

The rules engineers live by weren't  
always set in stone **By Philip E. Ross**

# COMMAND

## WHAT IS IT ABOUT TECHNOLOGY AND EPONYMOUS "LAWS"?

In between discovering the electron and putting 50 million transistors on an integrated circuit, engineers and their predecessors have poured out a torrent of mathematical observations, pithy pronouncements, and even a few enduring self-fulfilling prophecies.

Some are sublime, and a lot more are silly. But you needn't speculate wildly on the burn-

are the bedrock of electrical engineering. But the laws that have reverberated and become

ingrained in mainstream culture and even the popular consciousness aren't really laws at all, but folksy rules of thumb.

Murphy's Law was first uttered by the military/

ing issue of just how many there are of each: Sturgeon's Law, named for science fiction author Theodore Sturgeon, posits that "90 percent of everything is crud." Silly or sublime? You be the judge.

Mixed among the musings are a few laws that actually are laws, really and truly. For technology, they define how things operate. Ohm's Law relates voltage across a component to the product of its resistance and the current through it, and Kirchhoff's Laws deal with the sum of currents at any point in a circuit; they

aerospace engineer Edward A. Murphy Jr., who is said to have declared after an improbably botched test in 1949 that "if there are two or more ways to do something, and one of those ways can result in a catastrophe, then someone will do it." And then there's Moore's Law. People who don't know the difference between CMOS and Spanish moss generally have heard of it. A little more than half a century into the solid-state age, a half dozen or so rule-of-thumb "laws" have stood out. How have they fared? Let's take a look.

# MENTS

## 1

## MOORE'S LAW

The number of transistors on a chip doubles annually

## THE MOTHER OF ALL ENGINEERING LAWS,

Moore's Law, was suggested in a paper by Intel Corp. legend Gordon E. Moore 38 years ago. However, he never used the word "law" to predict an annual doubling of the number of transistors that could be fabricated on a semiconductor chip. The paper, in the April 1965, 35th anniversary issue of *Electronics*, was titled "Cramming More Components onto Integrated Circuits." At the time, Moore was director of the research and development laboratories at the Fairchild Semiconductor Division of Fairchild Camera and Instrument Corp. The entire article was just three and a half pages, including two charts and a corny cartoon-like drawing of a shopper eyeing a sales booth for "handy home computers" (yes, Moore predicted them as well; it was arguably the more prescient insight). He noted the historical trend in fabricating transistors, then rather brief; observed that no technical barriers stood against further improvements in the enabling technology, photolithography; and reasoned that the trend in fabrication would continue for at least another decade, raising the chip transistor count to 65 000.

In fact, by 1975 the leading chips had maybe one-tenth as many components as Moore had predicted. The doubling period had stretched out to an average of 17 months in the decade ending in 1975, then slowed to 22 months through 1985 and 32 months through 1995. It has revived to a now relatively peppy 22 to 24 months in recent years.

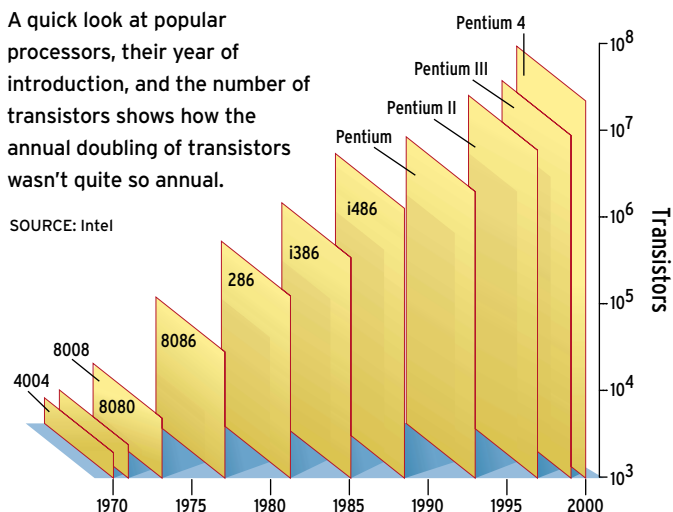
These statistics come from G. Dan Hutcheson, CEO of VLSI Research Inc., in Santa Clara, Calif., which compiles confidential industry data and releases it in aggregate form. Hutcheson, an economist, together with his engineer father, has studied Moore's Law perhaps more intently than anybody else. "It's averaged every two years since the late 1970s, although Intel's PR department likes to average the earlier number with the later and call it 18 months," Hutcheson notes.

