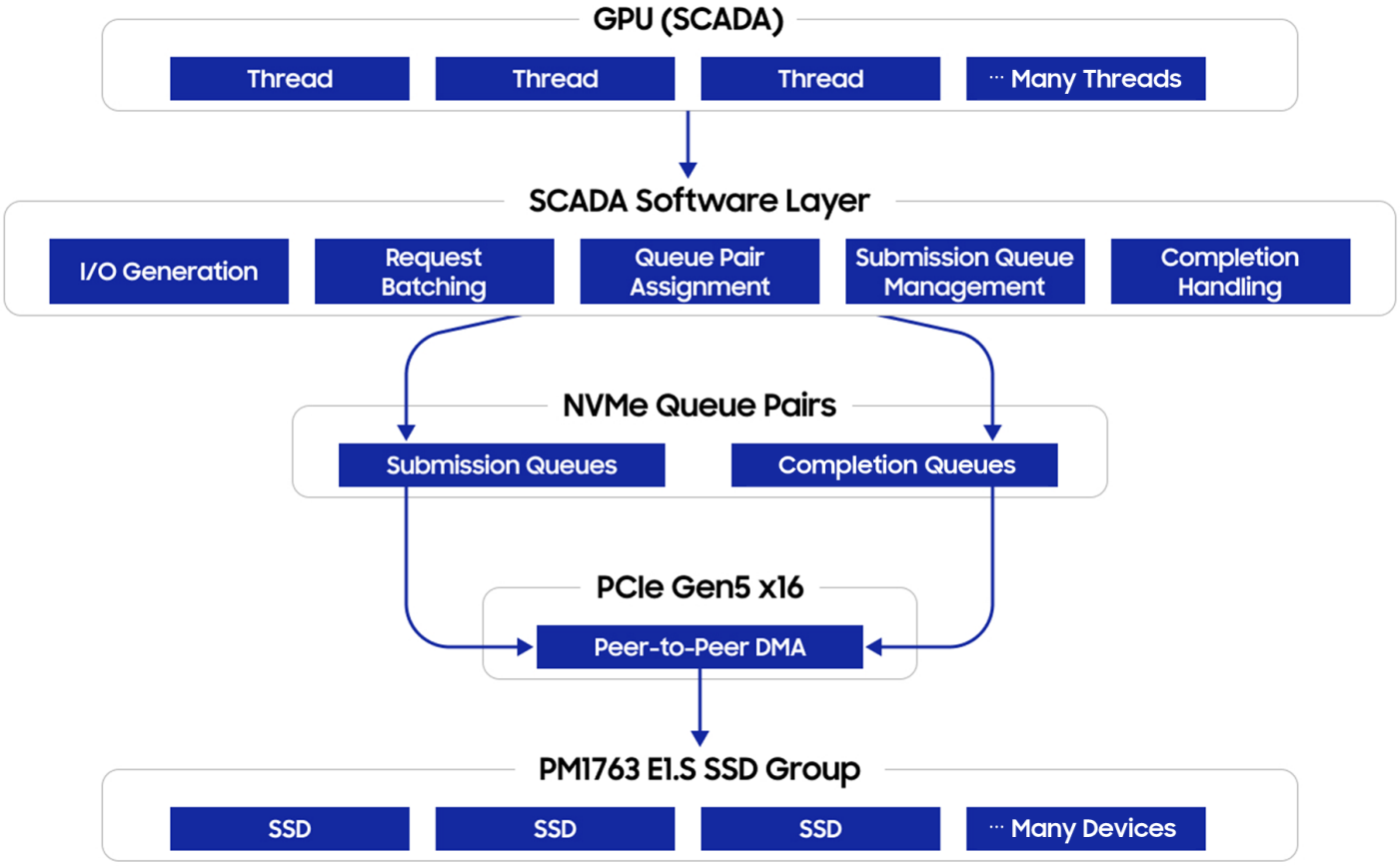

Figure 1. SCADA-Based GPU-Driven I/O Architecture

Samsung PM1763 NVMe SSD

The PM1763 is a PCIe Gen6 enterprise SSD designed to address the evolving storage requirements of modern AI and high-performance computing environments. With support for high-throughput data processing, improved power efficiency, and enhanced security capabilities, PM1763 SSD is well suited for deployment across hyperscale data center architectures.

In this study, PM1763 SSD served as a representative enterprise SSD platform for evaluating how storage behaves under sustained, high-concurrency GPU-initiated I/O. The test environment made it possible to observe latency behavior, throughput scaling, and SSD response consistency under realistic workload conditions.

