Microprocessors for Consumer Electronics, PDAs, and Communications

Sept 26-30, 1999

Jim Turley
Senior Editor, *Microprocessor Report*
Editor in Chief, *Embedded Processor Watch*
Contributing Editor, *Embedded Systems Programming*

http://www.MDRonline.com/embedded

ALL VOLUME, NO SEX

PCs and workstations may get all the press and the glory, but embedded microprocessors ship in far greater numbers. In fact, embedded microprocessors account for approximately 100% of worldwide microprocessor volume! The number of PCs, Macintoshes, workstations, servers, minicomputers, and mainframes is less than 1% of the number of embedded CPUs that ship every year. After all, how many computers do you own? Compare that to the number of electronic consumer items—VCRs, telephones, televisions, thermostats, appliances, etc.—and you begin to see the bigger picture. PCs may get 100% of the sex appeal, but embedded gets (approximately) 100% of the volume.

In 1998, almost 250 million embedded 32-bit microprocessors were shipped. That's more than double the number of PCs (at about 100 million), Macintoshes (around 3 million), and workstations (less than one million), in the same year. That's one 32-bit embedded microprocessor for every man, woman, and child living in the United States. And that's just 32-bitters. The numbers of 4-bit, 8-bit, and 16-bit chips are orders of magnitude higher.

I estimate that the average, middle-class household in North America has 40 to 50 microprocessors in it—55, if you own a PC. Actually, even that latter number is low because the PC itself includes a half-dozen smaller microprocessors in the hard disk drive, keyboard, mouse, modem, and graphics card, to name a few.

Low-end 4-bit and 8-bit microcontroller volume is measured in millions of units per month. Motorola, for example, shipped it two-billionth 68HC05 microcontroller in

April of 1997. Like any ecosystem, the smallest creatures appear in the greatest numbers. If 4-bit microcontrollers are the insects of the microprocessor food chain, 32-bit and 64-bit embedded microprocessors are the carnivores.

In economic terms, unit volume is inversely proportional to cost. That is, the 4-bit chips are more plentiful than the 8-bitters, which ship more volume than the 16-bit CPUs, and so on. That means that in the embedded market, 32-bit chips account for a relatively small slice of the overall embedded-microprocessor pie. Yet it is the 32-bit market that is growing the fastest. While 4-, 8-, and 16-bit growth rates are in single digits, the 32-bit market is growing by leaps and bounds. The year-over-year growth of 64-bit processors is ever greater.

What many observers fail to recognize is that many of these little-known embedded microprocessors can run circles around the more famous CPUs like Pentium and PowerPC . It seems counterintuitive, but it’s true: a Nintendo 64 video game console, that sells for just $129 (including power supply, case, joystick, memory, and video controller), is actually faster than a lot of Pentium PCs. Yet until recently, most Pentium processors alone sold for more than $150.

HUNDREDS OF CHIPS TO CHOOSE FROM

The sheer variety and quantity of microprocessors is surprising. Most people have heard of Pentium , PowerPC , and perhaps a few other microprocessors, like SPARC , Alpha , or MIPS . For desktop computers and workstations, there are only a few choices, and the options seem to be dwindling as Intel’s x86 architecture increasingly dominates the desktop.

But these chips represent just a tiny fraction of the total volume of microprocessors shipped each year. In the embedded world alone—just considering 32-bit more than 100 different microprocessors currently for sale, and that’s not counting all the different speeds grades or packaging options. More than 100 distinctly different microprocessors, representing more than a dozen instruction-set architectures and more than 30 different vendors worldwide. That’s 100+ choices that an engineer must face when selecting a 32-bit microprocessor for a new embedded application.

Presumably, each of these chips is in production for a reason; each one must have happy customers somewhere. And the fact is, they do. With more than 100 competing devices from more than two dozen vendors, many market watchers wonder: will there be an industry shakeout? Will the embedded-microprocessor market collapse, with the industry standardizing on just a few CPUs, the way desktop computers have?

