Filling the Memory Access Gap:
A Case for On-Chip Magnetic Storage

Steven W. Schlosser, John Linwood Griffin

David F. Nagle, Gregory R. Ganger

November, 1999

CMU-CS-99-174

School of Computer Science

Carnegie Mellon University

Pittsburgh, PA 15213

Abstract

For decades, the memory hierarchy access gap has plagued computer architects with the RAM/disk gap widening to about 6 orders of magnitude in 1999. However, an exciting new storage technology based on MicroElectroMechanical Systems (MEMS) is poised to fill a large portion of this performance gap, delivering significant performance improvements and enabling many new types of applications. This research explores the impact MEMS-based storage will have on computer systems. Working closely with researchers building MEMS-based storage devices, we examine the performance impact of several design points. Results from five different applications show that MEMS-based storage can reduce application I/O stall times by 80–99%, with overall performance improvements ranging from 1.1× to 20× for these applications. Most of these improvements result from the fact that average access times for MEMS-based storage are 5 times faster than disks (e.g., 1–3ms). Others result from fundamental differences in the physical behavior of MEMS-based storage. Combined, these characteristics create numerous opportunities for restructuring the storage/memory hierarchy.

David Nagle and Steven Schlosser are supported by an NSF CAREER Grant. John Griffin is supported by a National Science Foundation Graduate Fellowship.

Keywords: MEMS, memory hierarchy

1 Introduction

For decades, the memory hierarchy has suffered from significant access, bandwidth and cost gaps between processor, RAM, and disk [12]. Fortunately, the processor/RAM gap has been mitigated by fast cache memories [11]. Unfortunately, the RAM/disk gap has remained unfilled, widening to 6 orders of magnitude in 1999 and continuing to widen at about 50% per year. The result is a significant performance and scalability problem across a range of applications, including databases, web servers, mail servers, program development tools, and even Microsoft Word load times [4].

This RAM/disk performance gap is due directly to the physical characteristics of disk drives. While disks continue to deliver capacity growth of over 60% per year, the physics of a drive's mechanical positioning system limits disk access time improvements to a modest 7% per year [11]. EEPROM offers a portable high-performance alternative. However, EEPROM's per-megabyte cost is 2 orders of magnitude higher than disk storage (see Figure 1).

MEMS-based storage is an exciting new technology that could provide significant performance gains over current disk drive technology and at costs much lower than EEPROM technology [10, 2]. Based on MEMS (MicroElectroMechanical Systems), this non-volatile storage technology merges magnetic recording material and thousands of recording heads to provide storage capacity of 1–10 GB of data in under 1 cm^2 area with access times of 1–3 ms and streaming bandwidths of over 50 Mbytes/second. Further, because MEMS-based storage is built using photolithographic IC processes compatible with standard CMOS, MEMS-based storage has costs significantly lower than DRAM and access times an order of magnitude faster than conventional disks [10].

Another very important aspect of MEMS-based storage is its ability to incorporate both storage and processing into the same chip. Because MEMS-storage is CMOS-based, it is possible to integrate several microprocessors or hundreds of custom computational engines (e.g., MPEG encode/decode, cryptography) directly with the storage device. This integration will significantly improve performance, power consumption, and cost. More importantly, it will lay the foundation for a single computing brick [6] that contains processing, and both nonvolatile and volatile storage.

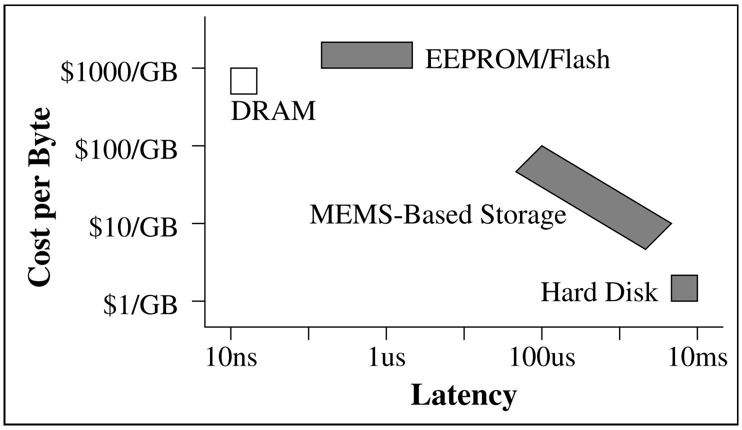


Figure 1: ***Predicted cost and latency for storage technologies in 2005.*** *MEMS-based storage fills the growing memory hierarchy gap between RAM and disk. The grey boxes represent nonvolatile storage. The EEPROM box is wide because of the wide gap between read and write latencies for "Flash" memories. The MEMS box is wide and tall because of the many design possibilities for this new type of storage (see Section 2).*

Although MEMS-based storage devices are still several years away from commercialization, their potential impact in reducing the memory gap makes them an important technology for systems architects' consideration. This work begins the exploration process, seeking an initial understanding of how MEMS-based storage can improve computer systems' performance and how different MEMS device characterictics can fundamentally change the behavior and design of storage systems. Our results show that MEMS-based storage can reduce application I/O stall times by over 80–99% for a set of five file system and database workloads. The resulting speedups for these applications range from 10% to 20X, depending mainly on the ratio of computation to I/O.

To ensure that our models of MEMS-based storage accurately reflect potential implementations, we work closely with a group of researchers who are actively building MEMS-based storage devices. This collaboration allows us to explore the system-level impact of various types of MEMS-based storage, evaluating which physical design trade-offs (e.g., acceleration speed, velocity, size, capacity) are most important across a range of applications. In turn, our results feed back to the MEMS researchers, focusing their attention on design parameters that significantly impact system-level performance and avoiding optimizations that provide little real benefit.

