New Processor Paradigm: V-IRAM

Patterson's Research Aims to Solve Bandwidth, Multimedia Issues

by Linley Gwennap

Dave Patterson, one of the early pioneers of both RISC microprocessors and RAID disk arrays, is not ready to rest on his laurels. His latest project at UC Berkeley involves a mixture of vector processing and DRAM on a single chip to create an ultrapowerful CPU for the coming multimedia age. If this work bears fruit, similar technology could appear in commercial products within five years. The characteristics of the proposed device are particularly suited to low-cost but media-rich handheld devices.

Patterson started with the idea of combining DRAM and logic on a single chip, a concept he calls IRAM, or intelligent RAM. This concept predates Patterson, of course; it is already popular in the graphics area, where companies including NeoMagic are building or designing chips that combine a graphics processor with a DRAM frame buffer. In addition, Mitsubishi has combined a general-purpose CPU with two megabytes of DRAM to create the M32R/D.

In a classic paradigm shift, the first products are merely combinations or variations of existing products. Patterson is already working on the next step: reimplementing the concept of a microprocessor using the new IRAM paradigm. His research rejects the current fetish for complex out-of-order designs in favor of the vector approach used in traditional supercomputers. Ironically, this out-of-favor technique may find redemption in a low-cost processor.

Characteristics of Embedded DRAM

The IRAM concept of integrating memory and logic on a single chip, which we call embedded DRAM (see MPR 8/4/97, p. 19), has several advantages. As with other types of integration, it reduces the number of chips in the system, allowing smaller and potentially less expensive products. Power consumption is also diminished, since the signals between the memory and logic are now entirely on chip and use far less current.

Embedded DRAM goes beyond other forms of integration by addressing a key bottleneck in many systems: memory bandwidth. By eliminating its dependence on pin count and PC-board traces, an on-chip memory bus can be wider and faster than an external memory bus. Embedded DRAM can easily increase memory bandwidth by four times or more over a traditional design. The faster, shorter bus can also improve memory latency, although only incrementally.

The improved bandwidth and latency are available only to the on-chip DRAM, however, which is limited in size to the amount that can fit on a chip. Adding a modest amount of logic to a mainstream 16-Mbit DRAM provides access to only 2 Mbytes of memory, far less memory than is in a typical PC or workstation today.

Due to the many differences between DRAM and logic manufacturing processes, embedded-DRAM chips are more expensive to build than either DRAMs or processors of the same size and type. The laws of IC manufacturing also state that one large die costs more to build than two small die, although in many cases the savings in package costs outweigh the extra silicon costs. To date, embedded-DRAM chips have carried a cost premium over discrete devices.

Opportunities for Embedded DRAM

These characteristics are uniquely well suited to notebook graphics accelerators, the first area successfully served by embedded DRAM. Notebook makers are willing to pay a premium to reduce board space and power consumption. For best performance, graphics accelerators require high-bandwidth frame buffers, but 2 Mbytes is an adequate size. NeoMagic (see MPR 3/6/95, p. 20) was the first company to take advantage of this concept, and others have followed.

Handheld computers (including high-end organizers, PDAs, and smart cell phones) match many of the embedded-DRAM characteristics. A small footprint and low power are critical in these devices, but they are more cost-sensitive than notebook PCs. One or two megabytes of DRAM is enough for many handheld devices, but high bandwidth usually isn't critical. The M32R/D (see MPR 5/27/96, p. 10) is intended for such designs but has had little success there. This market is small today but growing rapidly, creating many opportunities for future products.

The next best opportunity is in low-cost line-powered devices, ranging from set-top boxes to disk-drive controllers. These products don't need the lower power consumption of embedded DRAM but do benefit from the cost advantages of highly integrated devices. They typically require a relatively small amount of memory and, in cases such as video games, could take advantage of high memory bandwidth.

PCs and other general-purpose desktop systems are least likely to use embedded DRAMs. While they can use plenty of bandwidth, these systems require more memory than can fit on a single chip. Even with 64-Mbit DRAMs, a low-end PC requires at least four chips; it will be years, if ever, before a single DRAM will hold enough memory for even an entry-level PC.