From the beginning, Moore concentrated on the economic underpinnings of the trend, a focus he has always maintained, in contrast to the view that only what is technologically possible determines how long it takes for transistor density to double. The paper noted that the cost per electronic component was inversely proportional to the number of those compo-

## MOORE'S LAW IN ACTION

A quick look at popular processors, their year of introduction, and the number of transistors shows how the annual doubling of transistors wasn't quite so annual.

SOURCE: Intel



nents in simple circuits, but that diminishing returns occurred as the circuit grew more complex. In other words, eventually there would come a time when it just wouldn't be economically worthwhile to put more transistors on a chip.

"If you do see an end [to the law], it will be an economic end, not a technical end," Hutcheson says. "One of the most famous 'won't work' predictions was made in 1988 by Erich Bloch, then head of IBM Corp.'s research division, when chip features were around one micrometer. He said Moore's Law wouldn't work [at feature sizes] under a quarter micron." And, of course, it is holding up nicely, thank you, at today's 0.1  $\mu\text{m}$ .

Bloch left IBM not long after making that unfortunate prediction. Hutcheson credits Moore for having the economic basis of the law down cold but notes that later on he grossly underestimated the technical staying power of photolithography, thinking that the industry would soon have to turn to electron-beam techniques to continue doubling transistor density.

Since then, Moore's stock answer has been that "no exponential trend lasts forever, but forever can be postponed," Hutcheson says. "We may one day have to go to some sort of nanotechnology, with self-assembly of molecules, and so on, and that might not show the same economics. But we have a long time; semiconductors will be around for another 15 years at least."

One particularly intriguing question raised by the law is its "legal" status: does it go beyond description to prescription? In other words,



ILLUSTRATIONS: STEVE STANKIEWICZ  
PHOTO: BART NAGLE

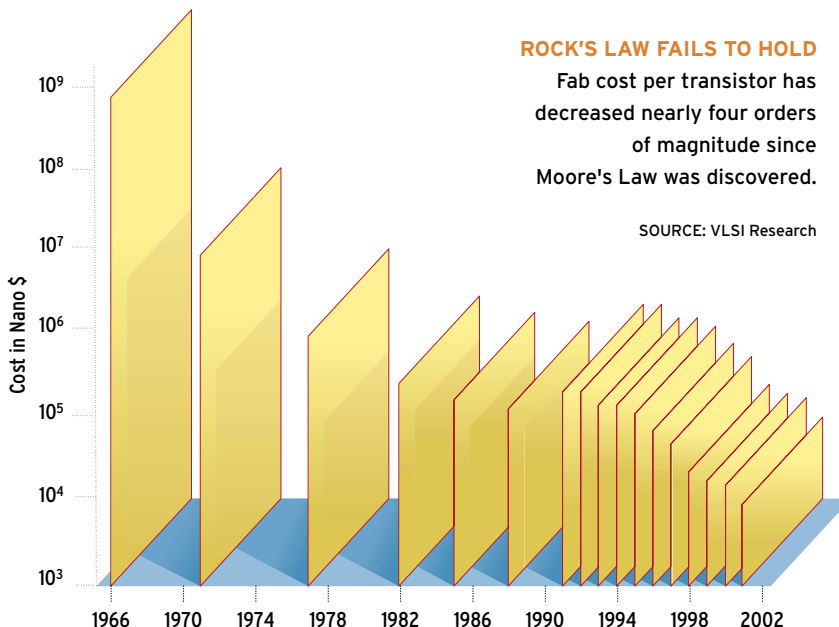
The "law" reflects economic constraints on the industry, above all, the yield rate that one obtains when producing the most complex chips of a given generation. -Gordon Moore

does the law merely describe reality or does it create it? Hutcheson says that it was Carver Mead, then at the California Institute of Technology, in Pasadena, and not Moore, who dubbed Moore's Law a "law," and he did so many years after Moore's paper was published. (Mead was instrumental in developing the MOSFET and a host of other inventions.)