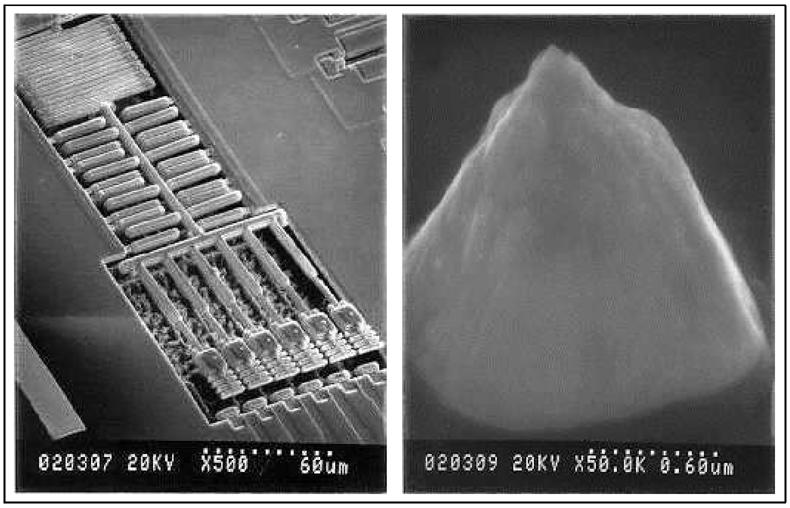
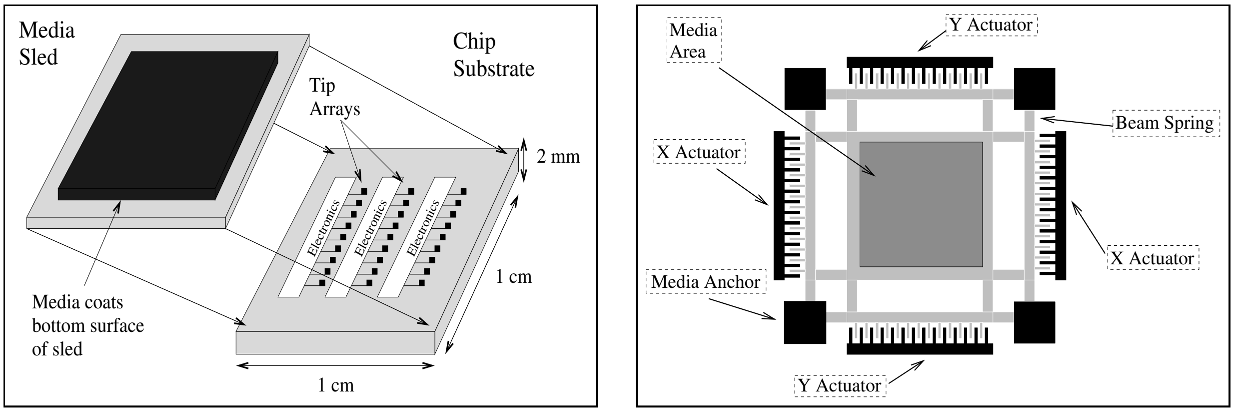


Figure 2: ***Prototype Positioning System and Probe Tip.*** *The CMU MEMS research group has developed the prototype probe tip and positioning system shown above. Because the recording material is not perfectly flat, the positioning system must be able to actively adjust the height of the probe tips. The tips could use one of several recording schemes, from simple "typewriting" with permanent magnets, to more complex GMR sensing techniques found in normal disk drives.*

The remainder of this paper is organized as follows. Section 2 describes MEMS-based storage, many of the physical trade-offs, and three models we have developed to explore the design space. Section 3 describes our performance model for MEMS-based storage devices and uses microbenchmarks to analyze its basic performance. Section 4 presents results for a number of applications. Section 5 discusses more general system-level issues and explores a wide range of applications for MEMS-based storage. Section 6 outlines our conclusions and continuing work.

2 MEMS-based storage devices

Microelectromechanical systems (MEMS) are very small scale mechanical structures—on the order of tens to thousands of micrometers—fabricated on the surface of silicon wafers. These microstructures are created using the same photolithographic processes used in manufacturing other semiconductor devices. Certain MEMS structures can be made to slide, bend, or deflect in response to an electrostatic or electromagnetic force from a nearby actuator or from external forces in the environment. MEMS-based microstructures are limited in mobility compared to standard mechanical systems. To illustrate, it is difficult to build durable



(a) Media sled and read/write tips (b) Media sled, beam springs, and actuators

Figure 3: ***An example of the "moving media" model.*** *In (a), we see how the media sled is attached above the fixed tips. The sled can move up to 100 μm along the X and Y axes, allowing the fixed tips to address 30–50% of the total media area. In (b), we see the actuators, the spring suspension, and the media sled itself. Anchored regions are shown in black and the movable structure is in grey.*

microbearings to actuate and position rotating components. Previous attempts at building micromachined gear series have shown that such devices lock up from friction within a few thousand revolutions. However, alternative designs such as spring-suspended masses which translate in the X and Y directions (instead of rotating in θ) circumvent these frictional barriers.