Analyzing Benchmark Results

Workloads Description

The workload used in this study is based on the SOL (Speed-of-Light ExecBench) benchmark. This benchmark generates highly parallel, GPU-initiated I/O requests intended to stress the storage subsystem under extreme concurrency.

In this evaluation, a synthetic random-read workload with a 512-byte block size was used to represent fine-grained data access behavior. The number of SSDs was varied from 1 to 42 devices, and I/O requests were distributed across all active SSDs to enable balanced utilization of the storage subsystem. This setup was intended to measure small-block random-read performance and scaling behavior in a GPU-driven storage environment.

Workloads Characteristics

This workload is characterized by massive concurrency generated directly by GPU threads, with a large number of I/O requests issued simultaneously across multiple queue pairs. Unlike conventional CPU-driven access models, the workload stresses the storage subsystem through highly parallel GPU-originated command submission.

At the storage level, the access pattern is dominated by small, random reads distributed across localized regions of the SSD address space. This allows the benchmark to exercise SSD internal parallelism under sustained load while also exposing how effectively the storage subsystem handles distributed, interleaved access.

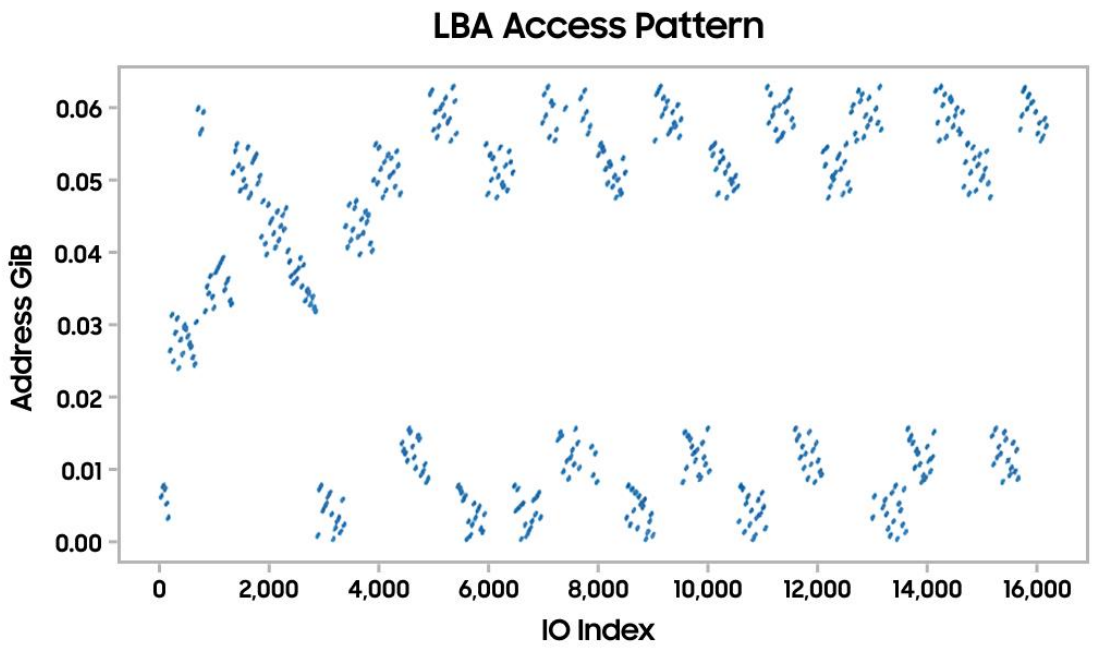

Figure 2. LBA Access Pattern During Benchmark Execution

Analysis of the logical block address (LBA) access pattern during benchmark execution shows that multiple GPU threads concurrently issue I/O requests to localized regions across different areas of the SSD address space. As a result, numerous small regions are accessed simultaneously rather than through a single contiguous range. In this environment, latency stability becomes an important factor alongside peak IOPS, because overall system throughput depends on how consistently multiple SSDs respond under concurrent GPU-driven access.

* LBA patterns were sampled from less than 0.1% of the total I/O.
* In this benchmark, the primary block size is 512B, and the graph above represents only a portion of the total access pattern.
* The pattern and frequency of LBA access may vary slightly depending on the iteration options used in the benchmark.

Performance Results

Under the SCADA benchmark, a single PM1763 SSD efficiently processed GPU-issued I/O requests and showed random-read performance slightly above the host-based baseline observed in the comparison setup.

- Under high-intensity SCADA-driven I/O conditions, the PM1763 SSD achieved approximately 6.92 million IOPS, representing an improvement of about 86% compared to the 3.72 million IOPS observed for the previous-generation, PM1753.
- To evaluate the maximum peer-to-peer DMA performance between the GPU and SSD, performance measurements were conducted while varying the Q-pair size.
- In this configuration, the highest single-SSD performance was observed at a Q-pair size of 8, reaching approximately 6.9 million IOPS.