The answer is: no. The sheer variety of embedded applications stretches the imagination. No single microprocessor could ever hope to address them all. And far from shrinking, the number and type of embedded applications is growing in areas like wireless communication, networking, printing, imaging, games, appliances, and personal productivity, thus providing even more market niches for new and innovative microprocessor designs.

EMBEDDED MARKET IS HUGE

Where do all these microprocessors go? The volume of microprocessors seems to be inversely proportional to their performance. That is, the lowliest 4-bit microcontrollers used in thermostats and the like account for thousands or even millions of times more volume than the fastest 64-bit chips used in engineering workstations. According to the market-research firm Information Architects, the market for 32-bit embedded microprocessors is divided roughly in thirds, as shown in Figure 1.

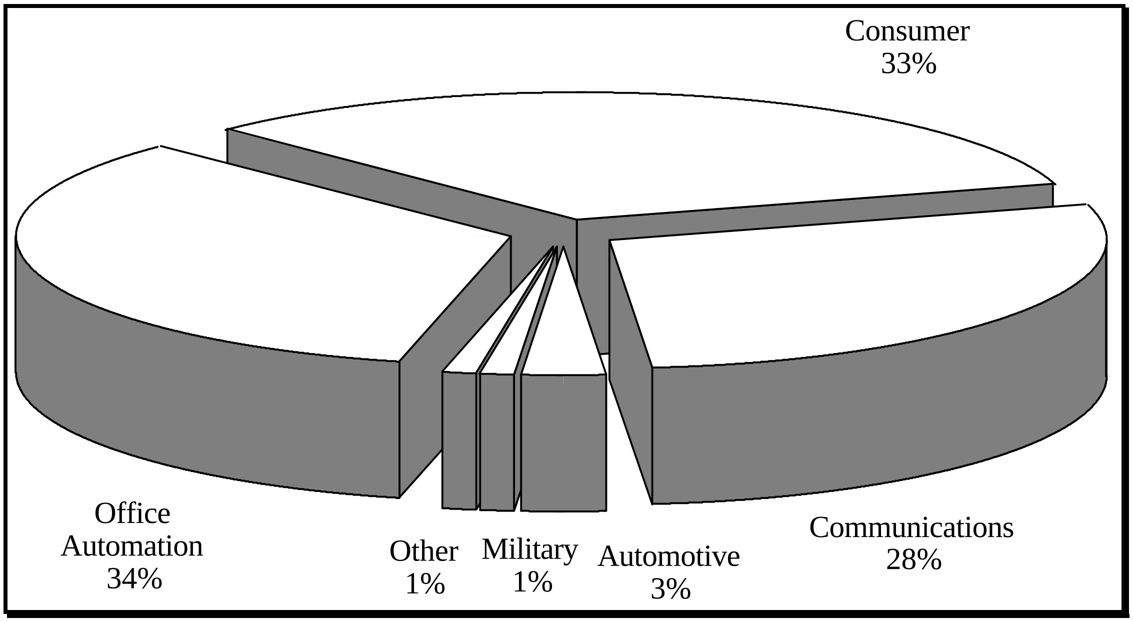

Figure 1: Worldwide Market For 32-Bit Embedded Microprocessors (Source: Information Architects)

The graph shows about one-third of the 32-bit volume going toward office automation. These are the “traditional” embedded applications, such as laser printers, fax machines, feature phones, and so on. These applications generally call for microprocessors that are powerful and available in high volume, but that don’t necessarily need to be conservative with power. Most office equipment is line powered, so power consumption and heat dissipation are often irrelevant.

Another one-third of the 32-bit embedded chips go into consumer-electronics items. Products like video games, portable games, CD players, and high-end audio/video equipment fit this category. Again, the processors must be available in high volume to succeed in this market and low price is very crucial. Some devices may also require low power consumption for battery-powered operation.

The communications sector includes devices like network hubs, routers, and switches as well as telephone infrastructure equipment. High volume is a requirement here, as is a reliable tool chain.