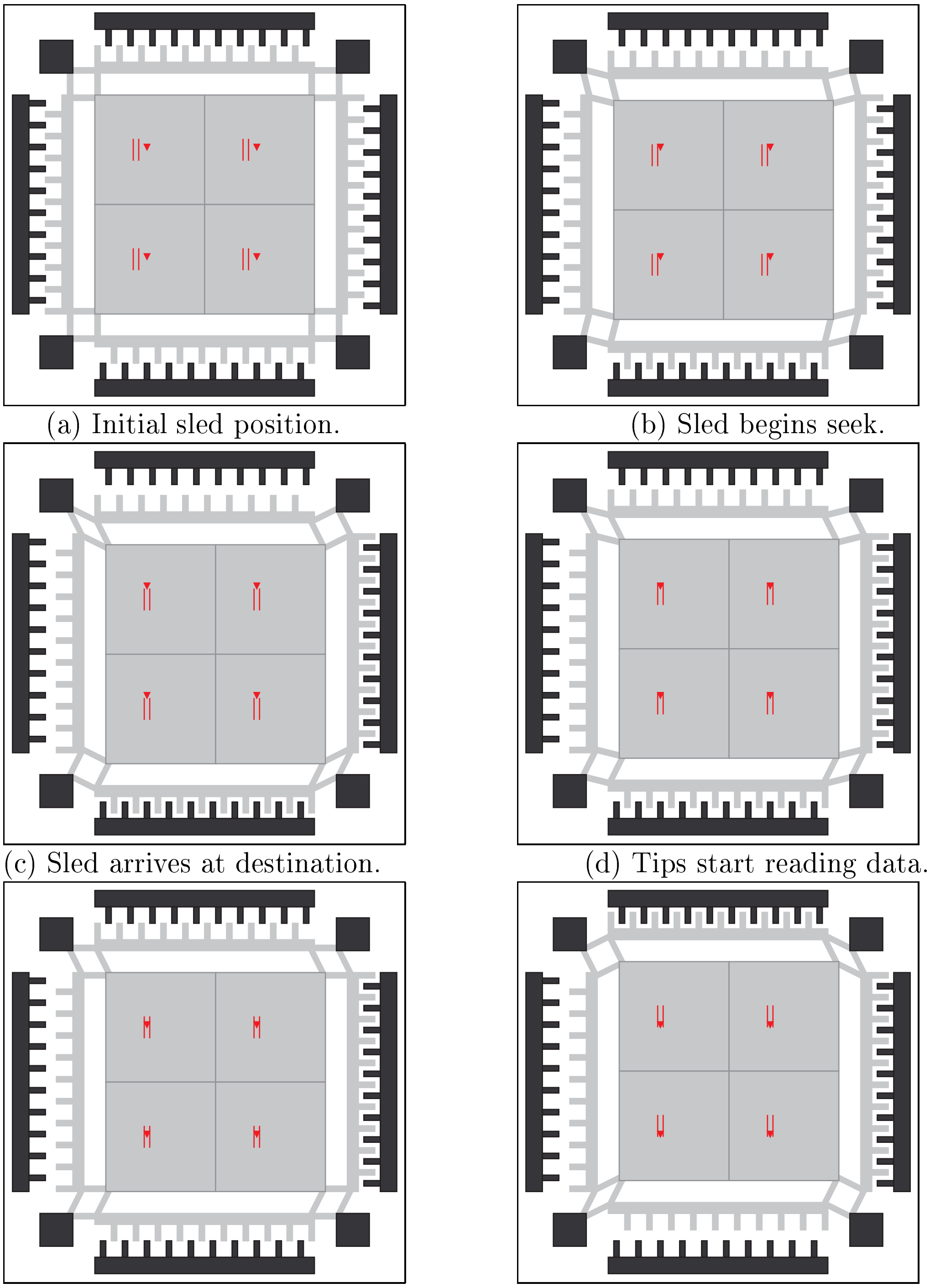
One class of MEMS-based storage system structures under investigation takes advantage of arrays of thousands of microscopic magnetic probes each accessing a dense substrate of magnetic material [2, 10]. This design offers several notable advantages over disk-based storage, including better cost, access time, power dissipation, mass, failure rate and shock sensitivity. Further, there is inherent parallelism across the wide array of read-write tips: multiple tips may be accessed concurrently to increase throughput, accesses may be redundant to enhance reliability, or completely independent accesses may occur in parallel. In addition, the MEMS fabrication process integrates seamlessly with standard CMOS processes. This ease of fabrication opens the door for mass manufacturing true MEMS-enhanced systems-on-a-chip—massively parallel manufacturing, small per-unit cost in high volume, a clear road map toward smaller processes, and large amounts of industry momentum.

For magnetic probe-based MEMS storage there are two basic design types for accessing data. The first employs an array of movable probe tips, each suspended on a cantilevered beam and positioned over a fixed substrate of magnetic media. The beam moves by applying a voltage to a set of X-deflectors and Y-deflectors, resulting in an electrostatic force that causes the tip to move deterministically to different positions over the media. This "fixed media" model has access times on the order of tenths of milliseconds. Unfortunately, each tip addresses only about 1% of the magnetic material under each cantilever. For comparison, a conventional rotating disk accesses about 50% of the available media.

A second design, used in the experiments described in this paper, significantly increases storage capacity by replacing the fixed media with a movable sled over an array of stationary tips. This movable sled is capable of moving 100 μm along the X and Y axes (see Figure 3) achieving 30% media coverage. To read or write data, the sled first seeks until the requested data is directly over a set of tips (See Figure 4). The sled then moves at constant velocity in the Y direction only, streaming data to or from the media. The movable sled is much more massive than individual cantilevered tips, to the extent that the movable sled model operates at an order of magnitude greater latency than the fixed media model.

2.1 MEMS device characteristics

MEMS-based storage devices have a rich set of characteristics. For example, moving the media sled to access data creates the equivalent of a disk-like seek time. This includes the time to move the sled to the correct starting position and initial read/write velocity for the access, and is bounded by the amount of force available from the sled actuators and the mass of the sled. Another parameter, the media access time, varies based on the number of tips active during the media access, the number of bits each tip needs to process, and the bit rate per tip. For example, to read 100 bits, one tip could read at 100 kbit/s while the media travels 100 bit widths (1 ms), or four tips could read at 100 kbit/s each while the media travels 25 bit widths (0.25 ms). There is unfortunately a limit to the number of tips that can be simultaneously active. Although specific numbers for power and heat generation are not known, we assume for now values of 1–3 mW per active tip and 100 mW continuous for the media positioning system. As we envision devices with 10,000 tips/cm^2, using all tips simultaneously results in a power dissipation of 10.1–30.1 W/cm^2! For this reason we limit



(a) Initial sled position.
(b) Sled begins seek.
(c) Sled arrives at destination.
(d) Tips start reading data.
(e) Sled drags further past tips.
(f) Data access complete.

