Cahners
MICROPROCESSOR REPORT
www.MPRonline.com
THE INSIDER'S GUIDE TO MICROPROCESSOR HARDWARE

THE EDITORIAL VIEW

Are There Too Many Processors?

By Keith Diefendorff {2/28/00-01}

Like rabbits, microprocessors proliferate at an astounding pace. Every day, it seems, a new architecture, a new processor, or a new core springs onto the scene. While the overpopulation provides a great source of food for *Microprocessor Report*, and while proposing a new microprocessor can be an effective way of conning money from venture capitalists or corporate bureaucrats, an argument can be made that too much engineering time is spent chasing down empty holes.

This phenomenon is most obvious in the embedded-processor space, where the count of different processors and cores has to be kept as a floating-point variable. But to only a slightly lesser degree, the phenomenon exists in the computing space as well. In either domain, it's hard to argue that the world really needs all the architectures and processors that exist. Luckily, we have Darwinian market forces and a finite food source to keep the population from going completely out of control. But natural selection is a slow, inefficient process. Surely the world would be a better place if all the engineering effort wasted on hordes of nearly identical processors could be channeled in a more productive direction. Isn't it time to start treating processors like reusable building blocks, rather than repeatedly redesigning the blocks each time?

Any number of excuses can be, and are, used to justify a new processor. Would-be product groups like to make the argument that none of the existing alternatives will do the job. Another popular rationalization is that starting with a clean slate will magically result in a product so superior that the world will beat the proverbial path to the company's door. Not in all cases, but certainly in many, these arguments are specious and self-delusional. History is absolutely clear that in a high percentage of cases, companies would have been further ahead using an existing design. A new design—once it finally makes it to market—is all too often no better, or only marginally better, than those it was designed to replace. In only a few cases can a company look back and really justify the additional effort it took to design a new architecture, processor, or core. Surely some unidentified force is at work that causes companies to repeatedly overestimate what they can achieve by starting anew and chronically underestimate the costs of doing so.

History is replete with examples of such miscalculations. Probably the most blatant case is that of RISC architecture. In the late 1980s, when RISC made its debut, it appeared to offer huge advantages over existing CISC architecture. Proponents, mistakenly it turns out, thought they could exploit these advantages to displace entrenched CISCs in computers. Obviously, RISC backers overestimated the potential of their technology and underestimated both the size of the software hurdle and the progress that would be made by CISC competitors. In a huge miscalculation, RISC proponents badly underestimated the impact of exponential growth in transistor budget, which rendered one of RISC's most touted advantages, transistor efficiency, all but moot.

One cannot help but draw a parallel between the RISC example and the path being taken by HP and Intel on IA-64.

Maybe IA-64 will be superior to superscalar RISCs, maybe not. But it is hard not to wonder whether placing the same effort behind the Alpha architecture, for example, might not have produced results that were as good or even better in less time. There are many other questionable examples. Is Motorola's M•Core architecture really all that much better for cell phones than the ARM architecture used in its current phones? Are DSPs really faster, smaller, or less power-hungry than RISCs? Is there really a need for a half-dozen similar-but-different VLIW DSP architectures? Probably not. It's just that many companies seek the same market, and each—probably incorrectly—believes that doing its own thing is best way to capture that market.

In my opinion, every company contemplating a new processor, and certainly a new architecture, should rethink its decision, secure in the knowledge that the outcome is likely to be far less compelling than it expects, and the costs—both hidden and real—will be far greater than it is anticipating. Most likely, the company would be better off using an existing architecture as is, adding a few new instructions, pushing a current processor into a new IC process, turning on compiler optimization, or rewriting the inner loops of its applications.

But on a higher plane, it seems to me that the phenomenal progress made so far in microprocessors may actually be due to this irresistible urge to redesign. Perhaps it's the thousands of companies struggling against each other, each heading in a seemingly random direction, that ends up producing the large collective gains we see over time. After all, the speed with which evolution creates more advanced species increases with increasing birth rates; higher birth rates provide more opportunity for random mutations, some of which will be winners; from these successes, others learn and adjust, as if guided by an invisible hand. So, while in many individual cases designing a new architecture or processor is probably not the best course of action, in the bigger picture it may result in the best situation for everyone—*Microprocessor Report* included. ◇