RISC VS. CISC IN WORLDWIDE VOLUME

There has been much written, said, and discussed over the virtues of RISC or CISC design principles. Regardless of the “correct” design philosophy, the fact remains that established CISC microprocessors outsell their RISC competitors by a wide margin. Looking again just at embedded 32-bit microprocessors, Figure 2 makes it clear that shipments of Motorola’s 68K family (including ColdFire and the 68300 microcontroller line) outstrips those of even the best-selling RISC chips. In fact, on this scale, many of the RISC architectures are lost in the noise , almost undetectable near the top of the graph. Also notice that “computers” (Macintoshes, PCs, and workstations) do about half the volume of 32-bit embedded processors.

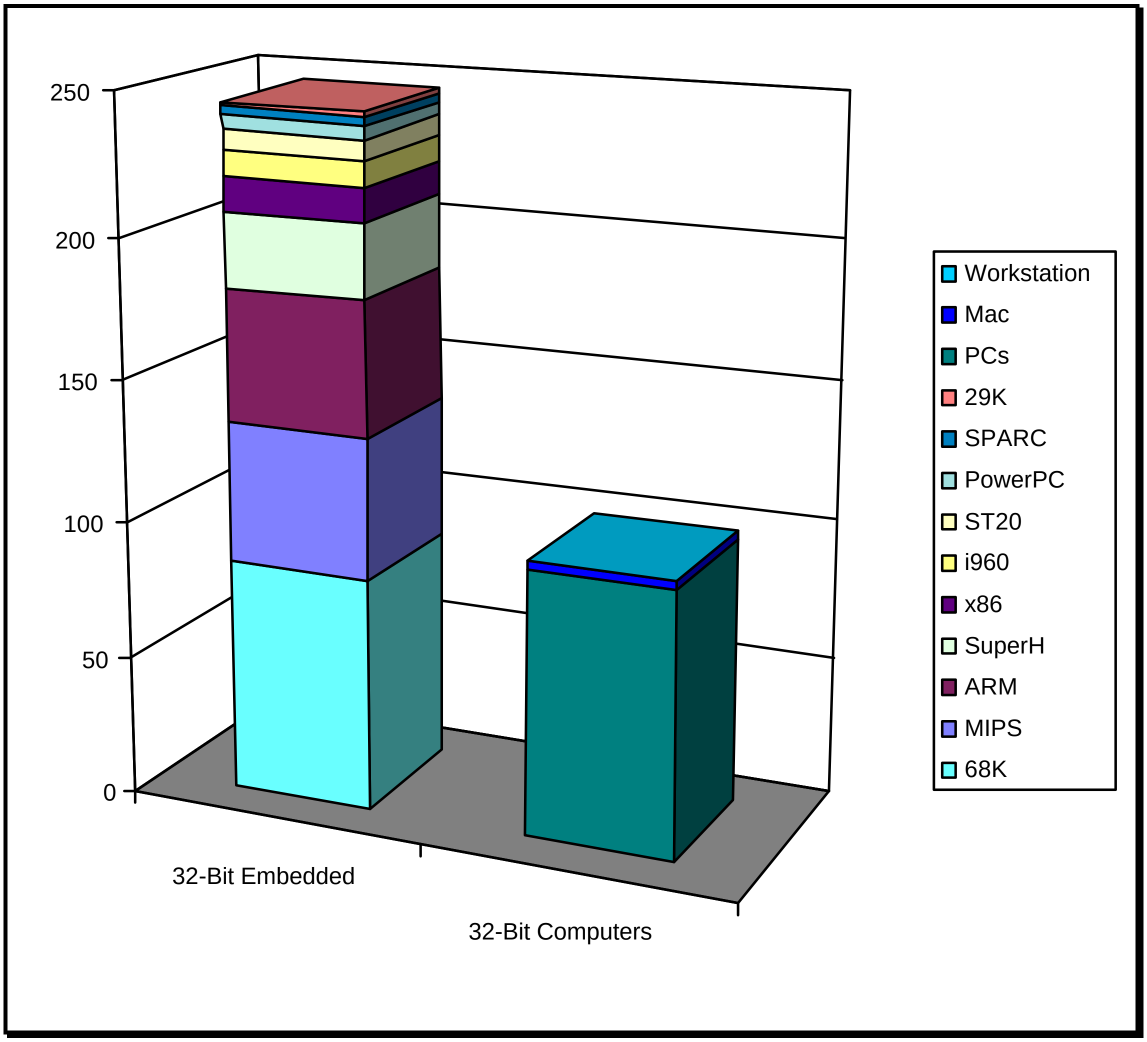

Figure 2: 32-bit processor volume, in millions (source: MicroDesign Resources)

