Written by
Nick Tredennick

DynamicSilicon

Published by
Gilder Publishing, LLC

The Investor's Guide to Breakthrough Micro Devices

Moore's law Shows No Mercy

N*erdvana*. Heathkit sold high-quality kits to build just about anything electronic (radios, TVs, amplifiers, multimeters, oscilloscopes…). Me and the rest of the world's nerds loved Heathkit. I built a high-end AM/FM receiver. I built several clocks that displayed time broadcast from the National Bureau of Standards (now NIST). I built one of the first motion-detectors. I would have built every kit Heathkit offered if I had had the money and the time. About the time I could afford any Heathkit I wanted, the company died. Moore's law killed it. The piles of components—that kit builders soldered into boards—became integrated circuits, killing the business for small system kits. Moore's law killed it without a thought for the unrequited desires of tens of thousands of nerds. It's a lesson industry should learn.

Moore's law says the number of transistors on a chip doubles about every eighteen months. Moore's law defines progress in semiconductor fabrication—it translates into performance increases and cost decreases in electronics. Moore's law defines the *supply curve* for electronics and *we think it drives the industry*. It doesn't; there's a complementary *demand curve* we forget. Industry analysis is frequently based on the assumption that the demand curve is always above the supply curve. But it is the relationship of the supply curve to the demand curve that is important. Changes in this relationship precipitate dramatic changes in the industry.

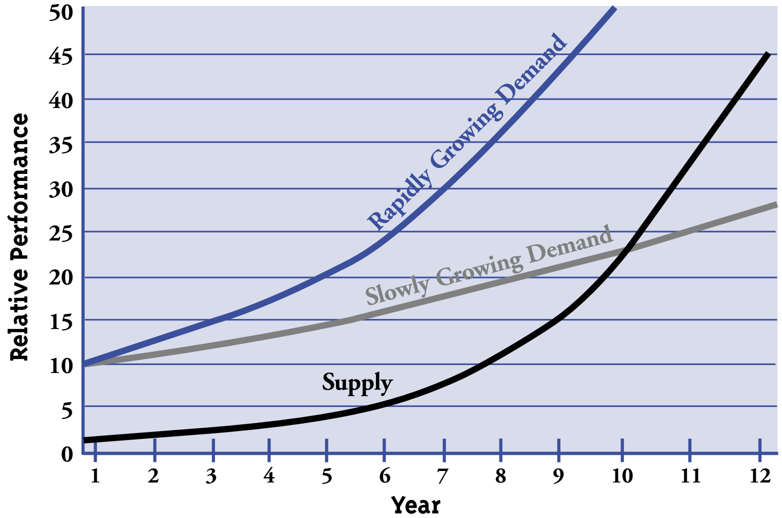


Fig. 1. When the PC was introduced demand for performance was higher than supply. Over time the PC's performance grows with Moore's Law. Demand for performance is difficult to gauge.

Take the personal computer. Such changes have transformed the world of computing and will soon overthrow the existing regime in networking as well. Let us follow the story, beginning with PCs. PCs compete on performance. When PCs were introduced, their performance was well below customers' demand. In fig. 1, I show PC performance as supply and demand curves for a hypothetical twelve-year period. Users wanted more performance than they could get. I arbitrarily chose demand starting at ten times performance at year one. From its starting point, I show PC performance doubling about every two years (close to the Moore's law rate of semiconductor fabrication improvement). The demand curve, which is the performance users want, is difficult to assess. The demand for performance started out above the supply. If the demand for performance grew faster, then the supply would never reach the demand. Intel's business model rests on this assumption. I show this in fig. 1 with the curve labeled "rapidly growing demand." If the supply of performance grows faster than the demand,

In This Issue:

Vol. 1, No. 3
March 2001

then the supply will eventually exceed the demand. I show this with the curve labeled "slowly growing demand." When demand exceeds supply, it is a seller's market and the manufacturers can charge what the market will bear. When supply crosses demand, it is a buyer's market and manufacturers have to lower prices.