Figure 4: ***Notional drawing of data access in a MEMS-based storage device.*** *This series of drawings shows the moving parts of the sled system as it accesses a region of data, as indicated with lines. The X and Y actuators pull the sled and the springs flex. All components shown in black (the comb actuators and the spring anchors) remain fixed while the components in grey (the sled and the springs) are free to move. It is important to notice that the tips, shown as small triangles, also remain stationary, as they are fixed to the underlying chip (as in Figure 3).*

	1st gen.	2nd gen.	3rd gen.
bit width (nm)	50	40	30
sled acceleration (g)	11.7	11.7	17.6
access speed (kbit/s)	200	400	400
access speed (mm/s)	10	16	12
resonant frequency (Hz)	220	220	330
post-seek X settling time (ms)	1.447	0.723	0.482
maximum throughput (MiB/s)	10.85	21.70	54.25
number of sleds	1	1	1
per-sled capacity (GiB)	2.098	3.219	5.880
bidirectional access	no	yes	yes

Table 1: ***MEMS device parameters used in our experiments.***

the number of concurrent tips in our models (described below) to only 640–3200 tips.

Given the wide range of parameters, exploring the entire MEMS-based storage design space would take a considerable amount of time. To reduce this effort, we constructed three models of MEMS-based storage based on what we anticipate will be the technology advances over the first three generations. We describe each model below and summarize the parameters in Table 1.

The "1st generation (G1)" model represents the initial MEMS storage devices, which we expect could be fabricated within the next three years [10]. Each sled will have a full range of motion of 100 μm along the X and Y axes, and the actuators will accelerate the sled at 11.7g. To access data, the device uses a typewriting method of moving the probe tip up and down over every bit. This typewriting method is inefficient and limits read speed to about 200 kbits/s per tip. This design only offers unidirectional accesses; for example, reads and writes may occur only when the sled is moving in the positive Y direction.

Under the G1 model the tip resolution and sled positioning system provide a square bit cell of 50 nm sides such that each tip addresses a 2000×2000 array of bits. The sled footprint is 0.64 cm^2 allowing 6,400 tips underneath each sled. The sled travels at 10 mm/s during media access but is not restricted to that speed during seeks. Although these numbers appear to yield a capacity of 2.98 GiB per sled, the capacity decreases because of two factors. Error detection and correction from the media demands a 10-bit-per-byte encoding. Also, sled tracking and synchronization information on the media introduces approximately

11% overhead—about 10 bits every 80 data bits. This yields an effective capacity of about 2.098 GiB per sled.

The "2nd Generation (G2)" model. Several fundamental changes occur in G2. Data are encoded in G2 to allows media access in either the positive or negative Y direction. Also, we expect G1's typewriting scheme to be replaced by a Giant Magnetoresistive (GMR) head design, logically similar to current disk technology, which allows at a minimum the doubling of the read speed to 400 kbit/s or 16 mm/s (not 20 mm/s because the bit width is 20achieve access speeds of 1 Mbit/s per tip, but our initial results indicate that given an acceleration for the sled of 11.7g, a point of diminishing return is quickly reached where the gains of faster media access are negated by the excess time needed to accelerate the sled to speed. G2 also increases the bit density by 20%, reflecting trends in magnetic materials.

The "3rd Generation (G3)" model approaches the high end of many of the MEMS parameters and characteristics. Although somewhat speculative, we believe many of these are achievable given an applied long-term MEMS research and development program. G3 anticipates a decreased sled mass providing a better resonant frequency for the sled as well as a higher acceleration drive, both of which improve the sled seek time. G3 also increases the resolution of the tips and positioning system to a 30 nm bit width, decreasing the sled access speed by 25% and increasing the overall capacity to over 5.8 GiB/sled! Access speed in Table 1 decreases because the smaller bit width drops the baseline read speed to 3 mm/s and we continue to use 400 kbit/s tip access times.

The reference disk. To compare MEMS-based storage against conventional disk drives we used a Seagate Technologies *Cheetah 4LP ST-34501W*. Representative parameters for the Cheetah 4LP are provided in Table 2. Although originally introduced in 1997, the Cheetah 4LP is still considered one of the high-end, high-performance disk drives available today. We are fortunate to have access to a validated DiskSim module for the Cheetah 4LP [15]; this enables us to execute a direct comparison of Cheetah performance to MEMS device performance.

The SuperDisk model was created to compare MEMS-based storage to an aggressive disk drive projection to the year 2005. Extrapolating on the current performance trends in disk drive technology, our SuperDisk achieves streaming bandwidth of up to 100 MB/second.

	Cheetah	**"SuperDisk"**
RPM	10,033	14,400
sectors per track	130–194	520–776
data surfaces	8	2
average latency	2.99 ms	2.08 ms
average rotational seek (read/write)	7.7 ms/8.7 ms	5.0 ms/5.0 ms
max full stroke	18.2 ms/19.2 ms	10.6 ms/10.6 ms

Table 2: ***A comparison of the Seagate Cheetah 4LP ST-34501W disk drive and the extrapolated SuperDisk model.*** *Specifications for the Cheetah 4LP are from [16, 15].*

Its seek time drops by 40% (i.e., 7% per year) to a 5 ms average and rotates at 14,400 RPM. The Cheetah and SuperDisk parameters are compared in Table 2.

3 Performance of MEMS-Based Storage Devices

This section overviews how we model the performance of MEMS-based storage devices. Because these devices are in their infancy, our simulation model's timings are derived from extensive discussions with researchers who are actively developing this technology. In return, our results are helping these researchers refine their designs by identifying system-level problem areas. A detailed description of our performance model for MEMS based storage and an exploration of its performance sensitivity to various design parameters and "disk" scheduling algorithms are presented in [8].