The rest is history: the industry accepted the law as enunciated by Mead and incorporated it into a "road map" that set the bar for achievements in many areas—minimum line width, maximum wafer size, tolerances of tools, cleanliness of clean rooms. The technology seemed to take on a life of its own. Imagine that the industry decreed that the doubling cycle must speed up, say, by 7 percent, and all the relevant disciplines set their sights that much higher and made the decree a reality. In that case, the speedup in the transistor doubling period would happen. That's the argument that the pace is governed by technological capability.

No, say Moore, Hutcheson, and all the economists who study these things. The "law" reflects the economic constraints on the industry, above all, the yield rate (how many good chips are produced on a wafer) that obtains when producing the most complex chips of a given generation. Semiconductor manufacturers keep on adding elements to their circuitry until it no longer pays to add more; then they stop.

In other words, it all comes down to fabrication costs, which are spelled out in Rock's Law.



## 2 ROCK'S LAW

The cost of semiconductor tools doubles every four years

**SOMETIMES CALLED MOORE'S SECOND LAW,** because Moore first spoke of it publicly in the mid-1990s, we are calling it Rock's Law because Moore himself attributes it to Arthur Rock, an early investor in Intel, who noted that the cost of semiconductor tools doubles every four years. By this logic, chip fabrication plants, or fabs, were supposed to cost \$5 billion each by the late 1990s and \$10 billion by now.

Not so. VLSI Research estimates that fabs cost \$2 billion apiece, the same as in the late 1990s, even as their productivity has gone up. "In the 1980s, the fabs increased their yield; in the 1990s, they started [increasing] their throughput," Hutcheson says. (Throughput refers to the number of wafers a fab produces in a given time.) Wafer throughput rose from 20 per hour in the early 1990s to about 40 to 50 an hour today.

Anyhow, the focus was wrongheaded; what matters is not the cost of the fab but the value of its product. If a \$100 billion fab made so many transistors per penny that it could under-

cut the prices of a \$10 billion competitor, it would be economical (if, of course, you could get the seed capital together from a coalition of companies—or continents).

When integrated circuits were first made, the template for a circuit pattern—called a resist—was put down in wax by a human being wielding a 10-cent camel's hair brush. Then the wafer was dipped in sulfuric acid or some other etchant so that only the circuit pattern remained. Today, this job is done by a \$14 million, 193-nanometer-wavelength stepper; soon, the torch will pass to a \$25 million machine. Yet that tool is a bargain, far cheaper to use than the 10-cent brush, because it turns out so many billions more transistors. "All of China couldn't do a week's production of an Intel factory using camel's hair brushes as the etching tool," says Hutcheson.

So, if transistors have gone from a dime a dozen to a buck for a hundred billion (no lie), what has that done to the price of PCs? The answer comes, or used to come, from Machrone's Law.

## 3 MACHRONE'S LAW

The PC you want to buy will always be \$5000

**ACCORDING TO MACHRONE'S LAW,** the price of the computer you want to buy will always be \$5000. The law was framed in 1984 by Bill Machrone, a long-time columnist for *PC Magazine* and now also vice president for editorial development at Ziff Davis Media Inc. (New York City). "The light bulb went on for me one day when we were messing around with a computer time line and started putting a price on it, and the price didn't change," he says. "It was true almost from the outset of personal computers, going back to 1976. A fully configured Altair or even Radio Shack machine would cost you an easy \$5000. For 15 years, the law was pretty solid."

