SSD Profiling

October 02, 2025

Joyal Mathew

Experiments

Experiments were done on a test partition using the Flexible I/O Tester (FIO).

Zero-Queue Baselines

FIO was used to generate these measurements. The following options were set to measure the zero-queue latency:

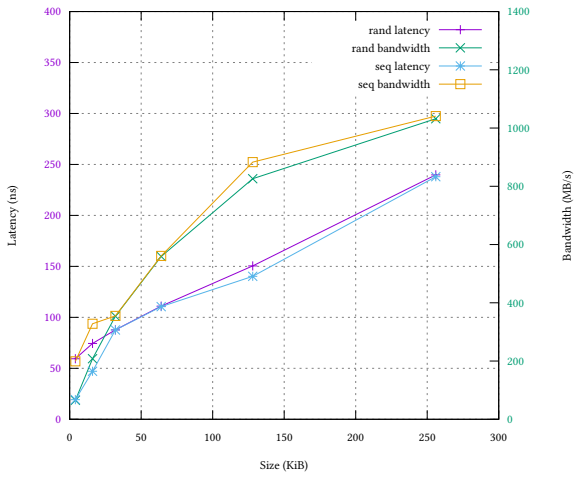
```
iodepth=1
direct=1
```

These options ensured no queuing and direct I/O access (bypassing the OS cache).

	Read Latency (ns)	Write Latency (ns)
Random Access, 4 KiB Blocks	62.11	19.8
Sequential Access, 128 KiB Blocks	704.89	106.22

Block-Size/Pattern Sweep

Here we sweep block size from 4 KiB to 256 KiB with access pattern constant, measuring latency and bandwidth.

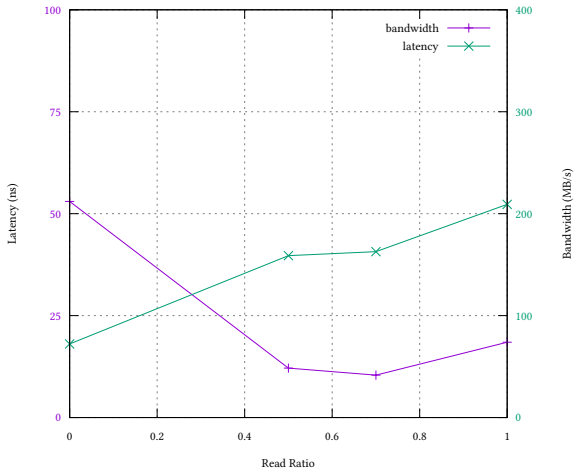


Block Size (KiB)	Rand BW (MB/s)	Rand Latency (µs)	Seq BW (MB/s)	Seq Latency (µs)
4	65.2	59.39	199	19.27
16	208	74.39	328	47.10
32	354	87.72	355	87.62
64	559	111.01	561	110.51
128	826	150.47	883	140.25
256	1032	239.79	1041	237.97

For larger block-sizes, fewer requests are made to the I/O controller for the same amount of data. This allows for larger bandwidth and controller overhead decreases. At this same time, it takes longer to process each individual request so latency increases.

R/W Ratio Sweep

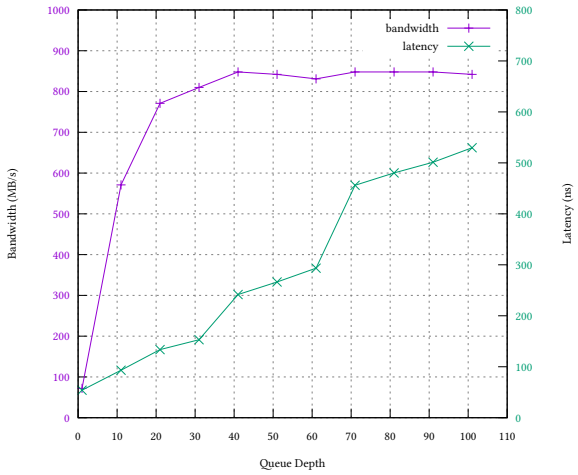
In this experiment, we varied the R/W ratio.



Bandwidth increases as more reads are done. This can be explained by reads being faster on SSDs than writes. Reading simply gets data off of the flash memory while writes require program/erase cycles. Latency for mixed operation is lower due to additional overhead in the controller in dispatching different operations to the SSD.

Queue Depth Sweep

Here we sweep the queue depth from 1 to 5.



Bandwidth increases directly with queue depth. This expected because when more I/O operations can be queued, the I/O bandwidth can be better saturated. Once the device bandwidth is saturated however, the achieved bandwidth levels off and increased queue depth does nothing.

Latency is observed to increase in somewhat discrete jumps. This can be explained by requests overflowing the queue and needing to be done in seperate batches. When an additional request overflows to the next batch, latency spikes up.