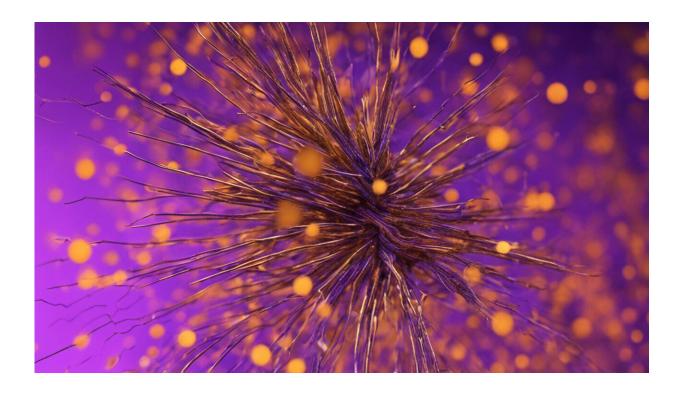# New network design exploits cheap, power-efficient flash memory without sacrificing speed

July 10 2015, by Larry Hardesty

Credit: AI-generated image (disclaimer)

Random-access memory, or RAM, is where computers like to store the data they're working on. A processor can retrieve data from RAM tens of thousands of times more rapidly than it can from the computer's disk drive.

But in the age of big data, data sets are often much too large to fit in a single computer's RAM. The data describing a single human genome would take up the RAM of somewhere between 40 and 100 typical computers.

Flash memory—the type of memory used by most portable devices—could provide an alternative to conventional RAM for big-data applications. It's about a tenth as expensive, and it consumes about a tenth as much power.

The problem is that it's also a tenth as fast. But at the International Symposium on Computer Architecture in June, MIT researchers presented a new system that, for several common big-data applications, should make servers using flash memory as efficient as those using conventional RAM, while preserving their power and cost savings.

The researchers also presented experimental evidence showing that, if the servers executing a distributed computation have to go to disk for data even 5 percent of the time, their performance falls to a level that's comparable with flash, anyway.

In other words, even without the researchers' new techniques for accelerating data retrieval from flash memory, 40 servers with 10 terabytes' worth of RAM couldn't handle a 10.5-terabyte computation any better than 20 servers with 20 terabytes' worth of flash memory, which would consume only a fraction as much power.

"This is not a replacement for DRAM [dynamic RAM] or anything like that," says Arvind, the Johnson Professor of Computer Science and Engineering at MIT, whose group performed the new work. "But there may be many applications that can take advantage of this new style of architecture. Which companies recognize: Everybody's experimenting with different aspects of flash. We're just trying to establish another

point in the design space."

Joining Arvind on the new paper are Sang Woo Jun and Ming Liu, MIT graduate students in computer science and engineering and joint first authors; their fellow grad student Shuotao Xu; Sungjin Lee, a postdoc in Arvind's group; Myron King and Jamey Hicks, who did their PhDs with Arvind and were researchers at Quanta Computer when the new system was developed; and one of their colleagues from Quanta, John Ankcorn—who is also an MIT alumnus.

Outsourced computation

The researchers were able to make a network of flash-based servers competitive with a network of RAM-based servers by moving a little computational power off of the servers and onto the chips that control the flash drives. By preprocessing some of the data on the flash drives before passing it back to the servers, those chips can make distributed computation much more efficient. And since the preprocessing algorithms are wired into the chips, they dispense with the computational overhead associated with running an operating system, maintaining a file system, and the like.

With hardware contributed by some of their sponsors—Quanta, Samsung, and Xilinx—the researchers built a prototype network of 20 servers. Each server was connected to a field-programmable gate array, or FPGA, a kind of chip that can be reprogrammed to mimic different types of electrical circuits. Each FPGA, in turn, was connected to two half-terabyte—or 500-gigabyte—flash chips and to the two FPGAs nearest it in the server rack.

Because the FPGAs were connected to each other, they created a very fast network that allowed any server to retrieve data from any flash drive. They also controlled the flash drives, which is no simple task: The

controllers that come with modern commercial flash drives have as many as eight different processors and a gigabyte of working memory.

Finally, the FPGAs also executed the algorithms that preprocessed the data stored on the flash drives. The researchers tested three such algorithms, geared to three popular big-data applications. One is image search, or trying to find matches for a sample image in a huge database. Another is an implementation of Google's PageRank algorithm, which assesses the importance of different Web pages that meet the same search criteria. And the third is an application called Memcached, which big, database-driven websites use to store frequently accessed information.

Chameleon clusters

FPGAs are about one-tenth as fast as purpose-built chips with hardwired circuits, but they're much faster than central processing units using software to perform the same computations. Ordinarily, either they're used to prototype new designs, or they're used in niche products whose sales volumes are too small to warrant the high cost of manufacturing purpose-built chips.

But the MIT and Quanta researchers' design suggests a new use for FPGAs: A host of applications could benefit from accelerators like the three the researchers designed. And since FPGAs are reprogrammable, they could be loaded with different accelerators, depending on the application. That could lead to distributed processing systems that lose little versatility while providing major savings in energy and cost.

"Many big-data applications require real-time or fast responses," says Jihong Kim, a professor of computer science and engineering at Seoul National University. "For such applications, BlueDBM"—the MIT and Quanta researchers' system—"is an appealing solution."

Relative to some other proposals for streamlining big-data analysis, "The main advantage of BlueDBM might be that it can easily scale up to a lot bigger storage system with specialized accelerated supports," Kim says.

This story is republished courtesy of MIT News (web.mit.edu/newsoffice/), a popular site that covers news about MIT research, innovation and teaching.

Provided by Massachusetts Institute of Technology