## ● Generation to Generation

Machrone's Law posits that the PC you want will always be US \$5000. That's held for a while, but with okay PCs for \$700 or less, you must want a lot of peripherals to spend \$5000.



**1976:** Altair 8800b with 8080A microprocessor, about \$5000 with all options and a terminal



**1984:** IBM 5155 Intel 8088 at 4.77 MHz, \$4225, amber monochrome monitor and no hard drive



**1994:** Apple Macintosh 8100 (G1), 80 MHz, configured with all options, about \$5000



**2003:** Apple Macintosh G5, 2 GHz, about \$5000, with largest flat-panel display and fully loaded

Today, however, it's almost as if the decimal point had moved a whole place, he acknowledges. "The magic number dropped to around \$3000 in the early 1990s and held there until about 2000, but now an okay machine costs well under \$1000, although a fully loaded one will still run \$2000 to \$3000."

Why was Machrone so happy to see that light bulb? "These 'laws' sell magazines," he admits.

Yet total spending on PCs continued to rise well after unit costs began to fall, because so many new buyers kept flooding into the market. Each new PC owner became fodder for online services, like AOL, and for the Internet, thus playing into Metcalfe's Law.

# 4

## METCALFE'S LAW

A network's value grows proportionately to the number of its users squared

**METCALFE'S LAW ASSERTS** that the value of a network grows as the square of the number of its users. "Having the only telephone in the world would be of zero value, but this value increases for each new telephone it can call," explained its author, Robert Metcalfe, in *The New York Times*, in 1996. Metcalfe, the inventor of the Ethernet standard and founder of the networking company 3Com Corp., in Santa Clara, Calif., first talked about the idea around 1980, but it was the journalist George Gilder who dubbed it a law, in an article he wrote for *Forbes ASAP* in 1993. Gilder maintained that Metcalfe's Law would amplify Moore's and, in so doing, remake the world.

Unlike the previous laws, Metcalfe's can't be quantified, because value—what economists call utility—can't be measured; you just know it when you see it. But that failing doesn't prevent economists from analyzing the law and correcting it. Consider the argument that most of the value you get from your telephone comes from being able to dial your 50 most frequently called numbers, and the rest of the network doesn't matter all that much. "You, like most human beings, may connect mostly to only 50 to 100 people, but those people, too, need 50 to 100 more. So adding to the network encourages them to join," says Hutcheson. That same sort of economy of network scale, he adds, explains why well-established standards, such as Windows, are well-nigh unbeatable.

Not all additions to a network make it more valuable, however, and some make it less so. "Some members of the network—we call them contaminants—subtract from the value, at least from your point of view," says Andrew P. McAfee, an assistant professor at Harvard Business School, in Boston. Examples include telemarketing firms that cold-call you while you're eating dinner, spam e-mail that promises larger body parts, and defective nodes that misroute the bytes you're trying to download.

In an article last year in *MIT Sloan Management Review*, "Confronting the Limits of Networks," McAfee and François-Xavier Oliveau, a consultant in the Boston Consulting Group's Paris office, identified four other problems in burgeoning networks—saturation, cacophony, clustering, and search. Saturation occurs



**“It is not the first time I am accused of having said something that I cannot remember having said—and most likely never have said.”** **-Martin Reiser**

when a network already contains most of the valuable material that new members can bring to it, say, all the music files that a file-sharing service can hope to glean from new members. Cacophony occurs when the interplay among members becomes too complex to follow, as in an Internet discussion group that spawns a “thread” containing hundreds of responses. Clustering occurs when members split into groups that use only part of the network, as happens when upper-class ladies try to raise money for charity through chain letters and end up corresponding only with each other. Finally, search costs grow to the point where most of the riches of a network remain inaccessible in practice.

There are ways around some of these problems, the authors argue. You can try to recruit new members possessing particularly valuable assets (as when a phone company offers incentives to sign up friends and family—people whom a subscriber is likely to call often). You can guide clustering in a logical and transparent manner by deliberately subdividing networks into formal subunits, as happens when a medical specialty produces a subspecialization. As for contaminants, such as spammers, “flamers,” and other lowlife of the Internet, well, you can banish them from the network or filter their messages robotically or by hand (as is done in moderated newsgroups).

