FLASH Disk Opportunity for Server-Applications

Jim Gray, Microsoft Research, Gray@microsoft.com, Bob Fitzgerald, BobFitz@microsoft.com
with help from Aaron Dietrich, Dennis Fettterly, James Hamilton, Chuck Thacker,
8 December 2006
Revised 18 Jan 2007

Executive summary: Future flash-based disks could provide breakthroughs in IOps, power, reliability, and volumetric capacity when compared to conventional disks.

Introduction

NAND FLASH densities have been doubling each year since 1996. Samsung announced that they their 32 gigabit NAND FLASH chips will be available in 2007. This is consistent with Hwang's FLASH memory growth model that NAND FLASH densities will double each year until 2010. They recently extended that 2003 prediction to 2012 suggesting 64x current density -- 250 giga-bytes per chip. This is hard to credit, but Hwang and Samsung have delivered 16x since his 2003 article when 2Gb chips were just emerging. So, we should be prepared for the day when a FLASH drive is a terabyte (!). As Hwang points out in his article – mobile and consumer applications, rather than the PC ecosystem, are pushing this technology.

Several of these chips one can be packaged as a disk replacement. Samsung has 32 GB FLASH disk (NSSD – NAND Solid State Disk) that is PATA now and SATA soon. It comes in standard 1.8" and 2.5" disk form factors that plug into a PC as a standard small-form factor disk. Several other vendors offer similar products. Ritek has announced a 16GB flash disk for 170$ and a 32GB disk later [Ritek], and SanDisk bought M-Systems which has long made FLASH disks for the military has a 32 GB disk for ~1,000$ [SanDisk].

These "disks" are expensive (list price for one is $1,800 on the web[1] and the $170 is not yet available.) But, they consume about 15x less power (0.9 watts vs. 14 watts) and they (potentially) deliver ~10x more accesses per second (2.5k iops vs 200 iops) than high-performance SCSI disks. They are also highly shock resistant (>1kG). Through good engineering they have circumvented two FLASH shortcomings: (1) a particular byte of FLASH can only be written a million times, and (2) one cannot write zeros to a page, so one must erase (reset) the page to all 1's and then write the ones – this makes writes slow. The system diagram in the NSSD article hints at some of the ways Samsung does this (see Figure 1).

Tom's Hardware did an excellent review of the Samsung product. Two articles by Microsoft research measure several USB Flash drives ["FlashDB: Dynamic Self-tuning Database for NAND Flash," Suman Nath, Aman Kansal, MSR-TR-2006-168 and "A Design for High-Performance Flash Disks," Andrew Birrell, Michael Isard, Chuck Thacker, Ted Wobber, MSR-TR-2005-176] but I wanted to do some tests with our tools (SqlIo.exe, and DiskSpd.exe). Aaron Dietrich gave me access to a 32-bit mode Windows Vista RC2 Build 5744 on a dual-core 3.2 Ghz Intel x86 with 1GB RAM with a beta NAND Flash 32 GB disk and Dennis Fettterly give me access to a 4GB M-Systems UFD Ultra USB device.

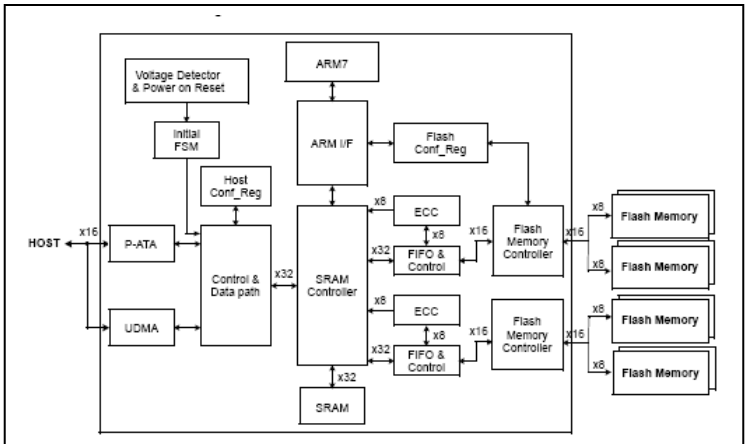


Figure 1: The block diagram for the Samsung flash disk from its

[1] http://www.dvnation.com/nand-flash-ssd.html has a PQI FLASH disk for $1800. The marginal cost is about 20$/GB for FLASH today so this device might be had for ~$600, comparable to the price of 15k RPM SCSI disk.