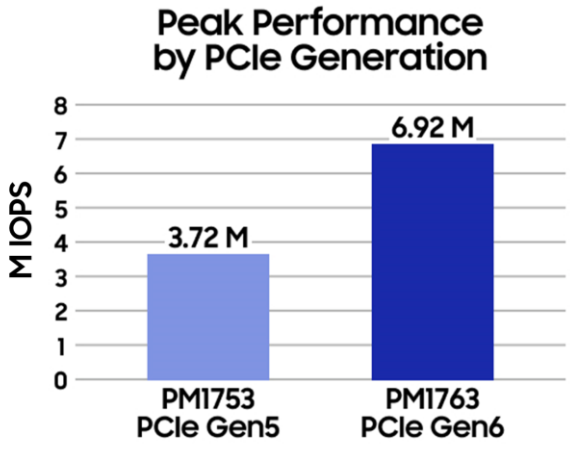

Figure 3. Peak Performance by PCIe Generation

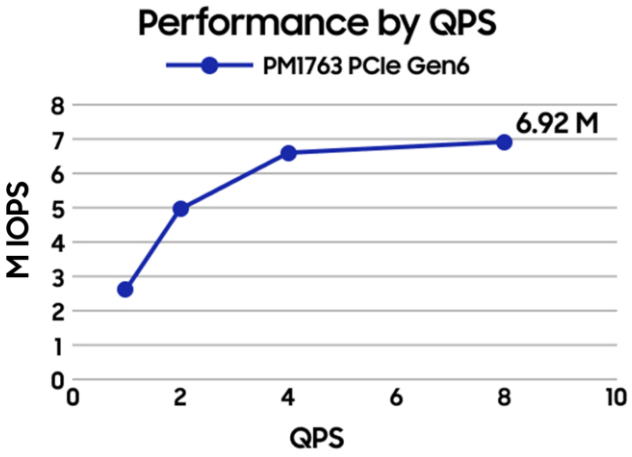

Figure 4. Performance by Queue-Pair Size

* When measuring peak performance, the highest IOPS were observed at queue pair size (QPS) 4 for PCIe Gen5 and QPS 8 for PCIe Gen6.
* To ensure accurate evaluation, device preconditioning was performed to fully populate the drive with data prior to the SCADA test.

Scaling Behavior

As the number of SSDs per GPU increased, aggregate system performance improved significantly relative to the single-SSD case. In this configuration, up to 14 PM1763 SSDs connected to a single Hopper-based GPU delivered aggregate performance approaching the practical throughput limit of the GPU-attached PCIe path. The PM1763 SSD also showed near-linear scaling behavior under the SCADA workload without introducing a visible storage bottleneck within the evaluated range.

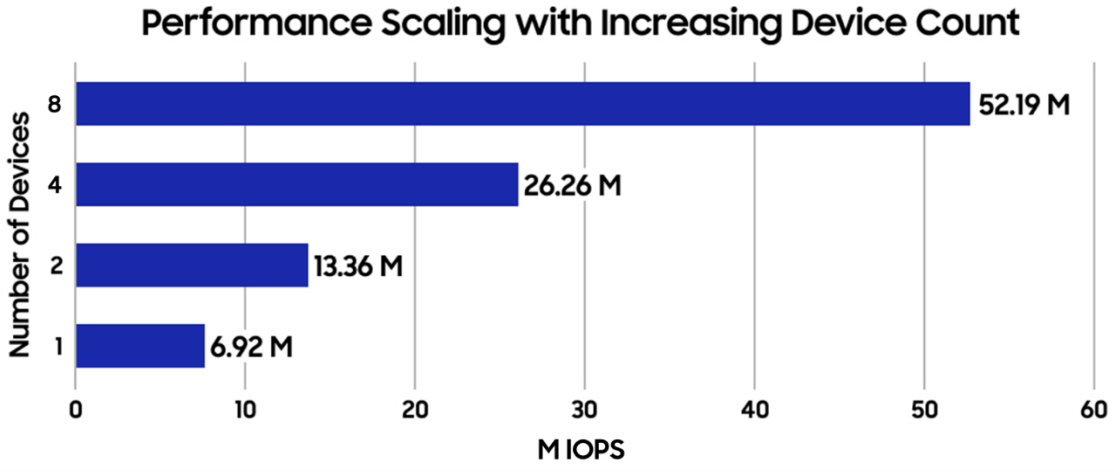

Figure 5. Performance Scaling with Increasing Device Count

When the configuration was extended to the maximum number of GPUs supported by the test server, a total of three GPUs, the system achieved up to 281 million IOPS in aggregate. This figure represents the combined throughput of all GPU–SSD groups operating concurrently and indicates that the SCADA-driven I/O model scales efficiently across multiple GPUs within the tested platform.

