Adv. Computer Systems - Project 3

Running Instructions

To run this project, first create a 2GB partition (formatted as ext4) on your SSD for testing. Make note of the name of this partition (in my case, it was /dev/nvme0n1p6). Next install the Flexible IO Tester (FIO) and download the ssd_test.c file from the repository. Within this file, you should change the "--filename=/dev/nvme0n1p6 argument on line 11 to match your partition name. Now you can compile with gcc -o ssd_test ssd_test.c and run it (while capturing the output data) via ./ssd_test > output_log.txt. (This txt file is included in the repository for viewing).

This program executes 64 FIO test commands, testing all combinations of block_sizes [4k, 16k, 32k, 128k], R/W ratios [100, 0, 50, 70]. The I/O depths [1, 32, 256, 1024]. The output log can be parsed to find the IOPS, bandwidth, and latency of each test. The resulting data and generated graphs can be seen in the google sheet at: https://docs.google.com/spreadsheets/d/1Igxt_eIk4_H00SV9GrEZ2uW2mJoXc-iMJnGU6dBBGMc/edit?usp=sharing

Formal analysis of these graphs can be seen in the PDF

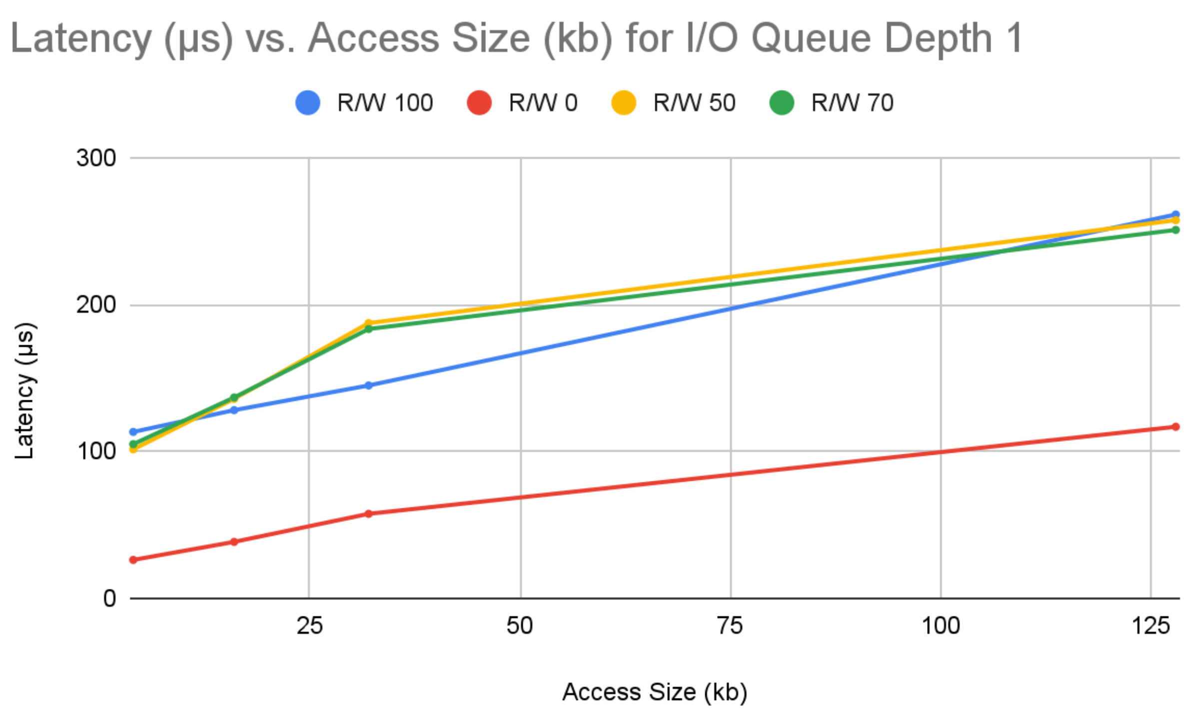
Note: The SSD used for these tests was a WDC PC SN730 SDBQNTY-512-1001.