Over the past few years, embedded RISC sales have taken off and there have been a number of winners, as Figure 3 shows. In the early 1990's, Intel's i960 architecture was the big success story. The i960 chips were used in popular products like laser printers (particularly HP's LaserJet series) as well as networking equipment.

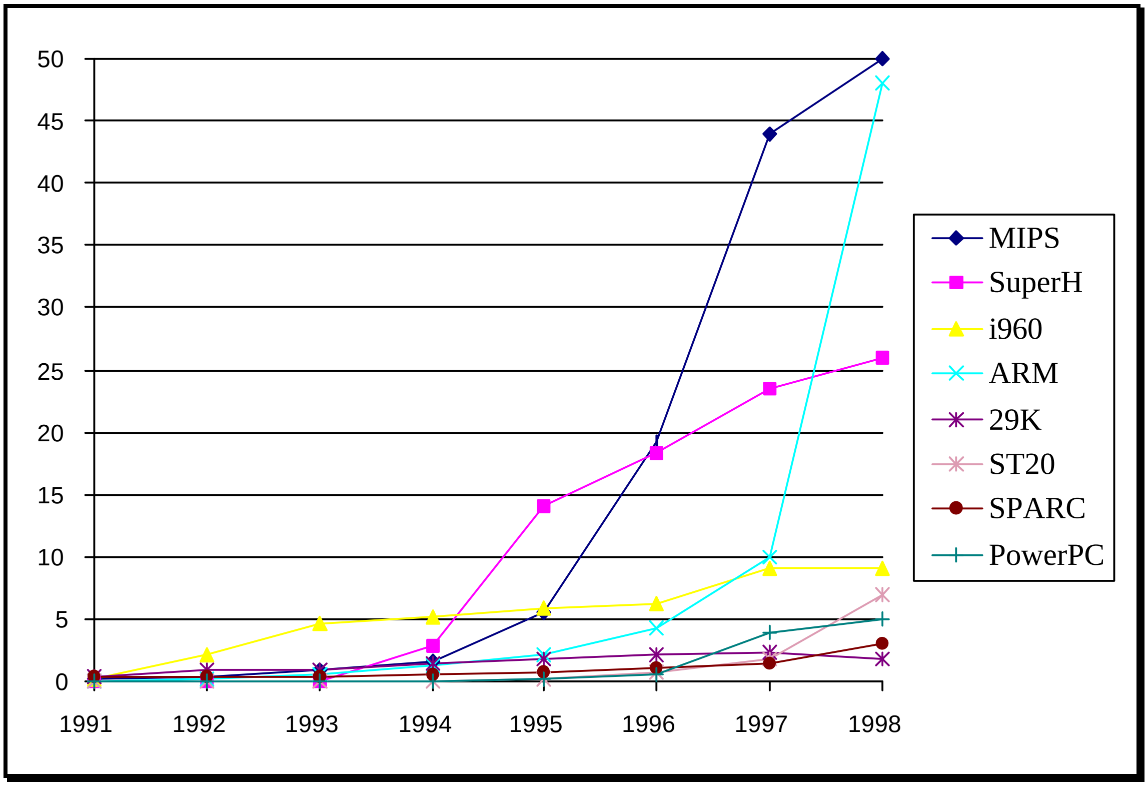

Figure 3: Embedded RISC volume, in millions (source: MicroDesign Resources)

More recently, a surprise newcomer usurped Intel's position: Hitachi’s SuperH family. This little known (and, until recently, little used) family of 32-bit RISC chips was chosen by Sega as the basis for its Genesis home video-game player. Every Sega Genesis had two Hitachi SH7604 microprocessors, and the system sold by the hundreds of thousands per month. Its successor, the Sega Saturn , uses no fewer than three SuperH microprocessors, plus a Motorola 68EC000. All by itself, Sega propelled Hitachi into the number-one spot for 32-bit RISC sales in 1995, far outstripping any other RISC processor, such as PowerPC, SPARC, or Alpha. Sega’s newest system, Dreamcast , also uses a Hitachi chip, the SH7750.