We integrated our simulation module for MEMS-based storage into DiskSim. DiskSim is a freely-available disk simulator that has been proven to very accurately model disk drives [5], including the Seagate Cheetah used as a baseline in this paper. DiskSim also includes a synthetic I/O workload generation module, which we used for the microbenchmark experiments in Section 3.2.

3.1 A Model of MEMS-Based Storage Performance

The set of bits that can be accessed by a given probe tip are arranged in a square, and each can be identified by its <x,y> coordinates. Bits are read by moving the media sled over the tips in the Y dimension at the *access velocity*, which is determined by the bit width and the rate at which a tip can read or write bits. To allow a tip to access a set of bits, the media sled must first *seek* to the proper <x,y> location. Then, to complete an access, the sled must slide past the active tips until all desired data are transferred. Thus, the access time for a given request is the sum of the seek time and the transfer time.

Because separate actuators are used for the X and Y movements, they are independent and the time required for a given seek is the maximum of the two. The model keeps track of the sled's position and velocity, and first-order mechanics provide timings for X and Y seeks. For example,

$$t_{seek\_x} = 2 * \sqrt{\frac{(\Delta x)}{a}} + t_{settle}$$

because an X seek starts at rest, accelerates at full speed for half the seek distance, decelerates at full speed for half the seek distance, and comes to rest at the destination column of media bits. t_{settle} represents the time required for oscillation of the spring-mounted sled to damp enough for the probe tip to function; it is dependent mainly on the resonant frequency of the sled. A similar equation is used for seeks in the Y dimension.

Requests arriving at a MEMS device are addressed to 512 byte logical block numbers (LBNs) as in SCSI. A mapping function translates LBNs to physical sled positions. In our model, LBNs of 512 bytes are striped across 64 concurrently *active tips*, which represents a subset of the total number of concurrently active tips at any point during a transfer. Combined with the servo and encoding overheads, 90 bits per tip must be read in order to read a full logical block. Thus, in the time required to read 90 bits, many logical blocks can be read (e.g., 10, if 640 tips can be concurrently active). Sequential logical blocks are placed at ascending <x,y> starting positions, filling each column of concurrently active tips before moving to the next. Thus, for multi-sled devices, the blocks of a single address space are striped across the sleds using the maximum possible stripe unit size (the capacity of a single sled).

3.2 Microbenchmark Results

To understand the base performance of a MEMS-based storage device, we measured its performance on a set of 10,000 random requests. Two thirds of the requests were reads, and the arrival rate was 20 requests per second. Figure 5 shows that the performance of all

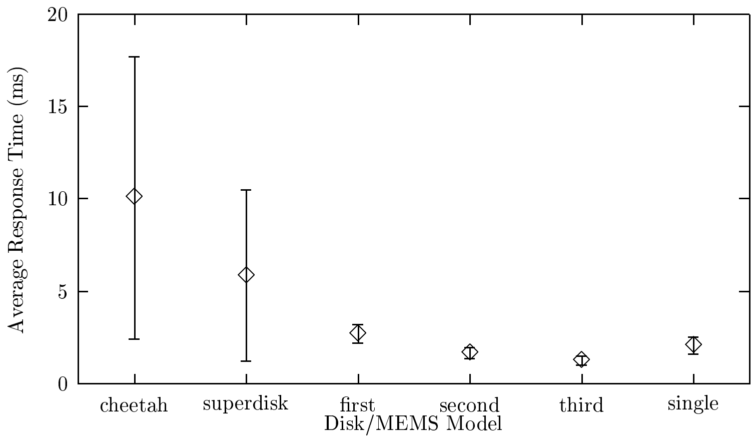


Figure 5: ***Average total response times of each model under the microbenchmark.*** *Interesting features to note are the overall better performance from the MEMS devices and their smaller variances.*

three MEMS models beat that of the Cheetah and SuperDisk disks by almost 3× and 2×, respectively.

Figure 5 also shows that the MEMS devices have much less access time variation than disk drives. In a disk drive, the maximum distances over which the heads and media must travel to reach an individual block vary quite a bit, causing the wide variation in access time. In this experiment, the coefficients of variation ($\frac{\sigma}{\mu}$) for the Cheetah and SuperDisk access times are 0.76 and 0.79, respectively. In contrast, the MEMS-based storage devices have coefficients of variation between 0.18 and 0.20. This is due equally to the absence of rotational latency and the fact that a full throw of the media is on the order of tens of microns as compared to centimeters in a disk drive. Therefore, seeks times are tightly constrained. The lower variances, and thus greater potential predictability, has intriguing consequences for the design of embedded systems with both storage and real-time requirements.

Another characteristic, which we do not show in this graph, is the benefit of parallelism. A MEMS-based storage device can easily consist of multiple fully-independent sleds over which data are striped. A conventional disk queues incoming requests when the device is already servicing a previous request, because there is only one mechanism for accessing the media. However, a multi-sled MEMS-based storage device can simultaneously service multiple separate requests if their data falls on separate sleds, much like disk arrays. To quantify the benefit of parallelism, we increased the arrival rate and compared a 4-sled model with the single-sled model – as expected, the 4-sled device provides 4 times the throughput for this random workload. We continue to characterize the device's parallelism characteristics, but early results indicate that inter-sled stripe unit trade-offs conform to the expectations given by earlier disk striping work [13, 3]. In addition, similar benefits can be gained by aggregating multiple single-sled devices together, as in a RAID system. Given the significantly lower volume of MEMS-based storage devices, many independent sleds could be fit into a standard drive enclosure, increasing both the performance and the capacity per volume relative to conventional disks.

4 Application results

To successfully fill the memory/storage gap, MEMS-based technology must offer a significant improvement in I/O and overall application performance. Using six different applications, this section compares the performance of our MEMS-based storage device models (G1, G2, and G3) against a 1997 Seagate Cheetah disk drive and our hypothetical SuperDisk.