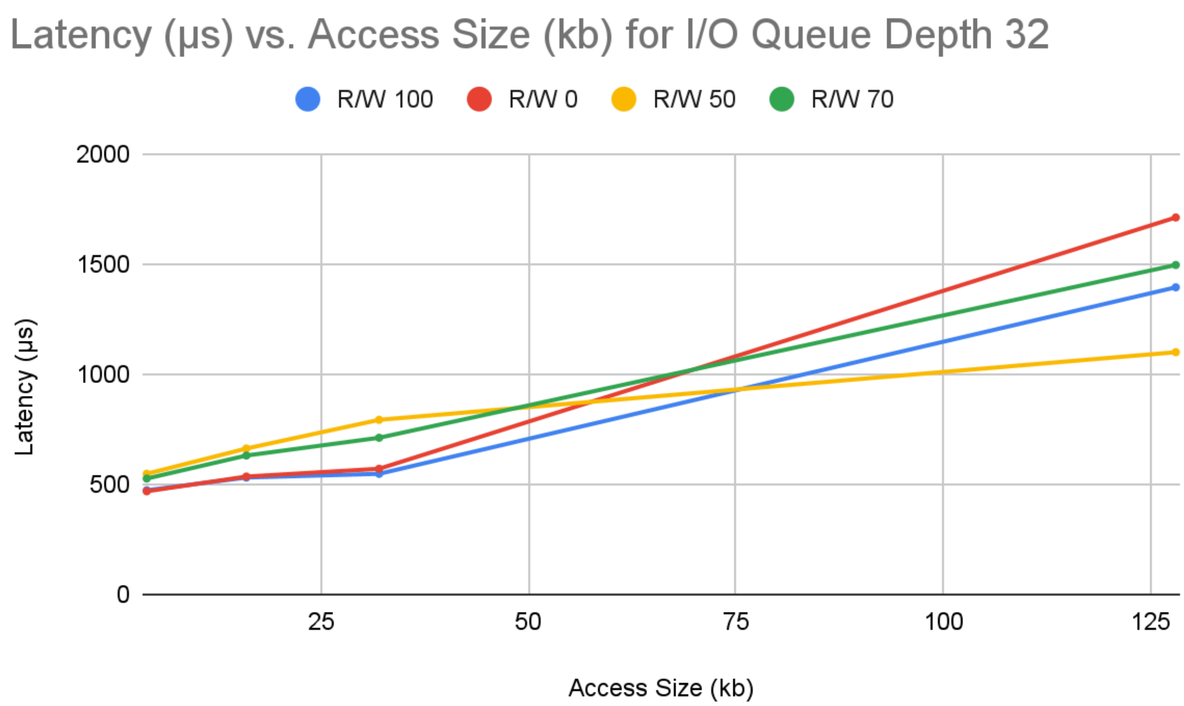
Grading Criteria

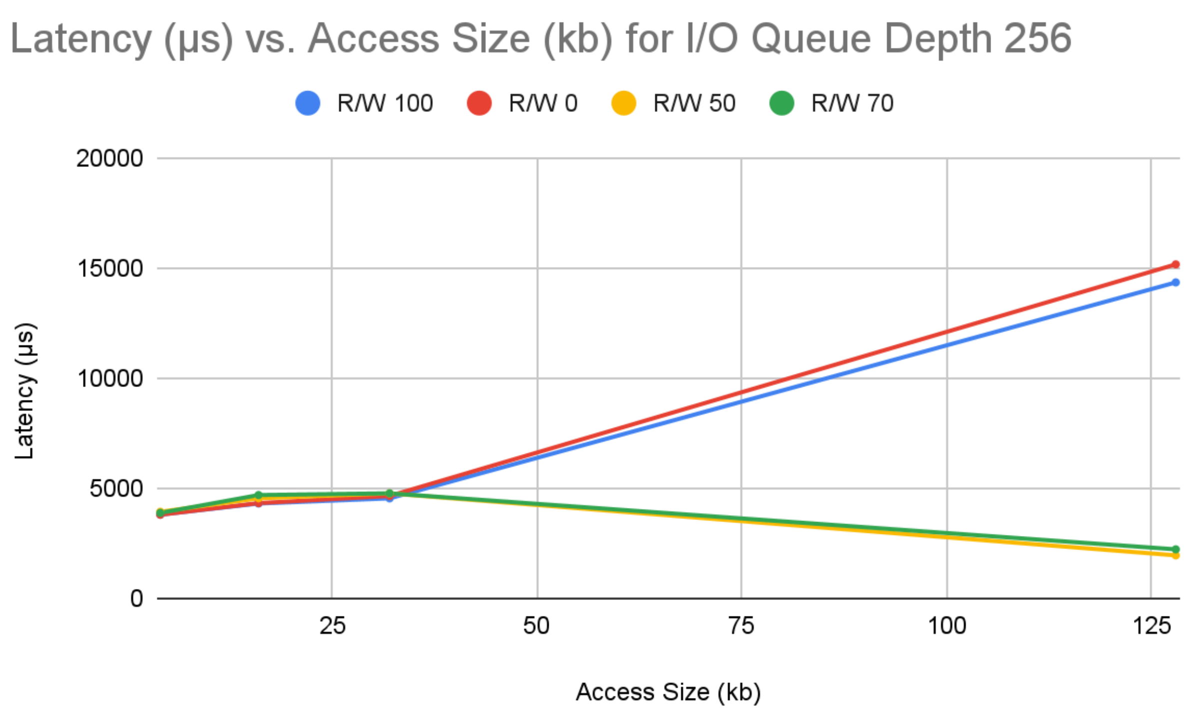
Latency Measurements

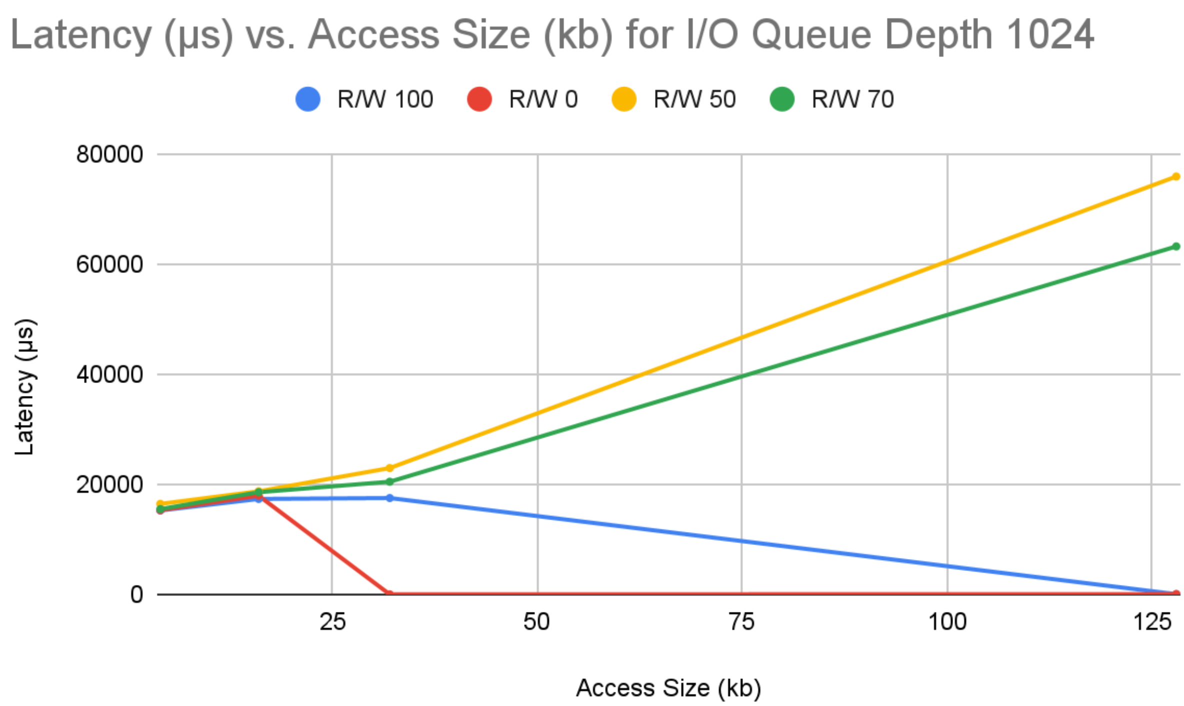
The latency measurement tests involved measuring the latency (μs) while changing the access size (4k, 16k, 32k, 128k) for all four R/W ratios (100, 0, 50, 70). The I/O Depth is held constant for all 4 graphs. To summarize the effect of I/O Depth, the latency was plotted with a fixed access size of 128kb as well for a fifth graph. All 5 graphs can be seen below.