In 1996, Sega was overtaken by Sony and Nintendo—good news for the MIPS camp. Both the Sony PlayStation and the Nintendo 64 are based on MIPS processors, although the chips themselves are built by different companies (LSI Logic and NEC, respectively). The PlayStation is based on an ASIC with an R3000-series MIPS CPU core; the N64 uses NEC’s VR4300 processor—a 64-bit powerhouse that anyone can buy for less than $25 in volume. The next generation of the PlayStation again uses a MIPS-based processor, this time fabricated by Toshiba.

Sega and Hitachi are poised to make a comeback with Dreamcast, Sega's successor to the Saturn. Dreamcast is based on a single Hitachi SH7750 processor with special 3D geometry instructions. The system also has an ARM-based sound processor from Yamaha and a PowerVR graphics accelerator designed by Videologic and built by NEC. Dreamcast also runs Windows CE, which wil make it easier than ever before to port games from the PC to the video console, and vice versa.

Why RISC? Why CISC?

Why are the RISC chips taking off? And why do the CISC processors still outsell their more modern counterparts by an order of magnitude? Time. It takes time for any CPU to reach truly high volume, and the fact is, the CISC chips have been around much longer and have won designs that are just now reaching volume production.

But CISC chips also have real advantages that will keep them on top for many years to come. Their code density (the ratio of object-code size to real work done) is usually far better than for most RISC chips. The cost of many embedded products—particularly consumer-electronics items—is set by their memory cost, not the cost of the microprocessor, which makes code size an important factor.

RISC chips, on the other hand, provide better performance in some—but not all—cases. Their more modern internal designs can churn data through the pipeline faster than an x86 or 68K part. That is, unless the system is data-starved anyway. If the process depends on low-bandwidth memory, all that extra processing power may go to waste as the chip stalls waiting for more data or the next instruction.

ALL METRICS ARE SUSPECT

Unfortunately for most designers, there is no single, simple, reliable measurement for judging the performance of an embedded microprocessor. There is no BTU rating, as for air conditioners; no MPG rating as for automobiles; not even a SPECint or WinMark score. Unlike more conventional computer platforms, there is no standardized way to benchmark embedded processors.

Dhrystone Disclaimer

What we do have today is the much-abused Dhrystone benchmark. Dhrystone, developed by Reinhold Weicker in 1984, is a synthetic benchmark. That is, it was created specifically to act as a performance benchmark; it does no useful work. Unfortunately, no single benchmark can hope to measure the many and varied aspects of a CPU’s performance that embedded designers will be interested in. Just for examples, Dhrystone does not measure interrupt latency, floating-point performance, or signal-processing capability.

What Dhrystone does measure is a system’s integer arithmetic and string-handling capability. Dhrystone has been criticized because it spends a disproportionate amount of time—in the opinion of many—handling ASCII strings, as a word-processing program would. Dhrystone is in the public domain, is written in C, and demands very little of the target system, apart from a small amount of memory. These features have made it very popular, and its use is widespread. So is its misuse.

Executing the Dhrystone 2.1 benchmark takes only a few milliseconds on most 32-bit microprocessors, so typically, the code is allowed to execute several thousand times and the total execution time is divided by the number of iterations through the program. This produces a result such as, say, 24,750 Dhrystones per second. A chip that performs 20,000 Dhrystones/second is presumably faster than one that delivers only 17,000 Dhrystones/second, for instance.

What’s a VAX MIPS?