Networks aren't the only thing that doesn't pay off quite as well as Metcalfe's Law would have it. Even improvements to hardware itself appear to offer diminishing returns, as Wirth's Law decrees.

## **5** **WIRTH'S LAW** Software is slowing faster than hardware is accelerating

**FIRST POPULARIZED IN 1995** by Niklaus Wirth of ETH Zurich (Switzerland), inventor of the Pascal computer language, this law states that software execution is slowing at a greater pace than hardware is accelerating. The key word is “popularized,” because Wirth explicitly credited the statement—which he put in an article right on the first page (in IEEE's own *Computer* magazine)—to Martin Reiser, formerly at IBM Research, now director of the Fraunhofer Institute for Media Communication, in Sankt Augustin, Germany. “It is not the first time I am accused of having said something that I cannot remember having said—and most likely never have said,” Reiser told *IEEE Spectrum*.

The same idea in Wirth's, or rather Reiser's, Law, has been expressed by an anonymous industry wit in Biblical cadences: “Groves giveth, and Gates taketh away.” That is, as the speed of calculation rises, thanks to Intel's Andy Grove, the amount of calculation needed to do the job rises also, thanks to Microsoft's Bill Gates, leaving hardly any gains for the user to enjoy.

Wirth complained, in the article, that text editors of the early 1970s worked with 8000 bytes of storage, whereas then-modern equivalents demanded 100 times as much. “Has all this inflated software become any faster?” he asked. “On the contrary. Were it not for a thousand times faster hardware, modern software would be utterly unusable.” Most of the features that bloated the programs were superfluous for most of the users, most of the time. For example, I can use Word 2000 to spell “Greek” in Greek letters: Γρῆκ. There! I've done it, for the first time in my writing career, and probably the last.

Wirth explains that users tolerate “feature bloat” for two reasons: an embarrassment of computing riches, thanks to Moore's Law, which makes the bloat possible, and an ignorance among consumers that prevents them from distinguishing the useful from the useless, even if they had some influence over Microsoft's software design.

He goes on to argue, however, that the root cause of bloat is not the tolerance of consumers but the interests of software companies. Features like color, gray-scale graphics, pop-up icons, and all the rest may make the computing experience more “friendly” some of the time, but they sure don't have to be in one's face all the time—yet they are, because design is set in stone. Programs needn't require massive manuals that no layman can penetrate, but if they do, they are more likely to hook customers for the long haul, Wirth maintains. Above all, the pressure to get new products out the door lest others beat them to the market prevents programmers from meticulously pruning their work.

Of course, the laws and rules of thumb don't stop with these five. One for the Internet Age is Nielsen's Law of Internet Bandwidth. In April 1998, Jakob Nielsen, an Internet usability expert, issued his own law. Nielsen predicted that a high-end user's connection speed to the Internet will grow by 50 percent per year, but Web site developers won't get to take advantage of this added bandwidth to make Web pages larger until 2003. Was he right? You be the judge. ●

### TO PROBE FURTHER

See Hutcheson's chapter on the economic aspects of Moore's Law in the upcoming *High-K Gate Dielectric Materials for VLSI Mosfet Applications*, Springer-Verlag, New York City, 2003.

Robert Metcalfe began talking about his law around 1980, and George Gilder dubbed it a law in “Metcalfe's Law and Legacy,” *Forbes* ASAP, 13 September 1993. Metcalfe first published his law in “There Oughta Be a Law,” *The New York Times*, 15 July 1996.

McAfee and Oliveau's article on problems in burgeoning networks, “Confronting the Limits of Networks,” can be found in *MIT Sloan Management Review*, Summer 2002, Vol. 43, no. 4.

Wirth's article crediting Reiser, “A Plea for Lean Software,” appeared in *IEEE Computer*, February 1995, Vol. 28, no. 2.