The Tests

I tested the sequential and random performance both read and write using both SqlIO.exe and the public-domain DiskSpd.exe. Both gave comparable results – so I report the DiskSpd.exe numbers here because you can see the code and perhaps change it.

The sequential tests run for a minute, use either read or write operations on a variety of block sizes (from 512 bytes to 1 megabyte) and either 1, 2, or 4 outstanding IOs – that is, DiskSpd issues *N* (= 1, 2, 4) IOs and then when one completes we issue one more till we are done. In the sequential test, each subsequent IO goes to the next block in the 1GB file generated by: `genfile -r128 -s- -b4096 80M test.dat`.

The random IO tests follow the same pattern except that they issue each IO to a randomly chosen block aligned location in the file. The M-system's performance was not as good as the other product, so I report the better numbers here.

The Bottom Line (*once write-cache was enabled – read on if you care about that detail*).

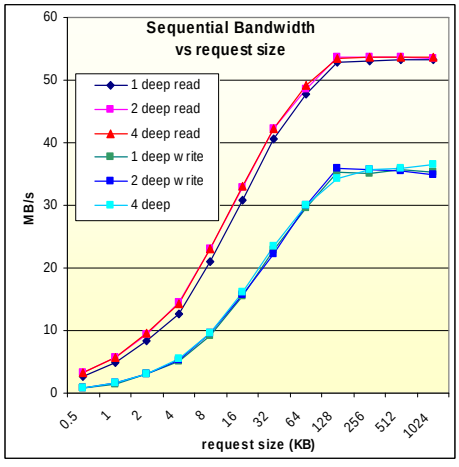
Table 1: Measurements of a current flash disk's performance.						
Device	**Sequential**	**Random 8KB**	**Price $**	**Power**	**iops/$**	**iops/watt**
SCSI 15k rpm	75 MBps	200 iops	500$	15 watt	0.5	13
SATA 10k rpm	60 MBps	100 iops	150$	8 watt	0.7	12
Flash- read	53 MBps	2,800 iops	?? 400$	0.9 watt	7.0	3,100
Flash - write	36 MBps	27 iops	?? 400$	0.9 watt	0.07	30

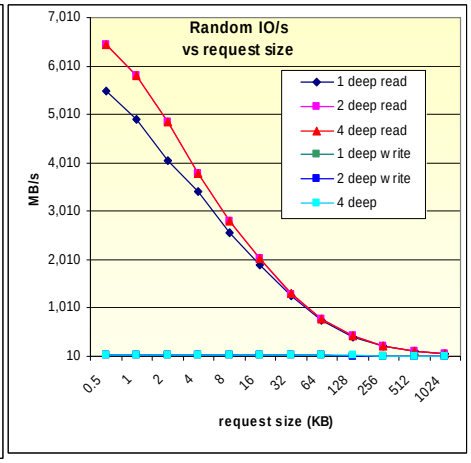
Measurements with Write Cache Disabled – Random Writes are Problematic

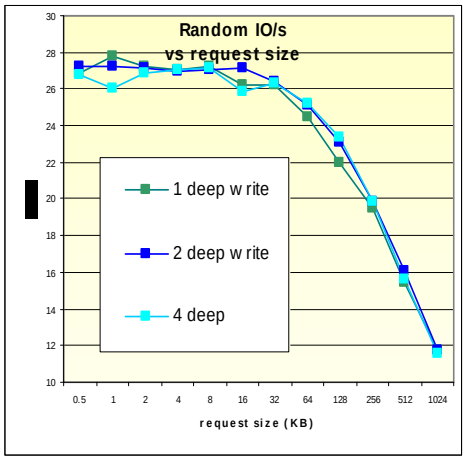
The graphs tell the story. The device sequential read-write performance is very good. At 4-deep 512B requests, the device is servicing 6,528 read requests per second (!) or 1,644 writes per second. Read performance is significantly better than write performance. Sequential throughput plateaus beyond 128KB requests giving 53MBps for reads and 35MBps for writes.