Another, different, result is often derived from this number. Supposedly, Digital Equipment Corporation’s venerable VAX 11/780 produces a result of 1,757 Dhrystones/second, running Dhrystone 2.1 with a particular version of the VMS operating system and with a particular workload. The VAX was generally considered by its creators and users to be a 1 MIPS machine; that is, it was believed to execute one million instructions per second. Thus it was inevitable that this result—1,757 Dhrystones/second—came to be used as a handy equivalent for 1 VAX MIPS. Any computer that could execute Dhrystone as quickly as a VAX became, by definition, a 1-VAX-MIPS computer. Every multiple of 1,757 Dhrystones/second was thus equivalent to 1 VAX MIPS . A benchmarking industry was born. (Note that 1 MIPS is still called “one MIPS.” There is no such thing as “one MIP;” MIPS is not a plural noun.)

Power Dissipation

No standard benchmarks are available today to measure a chip’s power consumption. Vendors will quote “typical,” “average,” or “maximum” power-dissipation numbers, but these measures are suspect. For example, what constitutes a typical workload? This issue becomes ever more important as power consumption decreases and differences of a few milliwatts equal a significant percentage of a system’s overall power budget.

Power consumption is often measured with the chip’s external bus quiescent or inactive. Especially with modern low-power CPU’s, a accounts for a significant percentage of its total power consumption. Instantaneous power consumption may double or triple when the CPU misses its on-chip cache and processes an external memory cycle. Like Dhrystone performance, published power consumption statistics must be taken with a grain of salt.

The Need For New Embedded Benchmarks

Clearly, there is a need for new benchmarks for embedded microprocessors. Designers need them to evaluate competing claims, vendors want them to gauge their own strengths and weaknesses, and analysts need them to draw accurate conclusions. The desire and the willingness are there—but the task is formidable.

First of all, can one benchmark ever satisfy everyone's needs? Can any single metric measure all of the things that a user would want? Or would a suite of benchmarks be more appropriate? By running several tests, each one designed to measure but a single aspect of the microprocessor's performance, a more detailed profile of the CPU can be generated. If desired, these individual results could then be averaged into a single CPU score.

Then there is the issue of distributing the benchmarks. Obviously, they can't be distributed in object-code form or they won't be portable across architectures. High-level source, such as C code, seems the most appropriate here.

Who develops the tests? If the vendors write them, conflict of interest issues may arise. Without confidence in the benchmarks, they serve no purpose. Samples of real-world code seem the most desirable, but most software vendors are reluctant to distribute source code for their important applications or algorithms.

How are the tests run? There is no such thing as a typical embedded system. Unlike PCs and workstations, the benchmark cannot depend on a standardized operating system, or disks, or a keyboard, or any other input/output device. Without these things, how can true performance be measured? Should the benchmarks be run with a standardized configuration (i.e., DRAM with a fixed access time), or on whatever hardware is available?

Who runs the tests? For the results to be valuable, either the source of the information must be trustworthy or the results must be reproducible and verifiable. If an objective third party tests all the systems, who pays for their efforts? If vendors test their own parts, how reliable are the results? And how fair is it to test competing microprocessors in wildly different systems?

If an independent third party is paid to perform all the benchmark testing, what happens to vendors that refuse to pay the testing fee—are their products left untested? And if not, what incentive is there for other vendors to continue paying?

NEW LICENSING BUSINESS MODEL

In stark contrast to business practices of just a few years ago, many microprocessor companies have broadly and openly licensed their architecture. The best examples of this strategy are MIPS Technologies (which used to be part of Silicon Graphics), ARM Holdings (Cambridge, U.K.), ARC Cores (London, U.K.), and Sun Microsystems (Mountain View, Calif.). Neither MIPS, nor Sun, nor ARC, nor ARM actually manufactures any chips. Instead, they license the design of their CPUs and instruction sets to chip makers around the world. These vendors then build and sell the chips to their customers, paying a royalty to the licensing firm.