- With 14 SSDs attached per GPU, maximum performance of approximately 96 million IOPS was observed for each GPU and its associated SSD group.
- With 42 SSDs connected to 3 GPUs, the average IOPS per SSD was approximately 6.7 million, showing a deviation of less than 5% from the peak single-SSD result.
- Variation in total performance across GPUs was primarily attributable to differences among individual SSDs within each 14-device group. However, even across repeated runs, the observed variation remained within a limited range, and cross-GPU total performance differences stayed within 5%.

Q Pair Size	GPU	IOPS (M IOPS)	Bandwidth (GB/s)
8	0000:07:00.0	96.01	45.78
	0000:44:00.0	94.07	44.86
	0000:0A:00.0	91.26	43.52
Total Performance		281.35	134.16

Table 2. SCADA Performance with 3 GPUs and 14 PM1763 SSDs per GPU

* Test condition: Request=512 Thread/1024 QueueSize, QueuePairSize=8, Block Size=512B.
* In GPU and multi-SSD evaluations, performance variation was observed across test cycles, with the magnitude of variation remaining below 1%.
* In this experiment, the peak effective bandwidth observed with SCADA was approximately 50 GB/s, corresponding to about 97 MIOPS for 512-byte transfers.

Although total system performance continued to increase as more SSDs were added, the incremental gain per additional SSD became smaller beyond a certain device count. This suggests that there is an efficiency range in which per-SSD contribution is maximized, even when aggregate throughput continues to scale. In multi-SSD operation, latency stability across devices also proved to be an important factor. Higher overall performance was observed when all SSDs maintained similarly low latency, rather than when a subset of devices exhibited localized latency excursions. These observations indicate that large-scale GPU-driven storage performance depends not only on peak IOPS, but also on balanced I/O distribution and latency consistency across SSDs.

Industry Implications

As GPU-driven storage architectures continue to evolve, storage is becoming an active part of the data path rather than a passive backend resource. The benchmark results in this study suggest that future enterprise SSDs for AI and high-performance computing environments should be evaluated not only for peak throughput, but also for latency stability, scaling efficiency, and their ability to sustain highly parallel, interleaved I/O patterns under GPU-driven access models.

These observations also imply that storage architecture is increasingly becoming a system-level design consideration rather than a standalone device choice. As data movement becomes a larger determinant of end-to-end system efficiency, storage must be assessed in the context of the full GPU-accelerated platform, including concurrency behavior, interconnect utilization, and consistency under load.

Conclusion

In this study, we evaluated the performance of a GPU-driven storage architecture using SCADA software stack and the PM1763 SSDs under highly parallel I/O workloads. The results show that storage performance scales efficiently with increasing device count, enabling high-throughput and low-latency data access under extreme concurrency.

These findings highlight the potential of GPU-initiated storage I/O as an important enabler for future data-intensive computing systems, particularly as GPU HBM and host DRAM capacity remain limited relative to rapidly growing datasets. As more workloads extend beyond in-memory capacity, storage is expected to play a more active role in the system data path.

In this context, SCADA-based storage architectures can function as a scalable, high-performance data tier within the GPU-accessible data path. The PM1763 SSD, as evaluated in this study, demonstrated latency and throughput characteristics suitable for such environments. Overall, the results suggest that GPU-driven storage, combined with high-performance enterprise SSDs, can contribute to the evolution of system architectures in which storage is treated not as a passive repository, but as an active component of data movement for GPU-accelerated computing.

Technical specifications

SCADA Evaluation Platform with PM1763 NVMe SSDs	
Server platform	H3 Falcon 6048 (PCIe Gen6 Server)
CPU	Intel 6th Xeon CPU
GPU	1× NVIDIA H100 and 2× NVIDIA H200
Switch	3x Broadcom PEX90144 PCIe Gen6 series
Storage	42× Samsung PM1763 PCIe Gen6 SSDs (E1.S 15.36 TB)
OS	Ubuntu 24.04 LTS

Table 3. SCADA Evaluation Platform with PM1763 NVMe SSDs

* All results are based on SCADA v0.3 software. The associated drivers and benchmark components were developed by NVIDIA.
* The results presented in this paper reflect the specific software stack and hardware configuration used in this experiment and should not be interpreted as the maximum achievable performance. Performance may vary with different system configurations and software versions.

Samsung Electronics Co., Ltd.
1-1, Samsungjeonja-ro, Hwaseong-si, Gyeonggi-do 18448, Korea www.samsung.com 1995-2026