A Swing of the Pendulum?

Future Architecture Changes Unlikely to Improve Performance

by Maurice Wilkes

For some time, there have been mounting signs of uncertainty in the microprocessor world. New forces appear to be at work, and subtle changes are occurring in the climate of opinion. This is true across the range, including processors and their architecture, benchmarks for both processors and software, and simulation. We may have reached one of those periods when we need to take cherished ideas out of our heads, and either discard them or at least dust them off.

At the end of the 1980s, effective benchmarks were developed, and it suddenly became possible to make comparative measurements of processor performance. The result was to focus attention on the processor rather than on the system. Up to that time, computers had been sold on system performance and on the quality of vendor support. No one knew, with any precision, how different vendors' processors compared, nor did they care. Not that customers took no interest in instruction sets; they did, but they judged them like pet dogs for their appearance and pretty points, not like racehorses for their speed. Benchmarking came in the wake of a great boom in simulation. No longer was it necessary to build a processor in order to evaluate an architecture. Given a room full of the fastest machines available, the architecture could be simulated. A great triumph of simulation was to establish beyond any doubt that RISC processors were faster than existing processors by a factor of approximately two.

However, what really gave RISC its chance was not its high speed, but its small size. A RISC processor needed only half as much silicon real estate as a conventional processor. Real estate was still in very short supply, although the situation was improving steadily. By 1989, room could be found on a single chip for a RISC integer processor, an MMU, a TLB, and cache control circuits. The MIPS R2000/3000 processor, designed on these principles, turned out to be fast enough to outperform the fastest minicomputer that DEC then made. Without RISC, this could not have happened for another two years—a conventional processor would by itself have taken up the entire available area, leaving no room for the MMU and other items. These two years were crucial in enabling the Unix workstation based on a RISC processor to establish itself in the market place.

The Unthinkable Happens

It was not long before people stopped buying VAXs and other minicomputers, and began to buy RISC workstations instead. A short time before, most people would have said that the VAX architecture was so firmly entrenched that its demise was unthinkable. But the unthinkable happened, and with so little fuss that the event was hardly noticed. The IBM 370 architecture was similarly hit, although it survived in mainframes.

Silicon real estate is still very expensive—according to Gordon Moore, it sells at a billion dollars per acre. You can still only buy 50 nano-acre lots, but the number of transistors that you can put on a single lot is going up and up. No-one worries any more about real estate as far as the integer processor is concerned.

The minicomputers that the R3000 challenged were based on bipolar circuitry. Thus was another article of faith shattered that, while CMOS was one for personal computers, "real" computers would always be bipolar. Because of this fundamental difference in underlying technology, the personal computer side of the industry had developed separately from the main industry. Now that the difference no longer existed, it was inevitable that PC manufacturers should seek a share of the workstation market. The 486 brought this within sight. The Pentium Pro came very near to the speed of the best RISC chips. This was a notable success and, in order to achieve it, Intel pulled out all the stops known to their chip designers and process engineers.

It is possible that Intel was not confident of being able to repeat this success for smaller feature sizes. Perhaps this is why they have teamed up with HP in the Merced initiative. On the other hand, it may have been a decision based primarily on marketing considerations. At all events, they evidently feel that a design combining RISC and x86 features is what is required. They have chosen to develop an entirely new design, instead of starting, as they might have done, with a proven RISC design, such as the Alpha. They have stated that Merced will be binary compatible with x86. The performance that can be delivered at this level will clearly play an important role in determining the future direction of events.

Future Innovation

In the recent past, increases in processor speed have come partly as a result of architectural improvements and partly as a result of shrinkage. I would be surprised if we were to see any further architectural improvements of a major kind; indeed, most of those that have recently found their way into silicon were foreshadowed, if not fully exploited, in the large mainframes of the 1970s. In my view, any further significant increases in the speed of uniprocessors will come from shrinkage alone.

It is hard to evaluate the Merced initiative. Judging by

the information that has emerged, the Merced team has not been able to come up with any major architectural innovation. If what I have just said is correct, this is not surprising. It may be that there is unexpected mileage to be obtained from old ideas such as VLIW (very long instruction word) and predicated execution. A critical factor will be the ability of innovative compiler techniques to expose instruction-level parallelism on a much greater scale than has hitherto been possible. This would be a notable breakthrough. There is an obvious danger that the combined hardware/software complexity of the system will defeat the ends aimed at.

I suspect that we may be seeing a natural swing of the pendulum. Old concerns, such as instruction-set compatibility, are reasserting themselves and the breed fanciers are making their voices heard again. I think it probable that we shall see less attention paid to small factors in processor speed. This will provide a favorable climate for processors with instruction sets compatible with that of Pentium II to make further inroads into the workstation market. However, I do not see RISC workstations being driven out altogether.

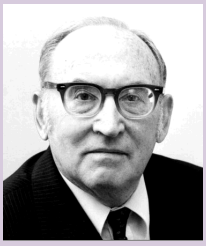
Performance of Software Systems

I would like to feel that in the future more attention will be paid to system performance as distinct from processor performance. For this purpose, we need effective benchmarks for software systems.

The benchmarking community is already devoting part of its effort to software benchmarks. I observed this recently when I had the privilege of sitting in on a day's discussions in the SPEC Open Systems Group. In hardware, the breakthrough came with the realization that it was strongly in the interests of competing vendors to have agreed upon benchmarks that were as fair and as proof against cheating as human ingenuity could make them. Once the rules of a contest are defined and seen to be fair, tension is reduced. This is well illustrated by the age-old procedure for dividing an apple between two people: one cuts and the other chooses. In the processor field, a common interest made it possible for otherwise competing companies to collaborate harmoniously in the development of benchmarks that could form the basis for subsequent competition.

The same common interest is there in the case of software benchmarks, but the technical problems are much greater. It is easy to be over-optimistic. Performance measurement thrives where there is competition, and languishes where there is monopoly. Perhaps the greatest hope lies in new areas, such as Java, where competition is fierce. Already, SPEC has a Java-client performance benchmark under active development.

The hardware side of the computer industry has had a severe shake-up in recent years. I would not like to see the software side suffer a similar fate, but a milder shake-up would undoubtedly be for the general good. MP



Maurice Wilkes, born in 1913, was for many years head of the Computer Laboratory of the University of Cambridge, where he designed and constructed the EDSAC computer. He was a pioneer of programming for stored-program computers and of microprogramming. He wrote the first paper on cache memories and was an early worker in wide-bandwidth local-area networks.

Now with Olivetti Research, Wilkes is a Distinguished Fellow of the British Computer Society and a Fellow of the Royal Society and of the Royal Academy of Engineering. He is a Foreign Assoicate of the U.S. National Academy of Science and of the National Academy of Engineering.

In 1967, Wilkes delivered the ACM Turing Lecture. He was the recipient of the 1980 Eckert-Mauchly Award and the 1992 Kyoto Prize for Advanced Technology. He has written a number of books, including the first book on computer programming, published in 1951.

For more biographical information, access the Web at *www.cs.stevens-tech.edu/~nar/614/Wilkes.html*.

For more information on modern high-performance CMOS processors, see the author's book *Computing Perspectives* (Morgan-Kauffmann, 1995).