The story for random IOs is more complex – and disappointing. For the typical 4-deep 8KB random request, read performance is a spectacular 2,800 requests/second but write performance is a disappointing 27 requests/second**.**

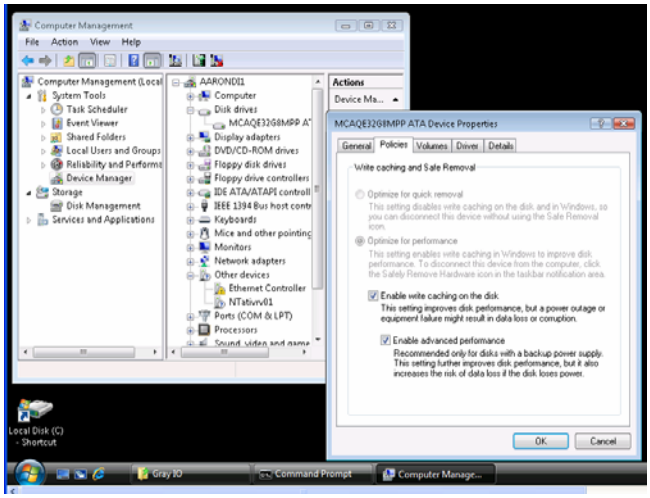
Clearly there is something wrong with the random write performance. As the next section shows, Windows is doing synchronous writes and the device has fairly long latency (30ms) for writes.







Enabling "Advanced Performance"



By default, in Windows Vista the disk must write data to non-volatile store before acknowledging the write completion. Some systems have a battery-backed disk cache and so can respond sooner and some foolish people just enable this cache – for example it has been enabled on my laptop for the last 5 years without ill effect (so far).

To take this risk, go to Start→Computer→ Manage→ DeviceManager→ DiskDrives, right click the disk you want to risk. Select "properties" from the menu and then select the policy tab and select "Enable advanced performance" after you read the warning label. This page is a bit confusing: it says you cannot modify the WCE performance and then it gives you two radio buttons that do.

When I selected "advanced write performance" I expected to see great performance but not so. I could never get good write performance. This could be a problem with the device driver or with the device. I believe that the article "A Design for High-Performance Flash Disks," Andrew Birrell, Michael Isard, Chuck Thacker, Ted Wobber, [MSR-TR-2005-176.] explains both the problem and a solution. Clearly, a little intelligence in the disk controller could buffer the writes and give performance comparable to the 1,100 IO/s that we get with sequential writes – but that is not what I see with the current software-hardware configuration.

What If FLASH Disks Delivered Thousands of IO/s and Were "Big"?

My tests and those of several others suggest that FLASH disks can deliver about 3K random 8KB reads/second and with some re-engineering about 1,100 random 8KB writes per second. Indeed, it appears that a single FLASH chip could deliver nearly that performance and there are many chips inside the "box" – so the actual limit could be 4x or more. But, even the current performance would be VERY attractive for many enterprise applications. For example, in the TPC-C benchmark, has approximately equal reads and writes. Using the graphs above, and doing a weighted average of the 4-deep 8 KB random read rate (2,804 IOps), and 4-deep 8 KB sequential write rate (1233 IOps) gives *harmonic average* of 1713 (1-deep gives 1,624 IOps). TPC-C systems are configured with ~50 disks per cpu. For example the most recent Dell TPC-C system has ninety 15Krpm 36GB SCSI disks costing 45k$ (with 10k$ extra for maintenance that gets "discounted"). Those disks are 68% of the system cost. They deliver about 18,000 IO/s. That is comparable to the requests/second of ten FLASH disks. So we could replace those 90 disks with ten NSSD if the data would fit on 320GB (it does not). That would save a lot of money and a lot of power (1.3Kw of power and 1.3Kw of cooling).

The current flash disks are built with 16 Gb NAND FLASH. When, in 2012, they are built with a 1 terabit part, the device will have 2TB of capacity and will indeed be able to store the TPC-C database. So we could replace a 44k$ disk array with a few (say 10) 400$ flash disks (maybe).

If one looks at the system diagram of the Samsung NSSD there are many opportunities for innovation. It suggests interesting RAID options for fault tolerance (combining the MSR-TR-2006-176 ideas with non-volatile storage map and a block-buffer, and with writing raid-5 stripes of data across the chip array), adding a battery, adding logic for copy-on-write snapshots, and so on. These devices enable whole new approaches to file systems. They are potential gap fillers between disks and RAM and they are interesting "hot data" storage devices in their own right.

Summary

In this new world, magnetic disks provide high-capacity inexpensive storage and bandwidth – they are cold storage and archive. FLASH disks provide nonvolatile storage for hot and warm data. FLASH may also be used within disk drives to buffer writes and to provide safe write caching. In this new world, disks look much more like tapes, and FLASH disks fill the direct-access block storage traditionally filled by magnetic disk. But, flash is a better disk (more IOps 10x less latency) and disk is a better tape (no rewind, mount times measured in milliseconds). FLASH $/GB prices are far below the disk prices of 5 years ago and disk $/GB is far better than tape when one considers the total system cost (readers, software, and operations). So, these changes are very welcome.