One possibility is using the on-chip DRAM as cache and retaining external DRAM for main memory. The latency of the on-chip DRAM is too long for it to act as a primary cache, but it could provide a large on-chip secondary cache backing small, fast primary caches built from SRAM. A

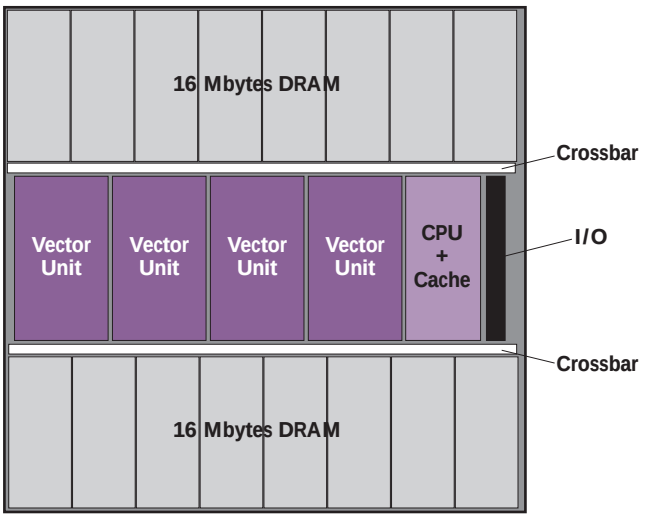

Figure 1. The tentative floorplan for V-IRAM-1 shows a 256-Mbit DRAM bisected by four vector processors and a small scalar CPU. The DRAM is connected to the vector units via a ring-based crossbar switch. Patterson estimates the die size will be 256 mm^2 in a 0.18-micron five-layer-metal process.

DRAM cache would provide up to eight times more storage than an SRAM-based secondary cache in the same area.

Embedded DRAM Demands New CPU Designs

Grafting an existing processor onto a DRAM chip provides some useful advantages, but this method doesn't take full advantage of the power of embedded DRAM. Inside a typical DRAM, a row access returns thousands of bits, but only a few are sent off-chip. With a row-access time of 20 ns, a single DRAM can generate an internal bandwidth of more than 100 Gbytes/s, three orders of magnitude faster than the memory system in a typical PC and two orders faster than the level-two cache interface in today's Pentium II.

Certainly, memory bandwidth is increasingly a bottleneck to processor performance. But improving bandwidth by orders of magnitude doesn't just break this bottleneck, it obliterates the entire container, demanding a complete rethinking of the processor function. Instead of focusing on increasing the speed of a linear set of calculations, one must focus on making the best possible use of the enormous bandwidth available from the embedded DRAM array.

Patterson's solution is to revive the concept of vector processing by performing simultaneous operations on a large set of data. Figure 1 shows a processor he is developing called V-IRAM (vector IRAM) that includes four vector units, each capable of processing two 64-bit operands per cycle. At 500 MHz, these units would consume 32 Gbytes/s of data and generate 16 Gbytes/s of results. Although the total is just a fraction of the available internal DRAM bandwidth, it is still far better than any modern CPU can achieve.

Peak performance at this rate would be 4 GFLOPS (single-precision floating-point operations) or 16 GOPS (8-bit integer operations), since the vector units can be configured to operate on small data items in parallel. These ratings may be matched by traditional CPUs in the next year or two, but only when operating on data in registers. The key advantage of V-IRAM is its ability to sustain this rate for any vectors that can fit into its 32M of on-chip DRAM.

Truly a Cray on a Chip

A good example of vector processing is the traditional Cray supercomputer. These systems were incredibly expensive, in part because the entire memory subsystem was constructed from megabytes of high-speed SRAM to provide the bandwidth needed by the vector units. V-IRAM reduces this entire system, both memory and vector units, to a single chip. By moving the vector memory on chip, high-density DRAM can replace the fast SRAM with no loss of bandwidth.

These vector machines were used for high-end scientific calculations, which typically apply repetitive operations to large data sets. Vector systems are less suited to traditional software that operates linearly with many branch points. Thus, vector processing would make little sense for many of today's mainstream software applications, even if an entire Cray could be placed onto a single chip.

The emergence of multimedia applications changes this situation. These applications are similar to their scientific forebears (indeed, they use many of the same algorithms) in their large vector size and high demand for memory bandwidth. For example, image-processing algorithms such as discrete cosine transform (DCT) and motion estimation can process an entire row of pixels at once, typically 100 to 1,024 pixels depending on image width. A fast Fouriér transform (FFT), frequently used in audio processing, can process 256 to 1,024 samples at once.

Most of the programs that will require high CPU performance in the future are multimedia applications well suited to vector processing. A reasonably fast scalar processor, which Patterson includes on V-IRAM, should be able to handle basic data-processing needs while the vector units create a lively user interface.

Advantages of V-IRAM

V-IRAM has several advantages, particularly over time. During the past 20 years, vectorizing compiler technology has become mature and widely available. In contrast, compilers for the next-generation EPIC architectures (e.g., IA-64) will be exceedingly complex and have yet to be deployed.

The V-IRAM hardware will also be much easier to design than other high-performance processors. As Figure 1 shows, the chip consists of a 256-Mbit DRAM, a simple vector unit (replicated four times), a scalar processor core (which could be licensed from MIPS or ARM), and some crossbar and I/O logic. Patterson plans to design the chip in two years with a small team of graduate students. Intel's Merced, in contrast, will take hundreds of engineers more than three years, due to the large amount of complex custom logic that must be implemented.

