Future flash-based disks could provide breakthroughs in IOPS, power, reliability, and volumetric capacity when compared with conventional disks.

Flash Disk Opportunity for Server Applications

JIM GRAY AND BOB FITZGERALD

NAND flash densities have been doubling each year since 1996. Samsung announced that its 32-gigabit NAND flash chips would be available in 2007. This is consistent with Chang-gyu Hwang's flash memory growth model¹ that NAND flash densities will double each year until 2010. Hwang recently extended that 2003 prediction to 2012, suggesting 64 times the current density—250 GB per chip. This is hard to credit, but Hwang and Samsung have delivered 16 times since his 2003 article when 2-GB 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 can be packaged as a disk replacement. Samsung has a 32-GB flash disk (NSSD-NAND solid-state disk) that is PATA (parallel advanced technology attachment) now and SATA (serial ATA) soon. It comes in standard 1.8-inch and 2.5-inch 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 16-GB flash disk for \$170 and a 32-GB disk later, and SanDisk (which bought msystems, a longtime manufacturer of flash disks for the military) has a 32-GB disk for about \$1,000.

These “disks” are expensive (list price is \$1,800 on the Web,² and the \$170 model is not yet available); however, they consume about 15 times less power (0.9 watts vs. 14 watts), and they (potentially) deliver approximately 10 times more accesses per second (2,500 vs. 200 I/O operations per second) than high-performance SCSI disks. They are also highly shock resistant (>1 kg). Through good engineering, they have circumvented two flash shortcomings: (1) a particular byte of flash can be written only 1 million times; and (2) one cannot write zeros to a page, so one must erase (reset) the page to all ones and then write them—this makes writes slow. The system diagram shown in figure 1 hints at some of the ways Samsung does this.

Tom's Hardware did an excellent review of the Samsung product.³ Two articles by Microsoft Research measure several USB flash drives,^{4,5} 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 1 GB of RAM with a beta NAND flash 32-GB disk, and Dennis Fetterly gave me access to a 4-GB msystems UFD Ultra USB device.

FIGURE 1

**The Block Diagram for the Samsung Flash Disk
Recreated from Its Spec Sheet**

THE TESTS

I tested the sequential and random performance of read and write operations 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 MB), and 1, 2, or 4 outstanding I/Os—that is, DiskSpd issues N ($= 1, 2, 4$) I/Os. When one completes, we issue one more till we are done. In the sequential test, each subsequent I/O goes to the next block in the 1-GB file generated by `genfile -r128 -s- -b4096 80M test.dat`.

The random I/O tests follow the same pattern except that they issue each I/O to a randomly chosen block-aligned location in the file. The msystem's performance was not as good as the other product, so I report the better numbers in table 1.

MEASUREMENTS WITH WRITE CACHE DISABLED—RANDOM WRITES ARE PROBLEMATIC

The graphs in figure 2 tell the story. The device sequential read-write performance is very good. At four-deep, 512-KB 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 128-KB requests, giving 53 MBps for reads and 35 MBps for writes.

The story for random I/Os is more complex—and disappointing. For the typical four-deep, 8-KB random request, read performance is a spectacular 2,800 requests per second but write performance is a disappointing 27 requests per second. Clearly, something is wrong with the random write performance. As the next section shows, Windows is doing synchronous writes and the device has fairly long latency (30 ms) for writes.

ENABLING ADVANCED PERFORMANCE

By default, in Windows Vista the disk must write data to a nonvolatile 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 past five years without ill effect (so far).

To take this risk, go to Start → Computer → Manage → DeviceManager → DiskDrives and right-click on the disk you want to risk. Select Properties from the menu, then the

TABLE
1

Measurements of a Current Flash Disk's Performance

Device	Sequential	Random 8 KB	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

FIGURE 2

- 1 deep read
- 2 deep read
- - - 4 deep read
- 1 deep write
- 2 deep write
- - - 4 deep write

Sequential Bandwidth vs. Request Size

Random IO/s vs. Request Size

Random IO/s vs. Request Size

Policy tab, and, after you read the warning label, select “Enable advanced performance.” This page is a bit confusing: it says you cannot modify the WCE (write cache enable) performance, but then it gives you two radio buttons that do so.