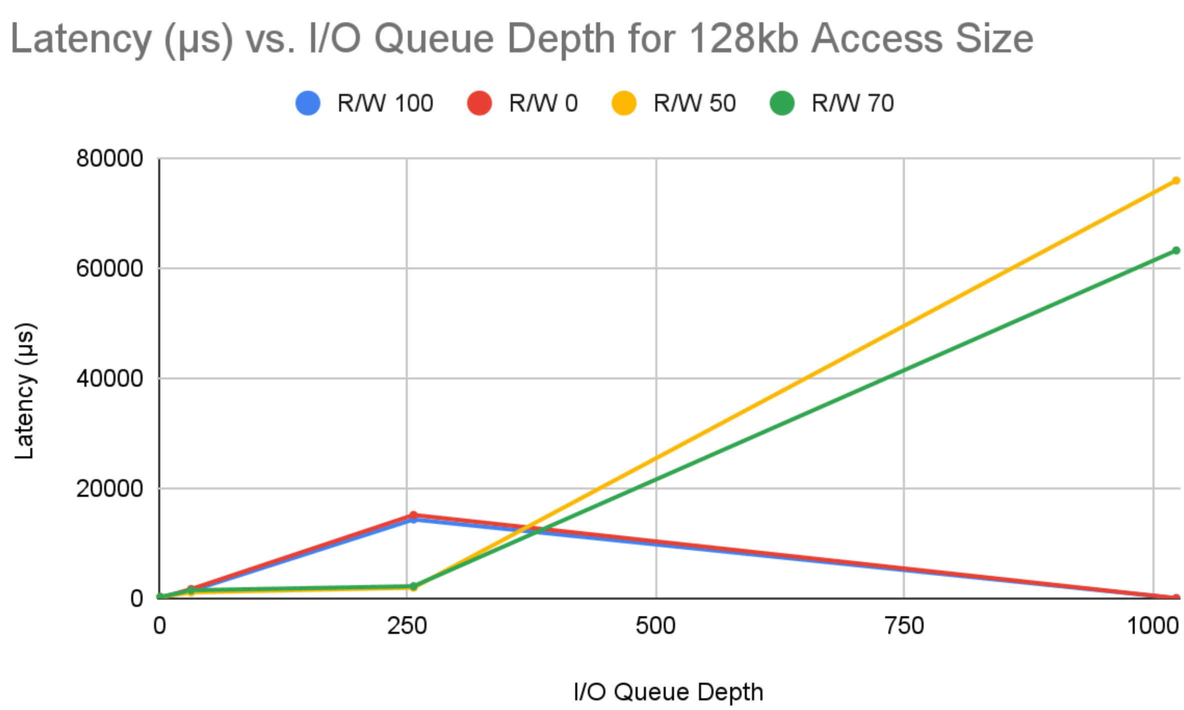
The trend from the I/O Depth 1 chart is that as access size increases, latency generally increases. This is likely due to queueing and processing overhead adding delays. It is worth noting that the read only/write only tests actually see latency decreases at higher access sizes as the I/O Depth increases. This could be due to the SSD reaching higher levels of optimization and workload distribution, or that writes and interfering with reads in the mixed-request test. I/O Depth 32 is the last case where we see reads have a lower latency than writes. This discrepancy is due to reads being simpler and easier to process than write requests (extra overhead and challenges with writing). Hardware resources are being more effectively used to process these read/write only requests. Initially, the fastest R/W ratio was the write only. However, for I/O depth 32 and 128, the ordering curiously swapped. The ordering returned to the initial state at I/O depth 1024. This shift is better captured in the fifth graph (latency vs. I/O depth for 128kb access size).




Latency (µs) vs. Access Size (kb) for I/O Queue Depth 32
R/W 100
R/W 0
R/W 50
R/W 70
2000
1500
1000
500
0
Latency (µs)
25
50
75
100
125
Access Size (kb)


Latency (µs) vs. Access Size (kb) for I/O Queue Depth 256
R/W 100
R/W 0
R/W 50
R/W 70
20000
15000
10000
5000
0
Latency (µs)
25
50
75
100
125
Access Size (kb)