4.1 Simulation Environment

To explore the impact of MEMS-based storage devices on real application performance, we incorporated our modified version of DiskSim into SimOS (see Figure 6). SimOS is a complete machine simulator, capable of booting real operating systems and running real applications [14]. SimOS was configured to model a 500 MHz 21164-based system (128 MB RAM) running Digital UNIX.[1] The OS runs atop the virtual machine, using special device drivers to interact with simulated I/O devices. We incorporated DiskSim into SimOS, replacing its default disk model. For each of our experiments, we varied only the storage device, fixing all other variables.

[1] It is important to note that a fairer comparison would scale the processor architecture. We are currently modifying our simulation environment to support the processor architecture we anticipate in 2005. This change should significantly improve the user and system CPU time, increasing the relative importance of I/O by making I/O performance a much larger percentage of each application's run time.

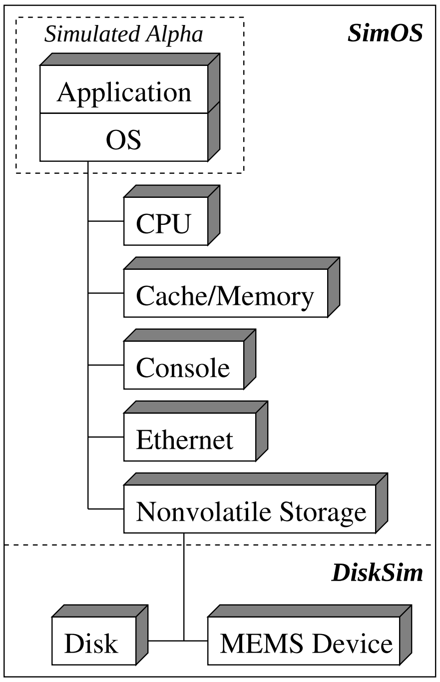


Figure 6: ***Our simulation environment.*** *The MEMS device model integrates with the DiskSim subsystem simulator to provide the storage component of the SimOS machine simulator. DiskSim can simulate either storage model (disk or MEMS device) or both models simultaneously.*

4.2 Results

Our first two applications, the Andrew Benchmark Suite [7] and PostMark [9] were designed for file system and I/O performance analysis. The Andrew Benchmark consists of a set of file and directory operations followed by a long compile. The PostMark benchmark performs many small file operations (e.g., create, delete, read, write) and was designed to be representative of the file system workloads seen in e-mail, news, and web commerce environments. Figures 7 and 8 show that MEMS-based storage devices can largely eliminate the I/O wait times for these workloads. For Andrew, the G2 MEMS-based storage device provides a modest 2% additional reduction in I/O wait time beyond the first. The improvement is due to the G2 model's ability to access data as the sled moves in either Y-direction (i.e., up or down).

The data for PostMark (Figure 8) shows a dramatic benefit for MEMS-based storage devices even when compared to the SuperDisk. This impressive improvement comes from a fundamental physical difference in how MEMS-based storage accesses data. Specifically, the frequent create and delete operations in PostMark cause repeated synchronous writes to file system metadata, forcing the storage devices to make same sector, back-to-back updates. For a conventional disk, such back-to-back same-sector accesses require a full rotation (typically 6–8 ms on today's disks) between updates. This explains why PostMark spends much of its I/O time waiting for full disk rotations. MEMS-based storage does not involve rotating platters, and so the MEMS models do not suffer from these full rotation latencies for back-to-back rewrites. Specifically, MEMS-based storage can write a sector, immediately reverse direction and then rewrite the sector in 0.3 ms. This physical difference gives MEMS-based storage a fundamental performance advantage over rotating media for this access pattern. While this specific problem could be significantly reduced with a small amount of write-back caching (either in the file system or at the disk), similar behavior is exhibited by many read-modify-write activities, such as transaction processing and RAID parity updates.

The next set of benchmarks, GNULD and the TPC-D queries also show significant performance improvements for MEMS-based storage. However, Figure 9 shows that the SuperDisk outperforms the G1 MEMS-device for TPC-D query 4, because SuperDisk's higher streaming bandwidth more than compensates for the higher access times for this data mining query. However, a disk drive's streaming bandwidth varies by ~40%, depending on the location of the data (i.e., outer vs. inner tracks). For these experiments, all of the data is located on the disk's outer tracks, making the performance best-case. In contrast, MEMS devices do not have any variation in streaming bandwidth (for contiguous data). Therefore, if the data had resided on SuperDisk's inner (i.e., slower) tracks, SuperDisk would have seen performance much closer to the G1 MEMS-device. With its increased bandwidth, the G2 MEMS device's lower access times allow it to outperform the SuperDisk.

The results for TPC-D query 6, shown in Figure 10, show the expected result for workloads that are CPU-bound rather than I/O-bound — eliminating the I/O stall time provides only a modest 8% decrease in overall runtime. As CPU speeds continue to increase relative to disk speeds, of course, the importance of I/O increases.

For several of the benchmarks, CPU time decreases slightly with the better-performing MEMS devices. All of these decreases are in the system time charged to the application. The reason for the decrease is that shorter runs times reduce the amount of time an application