Licensed designs have several advantages. For one, a licensed design is, almost by definition, available from multiple sources . Second, with multiple vendors supplying similar chips to the market, each vendor will compete by developing special variations for its target market, something a single vendor would not have the resources to do. Third, third-party tool developers are sometimes more eager to support an architecture with multiple sources of supply rather than a proprietary design. Licensing a design also allows smaller semiconductor companies to participate in the microprocessor market. By themselves, these companies would not have the wherewithal to design, develop, and support a microprocessor product line.

THE x86 IN EMBEDDED

The concept of using a “PC processor” (a member of the x86 family) in embedded applications draws one of two responses: enthusiasm or revulsion . Often, these debates are tied up in religious issues of the x86 architecture versus other, more recently minted, designs. In practical terms, x86 chips have some very strong advantages—and disadvantages—that make them worth careful consideration.

Proponents of embedded x86 chips point to the rapid price erosion that high-end PC processors have enjoyed. As the chart in Figure 4 shows, Pentium prices have plunged precipitously in the past, and show every sign of continuing to do so. This unprecedented price erosion, some say, gives these chips a very attractive price/performance ratio by the time they hit the embedded market. Cynics point out that the rapid decline in Pentium pricing only proves the chips were grossly overpriced to begin with.

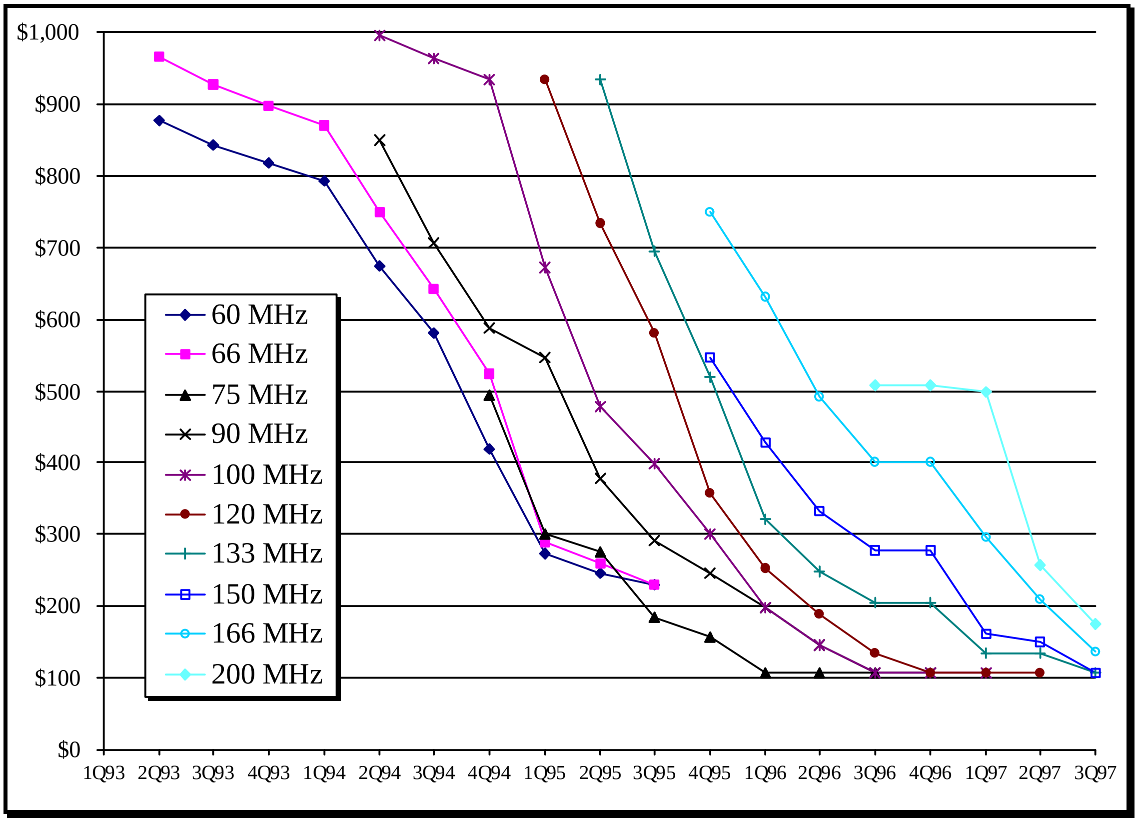

Figure 4: Historical Pentium price erosion (source: MicroDesign Resources)