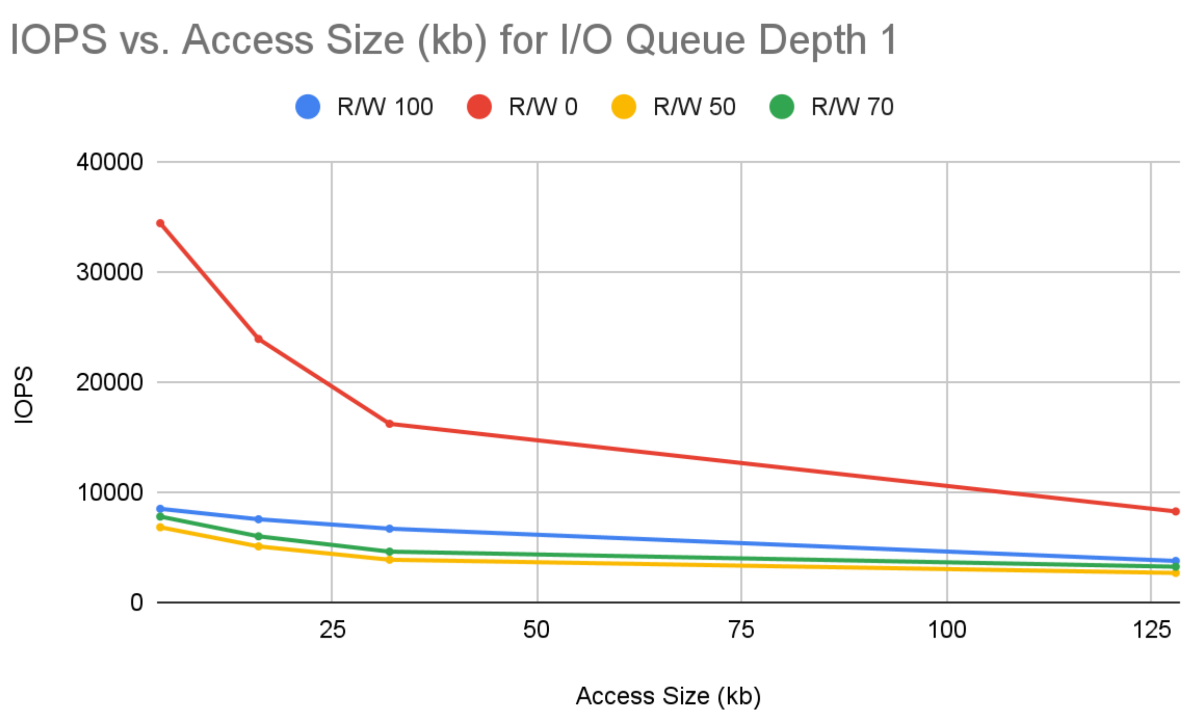

Latency (μs) vs. Access Size (kb) for I/O Queue Depth 1024
R/W 100
R/W 0
R/W 50
R/W 70
80000
60000
40000
20000
0
25
50
75
100
125
Latency (μs)
Access Size (kb)

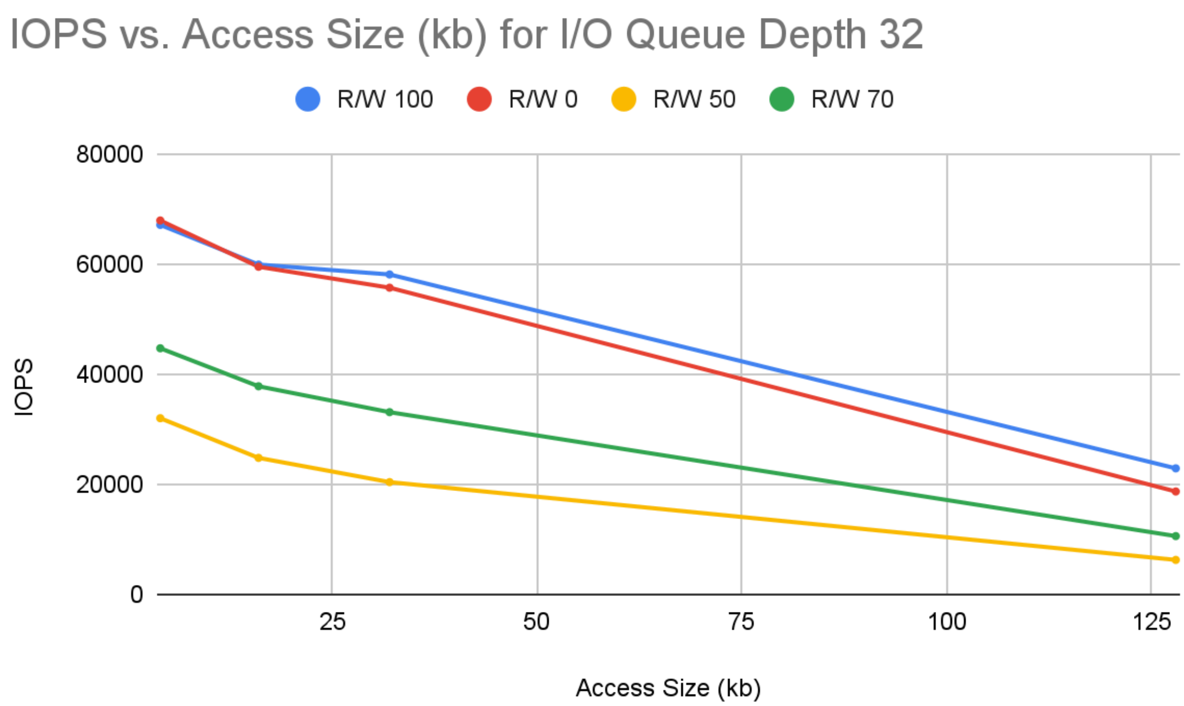

Latency (μs) vs. I/O Queue Depth for 128kb Access Size
R/W 100
R/W 0
R/W 50
R/W 70
80000
60000
40000
20000
0
0
250
500
750
1000
Latency (μs)
I/O Queue Depth