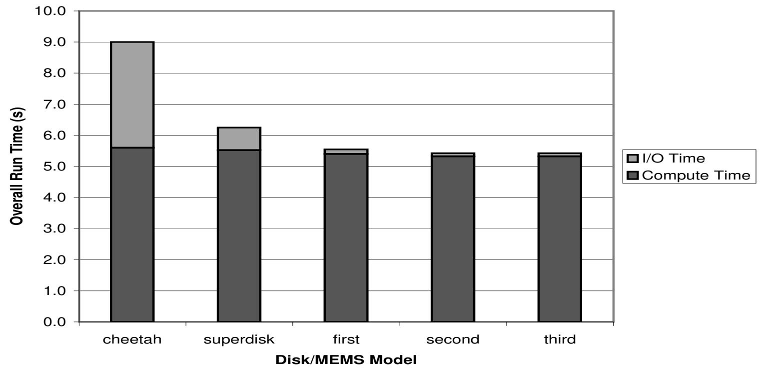


Figure 7: ***Runtime for the Andrew Benchmark.***

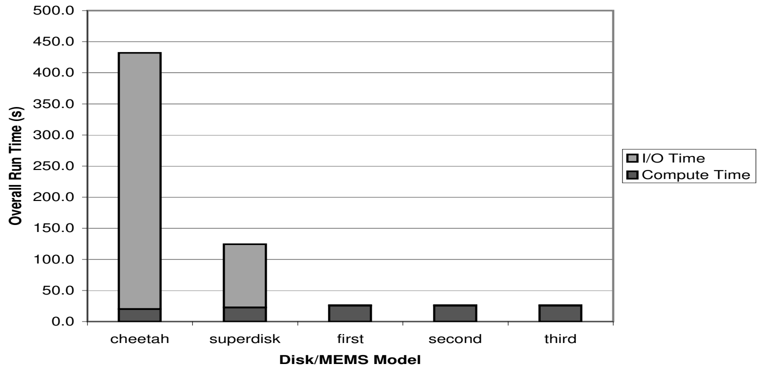


Figure 8: ***Runtime for the PostMark Benchmark.***

can be charged for general system overhead, such as I/O interrupt handling. Therefore, system time will generally decrease by a modest amount when applications complete in less time.

5 Potential of MEMS Storage and Computation

MEMS-based storage devices fill a growing hole in the classical memory hierarchy. Their I/O performance can be an order of magnitude better than disk drives and their physical characteristics provide some fundamental performance advantages that rotating media cannot compete against. Other advantages, such as their physical size, portability, and the potential to integrate processing within the same substrate, create many exciting possibilities for system architects.

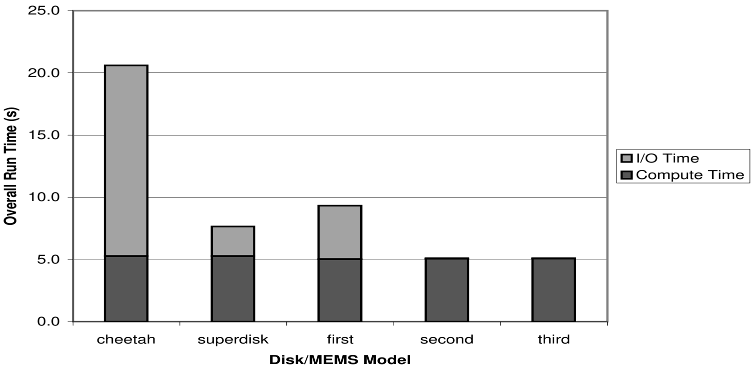


Figure 9: ***Runtime for TPC-D Query #4.***

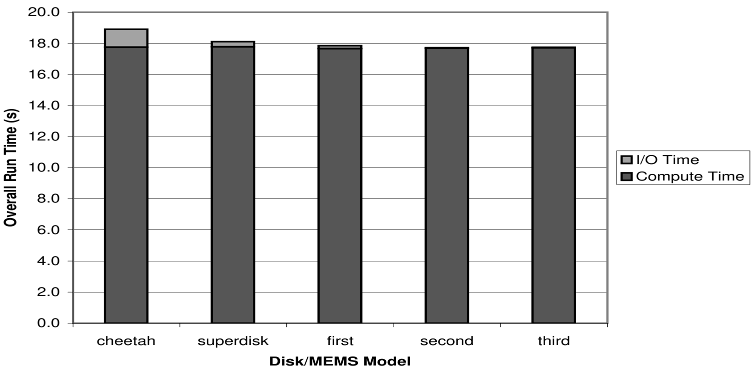


Figure 10: ***Runtime for TPC-D Query #6.***

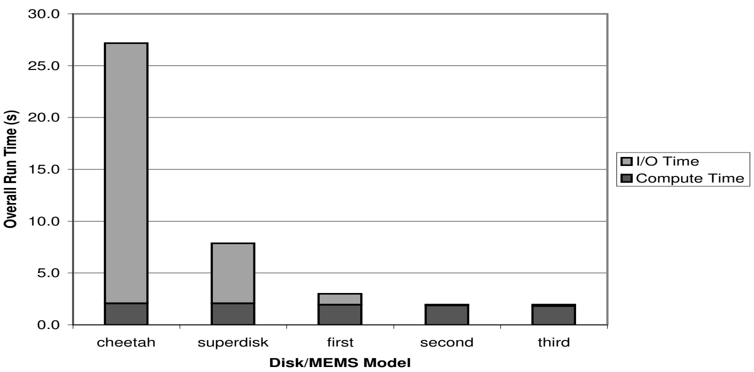


Figure 11: ***Runtime for Gnuld.***

As we have explored here, MEMS-based storage could be an attractive alternative to disk. Cost, however, is often the judge of a technology. MEMS-based storage creates a new low-cost entry point for modest-capacity applications of 1–10 GB. This is because disks' assemblies of mechanical components keep manufacturing costs from falling below a certain point, while MEMS-based storage rides the linear decline in IC manufacturing process costs. However, drives enjoy a 10× price advantage for high-capacity storage (*e.g.,* 50 GB in 1999) because the drive assembly costs are subsumed by the media cost.

For many "portable" applications such as notebook PCs, PDAs, and video camcorders, MEMS-based storage also provides a more robust and lower power solution. Unlike rotating storage, which cannot cope with device rotation (e.g., rapidly turning a PDA) and is very sensitive to shock (e.g., dropping a device), MEMS-based storage does not suffer any gyroscopic effects and can absorb much greater external forces. Further, while MEMS-based and disk storage consume approximately the same power per Megabyte when data is accessed, MEMS has much more agile standby and wakeup capability. Therefore, MEMS-based storage can rapidly switch between sleep and active mode, avoiding long and power-hungry spin-up cycles.

