Microprocessor Developments Beyond 1995

Experts See Limits to Superscalar Benefits—Memory Becomes Paramount

By Linley Gwennap

At the recent Microprocessor Forum, a panel of six industry experts took out their crystal balls to divine the state of microprocessor technology in the second half of this decade. They consistently warned that current techniques such as superscalar design and on-chip caches are reaching their limits. Several panelists believe that future performance increases will require new computing paradigms. As expected, the RISC vendors think that x86 chips will continue to fall further behind on the performance curve, while the x86 representatives assume that their chips will continue to close the gap and eventually match RISC performance levels.

The limits of superscalar designs came up quickly and repeatedly. Intel's Pat Gelsinger commented that:

> "Two-issue machines quickly bring us to the end of what basic block parallelism allows. There just isn't much more parallelism to garner in there.…As such, to move beyond that, you've got to go to out-of-order techniques and good branch prediction to garner any more of that parallelism. Even with these, however, there isn't much more parallelism to get in the basic integer stream."

Michael Mahon of Hewlett-Packard agrees, pointing to an ASPLOS paper by Jouppi and Wall (see µPR 9/19/90, p. 13). After acknowledging the problem, he points to a possible solution.

> "We are going to see uniprocessor performance continue at an 80% per year pace up to around the middle of the decade. Then, for a number of reasons, but particularly because of this problem in single-thread parallelism, we are going to see the rate of increase be cut to essentially the rate of increase we are accustomed to because of [IC] technology advances.…That is going to push us to go MP, whether we like it or not."

Alpha architect Dick Sites also sees single-chip multiprocessors in the future.

> "When it's cheap to have lots of functional units on a single die, one of the techniques for keeping those units busy is to have more than one program counter, and so you will end up, I believe, by the end of the decade seeing single chips that effectively are two-, three-, or four-way [multi-] processors."

In order to utilize these multiprocessor systems efficiently, advances must be made in software as well as hardware. Mahon gave an example of the problem with current software:

> "The payroll department used to [have] 50 or 60 clerks running around with filing cabinets and checkbooks, and it all worked fine. They didn't have a lot of the blocks to parallelism that we have in the COBOL program that replaced them, because the COBOL program is a big loop with a CASE statement inside of it. Now, we look at it and say, 'Gee, this isn't very parallelizable,' but, in fact, it's really wildly parallelizable if we go back to the problem."

Memory Issues

Several of the panelists were concerned that, even though processor speeds will continue to increase dramatically through the rest of the decade, memory bandwidth will not keep pace. Even worse, memory latency will probably improve very little, although memory sizes will continue to skyrocket. This trend does not bode well for processor performance. Sites detailed the problem:

> "If you're running a CPU cycle every 2 ns (500 MHz) and you're issuing four instructions in that cycle, you can't afford to wait very long for an off-chip memory reference. If you wait ten cycles, you've lost 40 issue slots, and ten cycles is only 20 ns."

Sun's Dave Ditzel elaborated, pointing out that new software techniques make the problem even worse:

> "All of your time is going to be spent taking cache misses, doing things with large, distributed, object-oriented, global databases with persistent objects. Let me tell you, if you take a cache miss on one of those and have to go across the building to fetch something, your average clocks per instruction is going to suffer quite a lot compared to what your peak rate could be.…I think you are going to see people simply working on better memory systems, so as not to be losing so much [performance] elsewhere in the system."

Perhaps we should be working on changing the way we write software as well. Mahon gave an example:

> "Right now, inside operating systems, we are mak-

> ing a lot of use of data structures called linked lists. No one would build a linked list on disk, but it's getting to be about like that [with the increase in relative memory latency]. So that's not a very clever idea any more."

So how will the hardware designers work on the memory problem? Sites sees some potential in the Rambus approach, but that improves only bandwidth and not latency. Mahon has another idea:

> "One of many possible ways to put a system onto a chip [is to combine a] 64 Mbit DRAM [and] about 256 Kbytes of cache; you add the DRAM controllers; the processor recedes into an insignificant dark corner of the chip; and you get video, high-speed networking, all that neat stuff."

If memory bandwidth becomes a major issue, watch out for the x86! As Mark Bluhm of Cyrix points out:

> "To achieve [a high] issue rate, the bandwidth required from the instruction cache could become prohibitive. This is where I see that the x86's variable length instructions, requiring an average of 2.5 bytes per instruction, will help minimize this problem. Also, as we have seen before with the DX2-type products, the disparity between the clock speed of the processor core and the bus…will cause the cache miss penalties to increase. This is where code density and average operand size will become a major factor in minimizing the cache miss rate."

The 1996 Microprocessor

The six panelists were in surprising agreement on the general parameters of the typical high-end microprocessor at the middle of the decade. They see a CPU running at 250–300 MHz with about 64K of on-chip cache and a maximum issue rate of 4–6 instructions per cycle. All of these numbers represent about a doubling from the leading microprocessors available today, so they could be considered conservative estimates. They also agreed that the sustainable instructions per cycle (IPC) on integer code would be about 2–2.5 for these processors.