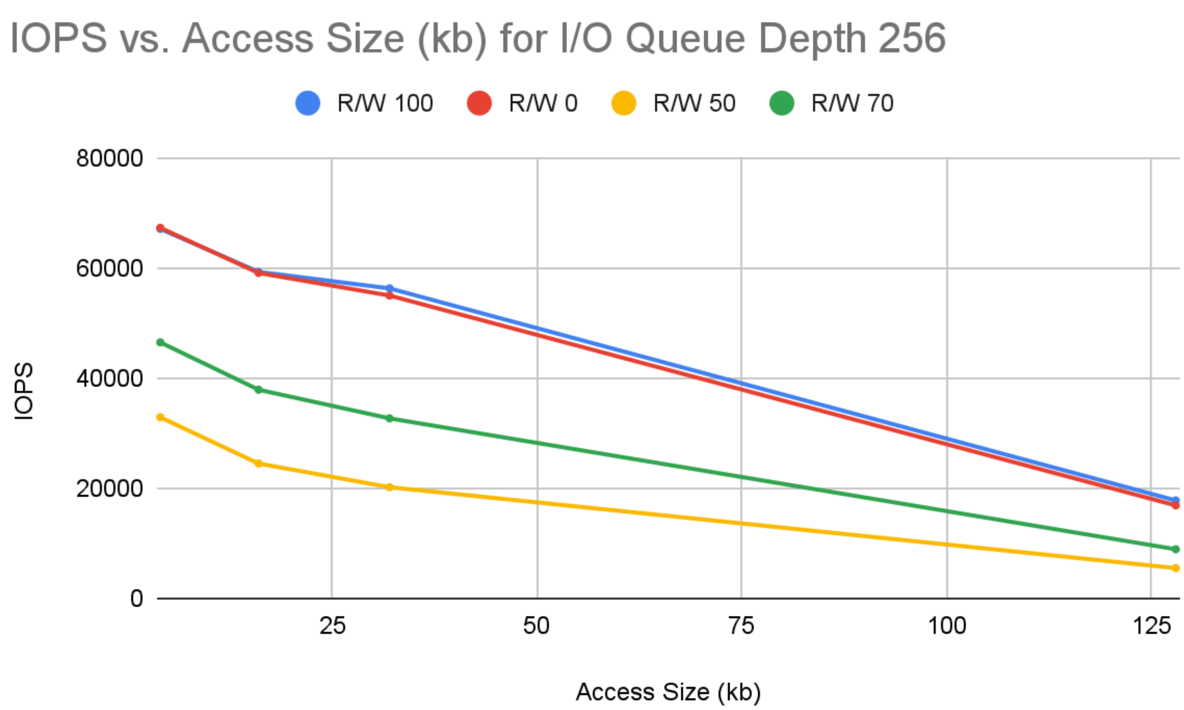
Bandwidth Measurements

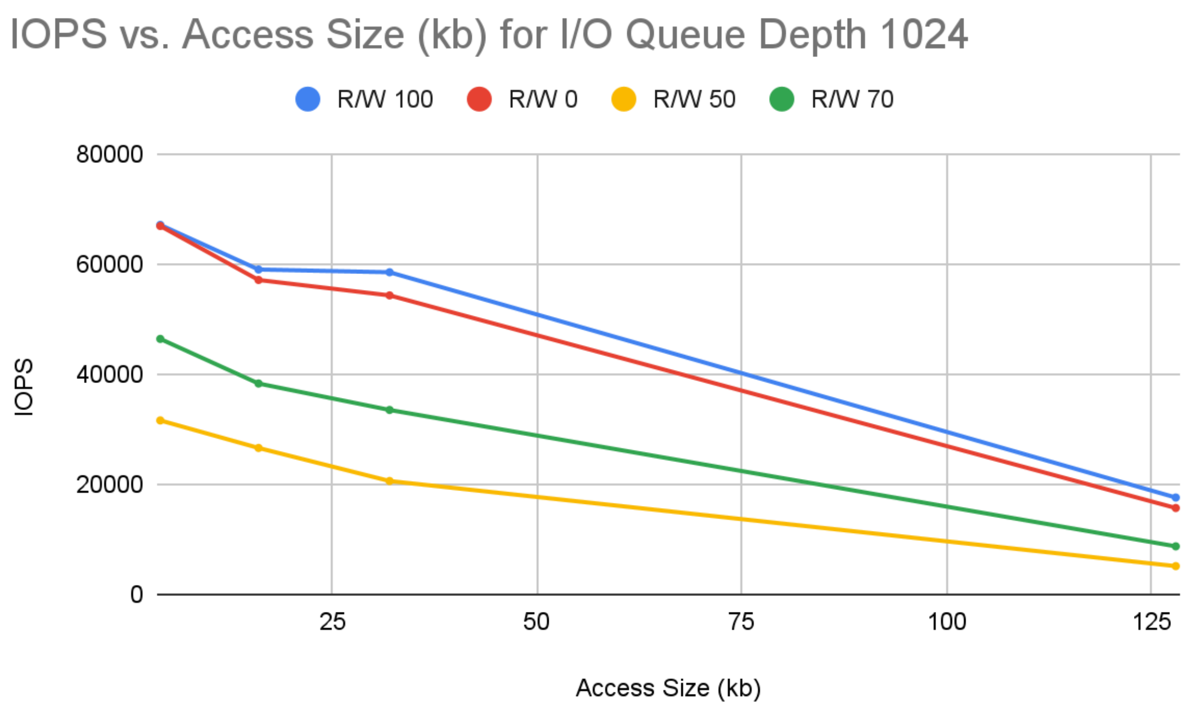
The bandwidth (focused on IOPS) measurement tests involved measuring IOPS while changing the access size (4k, 16k, 32k, 128k) for all four R/W ratios (100, 0, 50, 70). The I/O Depth is held constant for all 4 graphs. To summarize the effect of I/O depth, the IOPS and Bandwidth were plotted with a fixed access size of 128kb as well for a fifth and sixth graph. All 6 graphs can be seen below.