The basic V-IRAM design can be scaled upward or downward simply by adding or removing vector units and banks of DRAM. To improve performance, an EPIC or RISC processor must be completely redesigned for greater parallelism, more instruction reordering, and better memory bandwidth. The internal bandwidth of the V-IRAM chip is more than enough for future growth and can be easily increased by adding more internal memory banks.

As transistors become smaller, wire delays are becoming the critical component of processor speed (see MPR 8/4/97, p. 14). In V-IRAM, data can pass from one bank of DRAM to the nearest vector unit for processing, minimizing the physical distance signals must travel. Few other designs can collocate the data and processing units in this fashion.

Although the V-IRAM chip may be somewhat large, manufacturing yields should be relatively high, reducing cost. Patterson estimates the configuration in Figure 1 will measure 256 mm^2 in a 0.18-micron CMOS process, but redundant columns in the DRAM arrays make nearly two-thirds of the chip essentially immune to defects. Because the vector units are all identical, a future version of the chip could contain an extra one, protecting up to 90% of the die though redundancy.

Manufacturing costs would be reduced further by selecting a low-pin-count package. Assuming the application requires no external DRAM, the only interfaces would be to low-speed peripherals and ROM. Patterson proposes a single 8-bit bus running at 250 MHz to connect to these devices. Most other processors will require much wider buses to external DRAM, increasing package cost.

Keeping the memory interface entirely on-chip reduces power. Patterson estimates the first V-IRAM chip, despite containing 270 million transistors, will dissipate about 10 W at 500 MHz. While this is substantial, competitive RISC and EPIC processors are likely to dissipate more than 30 W, not including external cache or DRAM. V-IRAM also saves power by avoiding speculative execution; power is consumed only for necessary calculations.

Patterson believes power can be further reduced to about 2 W by slowing the clock to 200 MHz and lowering the supply voltage accordingly. This power level would be suitable for many portable applications, and the chip would still deliver 1.6 GFLOPS and 6.4 GOPS.

Enabling the Ultimate PDA

In this low-power mode, a V-IRAM processor could be the engine for the ultimate PDA. Such a device could encompass the functions of a Palm Pilot (organizer), Game Boy (entertainment), digital camera, pager, and cellular telephone. The V-IRAM chip would be powerful enough to handle speech recognition for the organizer, graphics for the video game, video processing for the camera, and signal processing for the cell phone, all with modest power consumption. Of course, the chip could also be used in any variant of this concept that might become popular.

For More Information

Contact Dr. Patterson at *patterson@cs.berkeley.edu* or access the Web at *iram.cs.berkeley.edu*.

Patterson also envisions applications at the other end of the scale: massive data-processing systems. In an increasingly popular technique known as data mining, each record in an enormous database is evaluated according to often-complex criteria to determine the best match. Today, this requires a single computer connected to a large disk farm; the central processors each fetch one record at a time and evaluate it.

By placing a single low-cost V-IRAM in each disk drive, data evaluation could instead be performed at the data source. Each drive could process a series of records in parallel with the others, forwarding only those that match the criteria. This would ease the bottleneck of transferring all records from the disks to the central processors. This design also greatly increases the number of processors in the system without the complexities of symmetric multiprocessing.

Changing the Business of Processors

Should Patterson's vision come to pass, changes will not be limited to technology. If processors and DRAM are merged into one chip, DRAM makers must develop CPU expertise, and vice versa, to remain successful. The server market could be turned literally inside out, with processing being handled by the peripherals.

V-IRAM also challenges the basic philosophy of EPIC, the cornerstone of Intel's future. If Patterson is right, PC processor makers should build V-IRAMs with an x86 scalar core and a massive vector unit rather than adopting the complex IA-64 instruction set.

Like EPIC, V-IRAM remains unproved. To date, embedded DRAM efforts have resulted in either slow logic or poor memory density; chip vendors are working on improved processes to ease this disparity, but higher costs are likely to remain.

As a microarchitecture, V-IRAM shows promise, but performance metrics on real applications are needed. If any critical application develops that requires high performance but is not vectorizable, V-IRAM will suffer. Given Intel's interests, such applications are likely to emerge in the PC space. V-IRAM will also be hampered by the need for multiple memory chips in typical PCs and workstations.

For low-cost consumer devices, particularly portable ones, V-IRAM could be an ideal fit, delivering strong performance on a limited but media-rich set of applications without breaking the cost or power budgets. Patterson hopes to demonstrate a working V-IRAM chip by mid-2000. If this research is as successful as his RISC and RAID work, the first commercial V-IRAM chips could appear by 2003. Given Patterson's track record, don't bet against him. M