Sites disagreed slightly. Perhaps reflecting Digital's emphasis on higher clock rates, he envisions a 400–500 MHz processor available at mid-decade. He also sees much larger on-chip caches, 256K or greater, on these future chips. This projection is probably at the high end of what could be accomplished by 1996, even with DEC's leadership in IC processes.

When prodded, each architect admitted that there were features of their architecture that they would like to change. John Mashey, from SGI, thought the R4000 transition would have been smoother if the original MIPS architecture had 64-bit floating-point registers instead of 32-bit ones. Ditzel would remove delayed branches, which have not been included in newer RISC architectures like POWER and Alpha, but still believes that SPARC's register windows "have some unique value."

Mahon would "throw out strongly ordered memory references." Again, this is a feature that newer designs (Alpha and SPARC v9) have not included, and perhaps we will see it eliminated from a future version of PA-RISC. Sites just finished designing Alpha, but he opined that he might drop VAX floating-point support "by the end of the decade." Bluhm has a different problem:

> "Since [Cyrix] implements x86, my problem is just the opposite. I've got so many things to throw away, I don't know really what to choose from. I think probably the one thing that is hardest to overcome is the small register set."

Gelsinger agreed about the register set and added that, because x86 instructions have a variable length, decoding them is "really nasty…but x86 isn't out to win beauty contests, it's out to win the architecture war."

No one foresaw any totally new architectures being successful in the rest of the decade. Instead, they expect the current architectures to evolve over time with changes similar to those just described. Of course, this response is just what one would expect from representatives of today's major architectures.

The Ante Goes Up

The panelists discussed the cost of developing new microprocessors. Some think the price is going up. When asked at what point companies would find it difficult to afford the development costs of new microprocessors, Intel's Gelsinger quipped:

> "I'll argue that almost everybody else on this stage is already past that point.…We're talking about half-billion dollar fabs. The technology development turns are going to be measured in the hundreds of millions of dollars. All of these [IC] processes are hard; they're all going to be pretty expensive to build. So the economies of playing in this business are going to be *the* factor in the second half of this decade."

Although everyone agrees that the cost of IC fabrication facilities is going up, these costs can be spread across other chips as well as microprocessors. Furthermore, these future fabs will have a higher capacity than current fabs due to larger wafers, and a much higher capacity when measured in actual transistors produced, since the transistors will be much smaller. Thus, the actual per-chip fabrication cost may not

(L to R) Pat Gelsinger, Mark Bluhm, Dick Sites, Michael Mahon, Dave Ditzel, John Mashey, moderator Michael Slater.

increase dramatically, although the up-front investment to own a fab will.

The cost to design a modern microprocessor is not increasing at the same rate as fab costs. As Sites put it:

> "The per-chip design effort seems to me to be either about constant or very slowly growing. It is bounded by the fact that you need to finish about every couple of years."

Mahon pointed out that system vendors such as DEC and HP have a different perspective than chip vendors like Intel and Cyrix.

> "A system manufacturer is not trying to leverage chip sales to recoup the development cost, so there is a substantial chunk of system revenue that's being leveraged."

Several panelists said, perhaps wishfully, that the number of microprocessor architectures would be cut down over the course of the decade as companies could no longer afford the development costs, or at least the cost of building a fab. Owning a fab can make a crucial difference; the highest-performance RISC vendors (DEC, HP, and IBM) and CISC vendors (Intel) all have their own fabs. Cyrix's Bluhm admitted that their designers are constrained by not having access to the latest fabrication technology, but this has not stopped companies like Sun and SGI/MIPS from having success in the RISC market.

Other Trends

Although most RISC vendors are busy pushing up the clock rate of their chips, resulting in ever-increasing power requirements, most of the action in the PC market is in low-power, portable systems. Of the RISC vendors, Sun has been most active in the portable area, and Ditzel believes this will be important throughout the decade.

> "I think the large majority of microprocessors built in the last half of the decade will be 'cold,' because I think they're going to go into portable computers, things with very small footprints, and these are the kind of machines that will dominate. Battery life is really going to be the driving factor for how we design microprocessors."

Sites acknowledged that lack of software has kept RISC chips from reaching the volumes of the x86. He sees the possibility for a change in the '90's.

> "I think Windows NT is a real wild card. Windows NT can run on machines other than x86s and run large amounts of popular software. That might end up breaking the software monopoly for x86s."

Conclusions

Although microprocessor performance should continue to increase at its current pace for the next few years, these increases may be reduced after the middle of the decade. The benefits from superscalar issue may top out at around 4–6 issue machines. The benefits of larger on-chip caches decline as these caches reach 64K and beyond. Even clock frequency improvements will be harder to come by as future IC processes trade off faster transistors against lower voltages. To break these barriers, multiprocessing will be used in all types of systems.

Looming over the processor issues is the widening gap between processor speed and memory latency. Future software will be increasingly object- and image-oriented, exacerbating the memory issues by reducing the effectiveness of caching. Objects reduce locality by scattering memory references, while large images simply overwhelm most caches.

Software can help as well as hinder. New compiler algorithms may increase the amount of instruction-level parallelism, improving the efficiency of highly superscalar processors. A different solution could involve parallelizing compilers that spread a single task over multiple processors. Unforeseen breakthroughs could occur in these or other areas that burst the mid-decade bottlenecks and allow unrestricted performance increases into the next century. ♦