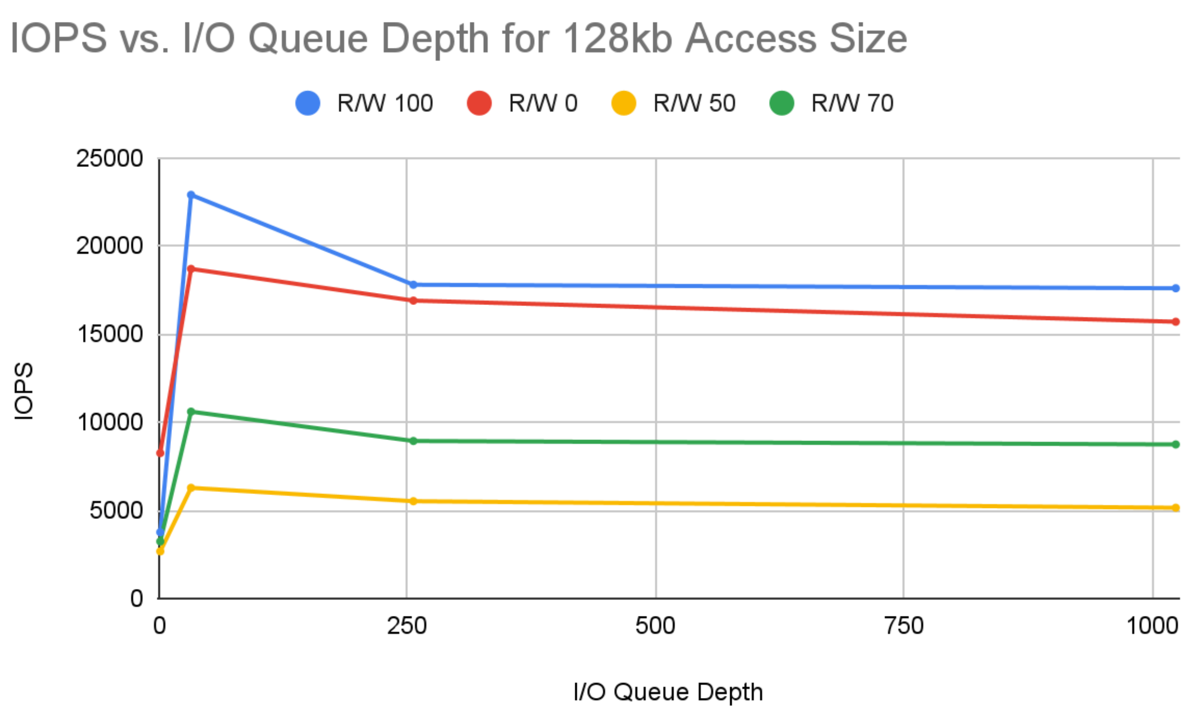
The trend from all of these graphs shows that as access size increases, IOPS will generally decrease for all four I/O depth tests. This is due to the SSD not being able to handle as many operations per second as the access size increases. This is because it will take longer for a single operation to occur if the access size is larger (inversely related, IOPS = 1/Avg. Latency per Operation). It is worth noting that 100% Read generally had the highest IOPS, followed by 100% write, 70% read and 50% read. Reads will typically have a higher IOPS than write does due to reads being generally simpler and faster (only retrieving data from NAND flash cells). Writes are far more complex due to additional processing such as garbage collection, wear leveling, write amplification, and caching. Endurance and longevity optimizations can also limit IOPS for write to minimize wear and tear on NAND cells. Blending reads and writes leads to further slow downs due to switching between processes, rather than a streamlined list of processes that deal with the same function. The final graph, which shows a bandwidth test, shows that Bandwidth will initially increase with increased I/O depth, but then level off. This could be due to the SSD reaching internal processing limits or diminishing returns.