MEMS-based storage also represents a non-volatile addition to the storage hierarchy. For example, with their low-cost entry point, MEMS-based storage devices could be incorporated into future disk drives as a very large (1-10GB) non-volatile MEMS cache. With their superior performance, the MEMS cache could absorb latency-critical synchronous writes to metadata and cache small files to improve small read performance. Further, if the MEMS-based storage device is exposed to the OS, file systems could deliberately allocate specific data onto it, depending on their access patterns and performance needs. For example, file systems could place small structures (e.g., file system metadata) on MEMS-based storage, while using the disk platters for large contiguous or infrequently accessed data. In [1], Baker et al. show that using fast non-volatile storage to absorb synchronous disk writes both at a client and at a file server increases performance from 20% to 90%. Although the systems in [1] required only a small amount of non-volatile storage, we postulate that as file servers grow to terabyte size and network bandwidth continues to increase, larger amounts of non-volatile storage could provide further increases in performance. For these same reasons, RAID arrays would also benefit from MEMS-based storage, creating AutoRAID-like systems [17]. Further, because RAID arrays are less cost-sensitive than individual disks, arrays of MEMS-devices could be incorporated more cost-effectively.

Another application domain for MEMS-based storage is as bulk non-volatile storage for embedded computers. Single-chip "throw-away" devices that store very large datasets can be built for such applications as civil infrastructure monitoring (*e.g.*, bridges, walls, roadways), weather or seismic tracking, and medical applications. For example, one forthcoming application is temporary storage for microsatellites in very low earth orbit. Given that a satellite in a very low orbit moves very quickly, communications are only possible in very short bursts. Therefore, a low-volume, high-capacity, non-volatile storage device to buffer data is required. MEMS-based storage devices could also add huge databases to single-chip continuous speech recognition systems and be integrated into low-cost consumer or mobile devices. Such chips could be completely self-contained, with hundreds of megabytes of speech data, custom recognition hardware, and only minimal connections for power and I/O.

Another compelling opportunity presented by MEMS-based storage is near-absolute data security. With true systems-on-a-chip, sensitive data never has to move beyond the processor and the on-chip data store without being properly encrypted via on-chip circuitry. Such a design would provide no opportunity for traffic snooping devices, even if on the storage network, to capture a cleartext copy of sensitive information. Further, the self-contained nature of these components allow for the construction of inexpensive, high-capacity, tamper-proof smart cards.

6 Conclusions

MEMS-based storage has the potential to fill the ever-growing gap between RAM and disk access times. This paper describes MEMS-based storage, evaluates the impact of some emerging designs on the performance of real applications, and discusses a number of interesting architectural uses for MEMS-based storage in systems. Our results indicate that MEMS-based storage can reduce I/O stall times by 80-99%, reducing overall runtimes by 10-2000%, which suggests a very promising future for this technology. Looking ahead, our ongoing work includes explorations of how to restructure storage systems (hardware and software) to best exploit MEMS-based storage devices and of new applications enabled by this new technology.

References

[1] M. Baker, S. Asami, E. Deprit, J. Ousterhout, and M. Seltzer. Non-volatile memory for fast, reliable file systems. In *ASPLOS*, pages 10–22, October 1992.

[2] C. Brown. Microprobes promise a new memory option. *EE Times*, pages 6,41,44, January 12 1998.

[3] P. Chen and D. Patterson. Maximizing throughput in a striped disk array. In *International Symposium on Computer Architecture*, pages 322–331, June 1990.

[4] Rick Colson. Sorting disk blocks to reduce load times. Personal Communication, Intel Corporation, 1999.

[5] G. Ganger, B. Worthington, and Y. Patt. The disksim simulation environment version 1.0 reference manual. Technical Report CSE-TR-358-98, The University of Michigan, Ann Arbor, February 1998.

[6] J. Gray. What happens when processing, storage, and bandwidth are free and infinite. In *IOPADS Keynote*, November 1997.

[7] J. Howard, M. Kazar, S. Menees, D. Nichols, M. Satyanarayanan, R. Sidebotham, and M. West. Scale and performance on a distributed file system. *ACM TOCS*, 6(1):51–81, February 1988.

[8] J. Griffin, S. Schlosser, G. Ganger, D. Nagle. Modeling and scheduling of mems-based storage devices. Technical Report CMU-CS-00-100, Carnegie Mellon University, November 1999.

[9] J. Katcher. Postmark: A new file system benchmark. Technical Report TR3022, Network Appliance, October 1997.

[10] L. R. Carley, J. A. Bain, G. K. Fedder, et al. Single chip computers with mems-based magnetic memory. In *44th Annual Conference on Magnetism and Magnetic Materials*, November 1999.

[11] David A. Patterson and John L. Hennessy. *Computer Architecture: A Quantitative Approach*. Morgan Kaufmann Publishers, Palo Alto, California, 2nd edition, 1996.

[12] E. Pugh. Storage hierarchies: Gaps, cliffs, and trends. *IEEE Transactions on Magnetics*, pages 810–814, December 1971.

[13] A. Reddy and P. Banerjee. An evaluation of multiple-disk I/O systems. *IEEE Transactions on Computers*, 38(12):1680–1690, December 1989.

[14] M. Rosenblum, S. Herrod, E. Witchel, and A. Gupta. Complete computer system simulation: The simos approach. *IEEE Parallel & Distributed Technology*, 3(4), Winter 1995.

[15] J. Schindler and G. Ganger. Automated disk drive characterization. Technical Report CMU-CS-99-176, Carnegie Mellon University, November 1999.

[16] Seagate Technology. St-34501w (cheetah 4lp family). http://www.seagate.com/support/disc/specs/st34501w.shtml, September 1997.

[17] J. Wilkes, R. Golding, C. Staelin, and T. Sullivan. The HP AutoRAID hierarchical storage system. In *15th ACM SOSP*, pages 96–108, December 1995.