Perhaps the biggest reason to use an x86 chip in a non-PC application is the wealth of development tools available for them. Like no other 32-bit architecture in the world, x86 processors enjoy an incredible assortment software- and hardware-development tools. Compilers, assemblers, emulators, debuggers—they're all available for x86 chips. Engineers familiar with the architecture are also easy to find.

Many embedded designers also eye enviously the great variety of PC-compatible peripheral chips available. Such hardware support can be a double-edged sword, however. While there are certainly many, many different PC-compatible graphics chips, core-logic chip sets, and other support logic, virtually all of those chips are designed exclusively for the PC market—a market that moves and changes very rapidly. Although the perfect graphics controller might be available today, tomorrow it may well be discontinued, leaving the embedded designer in the lurch. Most PC chip-set vendors are interested only in the PC market, and an embedded designer pleading for more samples of a discontinued item will get little attention.

THE RELATIONSHIP BETWEEN POWER, PERFORMANCE, AND PROCESS

Advances in semiconductor manufacturing technology have enabled a new class of microprocessors, and given impetus to a new segment of consumer electronics: portable electronics. Devices like PDAs (personal digital assistants), cellular telephones, pocket electronic organizers, and handheld games, among others, have all been enabled by a new class of very low-power microprocessors. What do all these chips have in common, why have they appeared only recently, and in what new directions are these chips headed?

Power Consumption Depends On Process Technology

Any microprocessor's total power consumption depends on a number of factors. Total consumption is proportional to:

- total clocked-device capacitance
- square of the supply voltage
- average clock frequency

Vendors have been steadily reducing the first two parameters to improve battery life while increasing the third to improve performance.

Smaller process geometries (for example, moving from a 0.8-micron process to a 0.5-micron process) reduce gate delays and capacitance, simultaneously improving performance and reducing capacitance, a major source of power drain. Reducing the feature size also reduces the overall size of the die, increasing a vendor's yield on a given device. (Although more aggressive processes usually have a higher defect rate, especially for the first several months, the shrink results in an overall increase in yield because the chips are much smaller.)

Vendors have also been reducing the supply voltage of their devices. In part, this change is driven by the smaller process geometries themselves; finer features need a lower supply voltage to avoid being damaged. But the lower voltage also pays off in lower overall power consumption, according to Ohm's law. At the present, there is a de facto standard at about 3.3 V, although this is falling fast. Currently, microprocessor vendors are shipping parts with supply voltages from 3.3 V all the way down to 1.65 V. Naturally, reducing the DC voltage reduces total power geometrically.

To avoid interface problems with these unusual voltages, many vendors are designing their microprocessors to use two separate voltages simultaneously—one for the CPU core and another for the external bus interface. For example, Intel's StrongARM-110 microprocessor uses both a 1.65-V supply and a 3.3-V supply.

Finer Processes Increase Yields, Reduce Costs

Power consumption increases linearly with clock frequency and geometrically with supply voltage. Thus, in order to increase performance more than power, vendors are motivated to lower operating voltages—a strategy that requires a substantial investment in modernizing its fab lines.

More modern fab lines pay off in lower power consumption, but also in increased volume as die sizes decrease. For most microprocessors, a defect anywhere on the die will render that die unusable. The larger the die, the greater the odds that a single flaw will ruin the device, wasting the company's time and money. A good rule of thumb is that the cost of a chip increases with the square of its area (which is, of course, equal to the square of its linear dimensions), if not more. Thus, there is again a strong incentive to move to new fab processes.

The coming months and years will see continued investment in leading-edge fabrication processes as the issues of power consumption, yield, and die size drive the most lucrative segments of the embedded market.

For a free weekly subscription to Embedded Processor Watch, e-mail to: join-embedded@list.MDRonline.com.