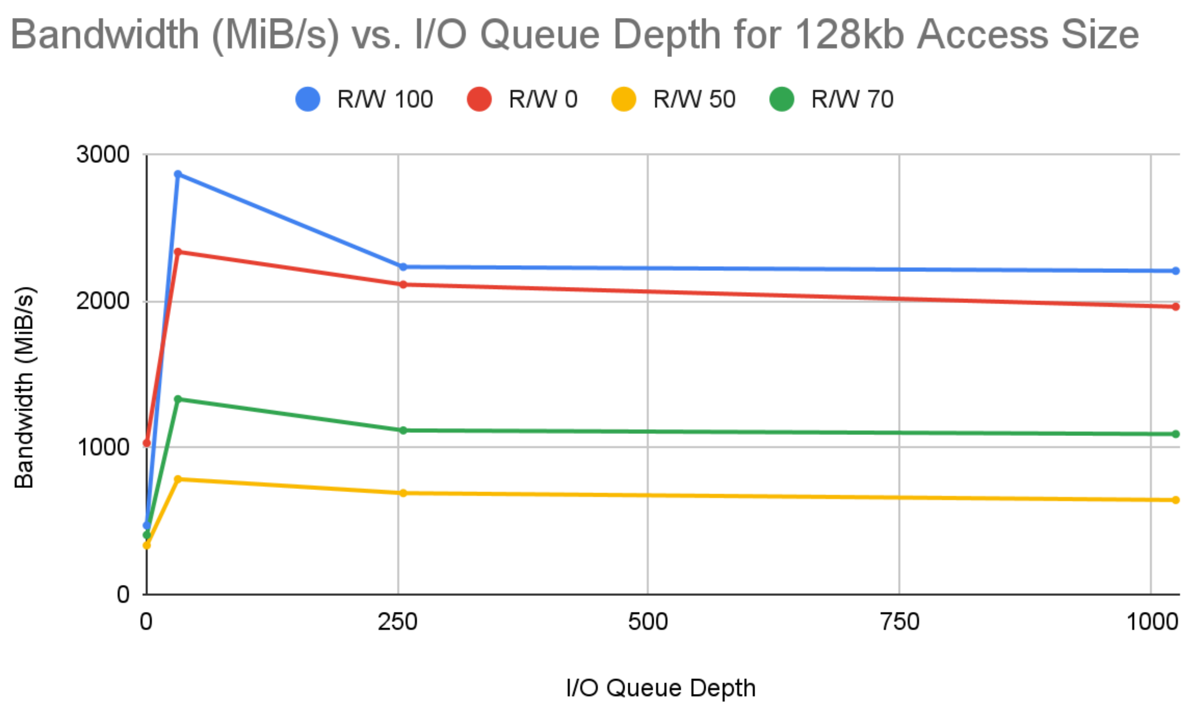



IOPS vs. Access Size (kb) for I/O Queue Depth 32
R/W 100
R/W 0
R/W 50
R/W 70
80000
60000
40000
20000
0
IOPS
25
50
75
100
125
Access Size (kb)


IOPS vs. Access Size (kb) for I/O Queue Depth 256
R/W 100
R/W 0
R/W 50
R/W 70
80000
60000
40000
20000
0
IOPS
25
50
75
100
125
Access Size (kb)


IOPS vs. Access Size (kb) for I/O Queue Depth 1024
R/W 100
R/W 0
R/W 50
R/W 70
80000
60000
40000
20000
0
IOPS
25
50
75
100
125
Access Size (kb)


IOPS vs. I/O Queue Depth for 128kb Access Size
R/W 100
R/W 0
R/W 50
R/W 70
25000
20000
15000
10000
5000
0
IOPS
0
250
500
750
1000
I/O Queue Depth



Conclusion & Analysis

By using FIO, SSD performance could be experimentally observed and several points could be drawn from the performance metrics gathered with varying I/O Depth, R/W Ratios, and Access Size. As I/O Depth increases, IOPS increases and levels out. Meanwhile,latency increases for mixed read/writes and initially rises then drops for read-only/write-only. With increased access size, latency typically increases while IOPS will decrease. Read-centric tests should have higher IOPS and lower latency due to reads being easier to execute than writes.

The project handout stated that an Intel enterprise-grade SSD lists a random write-only 4KB IOPS of 130k. My ssd could only achieve around 67k IOPS in similar conditions. This could be due to CPU limitations, linux I/O scheduling, laptop power settings, or thermal throttling. Generally, you would see modern consumer SSDs to have higher IOPS than enterprise SSDs in this field. This is likely due to enterprise SSDs having extra operational overhead from data correction tools and redundancy methodologies. This would degrade raw performance on enterprise SSDs, but greatly increase reliability and data assurance/quality.