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. The article “A Design for High-Performance Flash Disks”⁶ explains the problem and offers a solution. Clearly, a little intelligence in the disk controller could buffer the writes and give performance comparable to the 1,100 I/Os per second 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 IOPS AND WERE “BIG”?

My tests and those of several others suggest that flash disks can deliver about 3,000 random 8-KB reads per second, and with some reengineering, about 1,100 random 8-KB 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 four times or more. Even the current performance, however, would be *very* attractive for many enterprise applications. For example, the TPC-C benchmark has approximately equal reads and writes. Using the graphs in figure 2, a weighted average of the four-deep, 8-KB random read rate (2,804 IOPS) and the four-deep, 8-KB sequential write rate (1,233 IOPS) results in a *harmonic average* of 1,713 IOPS (one-deep is 1,624 IOPS). TPC-C systems are configured with approximately 50 disks per CPU. For example, the most recent Dell TPC-C system has 90 15,000-RPM 36-GB SCSI disks costing \$45,000 (with \$10,000 extra for maintenance that gets “discounted”). Those disks are 68 percent of the system cost. They deliver about 18,000 IOPS. That is comparable to the requests per second of 10 flash disks. So we could replace those 90 disks with 10 NSSDs if the data would fit on 320 GB (it does not). That would save a lot of money and 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 2 TB of capacity and will indeed be able to store the TPC-C database. We could replace a \$44,000 disk array with a few (say, 10) \$400 flash disks (maybe).

The system diagram of the Samsung NSSD suggests many opportunities for innovation: interesting RAID options for fault tolerance (combining the ideas from “A Design for High-Performance Flash Disks”⁷ with a nonvolatile 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—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 role traditionally filled by magnetic disks. Flash is a better disk (more IOPS, 10 times less latency), however, and disk is a better tape (no rewind, mount times measured in milliseconds). Flash cost per gigabyte is far below the disk prices of five years ago, and disk cost per gigabyte is far better than tape when one considers the total system cost (readers, software, and operations). Thus, these changes are very welcome. □

REFERENCES

1. Hwang, C. 2003. Nanotechnology enables a new memory growth model. *Proceedings of the IEEE* 91(11): 1765-1771; <http://ieeexplore.ieee.org/Xplore/login.jsp?url=/iel5/5/27802/01240069.pdf>.
2. This Web site (<http://www.dvnation.com/nand-flash-ssd.html>) had a PQI flash disk for \$1,800. The marginal cost is about \$20 per gigabyte for flash today, so this device might be had for about \$600, comparable to the price of a 15,000-RPM SCSI disk.
3. Zushman, J. 2006. Hard drives go flash: Samsung Flash SSD. Tom's Hardware; <http://www.tomshardware.com/reviews/conventional-hard-drive-obsoletism,1324.html>.
4. Nath, S., Kansal, A. Flash DB: Dynamic self-tuning database for NAND flash. Microsoft Research Technical Report MSR-TR-2006-168; <ftp://ftp.research.microsoft.com/pub/tr/TR-2006-168.pdf>.
5. Birrell, A., Isard, M., Thacker, C., Wobber, T. 2005. A design for high-performance flash disks. Microsoft Research Technical Report MSR-TR-2005-176; <ftp://ftp.research.microsoft.com/pub/tr/TR-2005-176.pdf>.
6. See reference 5.
7. See reference 5.

ACKNOWLEDGMENTS

This article was prepared with help from Aaron Dietrich, Dennis Fetterly, James Hamilton, and Chuck Thacker.

LOVE IT, HATE IT? LET US KNOW

feedback@acmqueue.com or www.acmqueue.com/forums