Supply and demand curves crossed in the hard disk market. Storage capacity is being supplied and it grows fifty percent faster than Moore's law because it is simpler to create smaller magnetic domains on a disk than to etch smaller transistors and interconnect them on a chip. The demand for capacity (at least in individual PCs) grew more slowly. Before the supply exceeded the demand, 5.25-inch hard disks dominated the market and 3.5-inch hard disks held niches such as the notebook segment. But disk capacity grew faster than the demand for capacity, so the supply of disk capacity eventually exceeded the demand for it. When the capacity of 3.5-inch hard disks could satisfy the market demand, 3.5-inch hard disks displaced 5.25-inch hard disks and soon dominated the market. The cheaper, lower-capacity 5.25-inch hard disk had displaced its higher-capacity, 8-inch cousin. History repeated itself when the lower-capacity, but still satisfactory, 3.5-inch hard disk displaced the 5.25-inch hard disk.

Integrated circuit macros and the microprocessor

An integrated circuit (IC) is an electronic circuit on a single silicon chip. The invention of the integrated circuit spawned the IC macro business. So-called IC macro functions displaced discrete components with families of compatible chips (registers, adders, multiplexers, counters, decoders, etc.). Beginning with the first commercial IC macro functions introduced by Fairchild (FCS) in 1961, the family of IC macro functions grew in variety and in complexity with Moore's law. By 1969, enough transistors would fit on an IC to build a rudimentary central processing unit (CPU) for a computer. In 1971 Intel introduced the first commercially available single-chip CPU. It was the birth of the microprocessor.

The microprocessor didn't start out as the CPU in a computer system; instead, it saw its main use in embedded systems. In embedded systems, the microprocessor's function isn't general-purpose computing; it is specific to the application (e.g., controlling the heating element in a hair dryer or regulating the fuel-air mixture in a fuel-injection system). The microprocessor and a few standard components (ROM, RAM, and peripheral chips) spanned a huge range of applications. Building a microprocessor-based controller for a microwave oven was similar to building a microprocessor-based controller for a washing machine or for a service-station gas pump. The whole range of applications benefited from a common set of development tools, programming languages, design methods, and components. Since a vast range of applications employed the same small set of common components, the microprocessor and its standard components achieved high production volumes. High production volumes led to lower cost and to higher performance. High volume designs are updated more frequently to reduce chip size. Smaller chips are cheaper to make and they run faster. Lower cost and higher performance increased the microprocessor's range of applications and improved its competitive position against IC macro functions.

Originally, the microprocessor was too expensive for very small problems, not capable enough for large problems, and not fast enough for high-performance problems. IC macros implement functions more directly than the microprocessor can and thereby always achieve higher performance, but, as the microprocessor's performance improves, it displaces IC macros where the microprocessor's performance is adequate. The microprocessor displaced IC macros because it was cheap and had adequate performance and because it improved the designer's productivity.

PCs displace proprietary computer systems

Commercial mainframe computers have been around since the 1950s. Minicomputers followed in the '60s and '70s. Mainframes and minicomputers were proprietary designs; each company had its own instruction set and its own system design. Minicomputers proliferated in the '70s. As Moore's law improved the performance of underlying components, the minicomputer displaced (but didn't completely replace) the main-

DynamicSilicon

Editors	**Nick Tredennick**
	Brion Shimamoto
Publisher	**Lauryn Franzoni**
Associate Editor	**Keri Geiger**
Designer	**Julie Ward**
President	**Mark T. Ziebarth**
Chairman	**George Gilder**

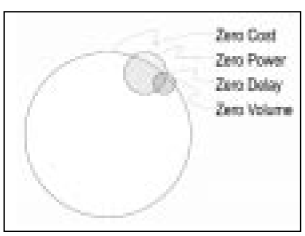
DynamicSilicon is published monthly by Gilder Publishing, LLC. Editorial and Business address: P.O. Box 660, Monument Mills, Housatonic, MA 01236.

Editorial inquiries can be sent to: bozo@gildertech.com Single-issue price: $50. For subscription information, call 800.229.2573, e-mail us at dynamic@gildertech.com, or visit our website at www.dynamicsilicon.com

frame. What was being supplied was performance or price/performance. When the absolute performance or the price/performance of the minicomputer exceeded the demand in a particular market segment, the minicomputer displaced its mainframe rival. In the '80s, workstations, improving performance and price/performance at a Moore's law rate—from below the minicomputer—soon crossed the demand curve to displace and to eventually wipe out the minicomputer.

Workstations and PCs, the first computers with a microprocessor as the central processing unit, got their start in the early '80s. Until the introduction of workstations and PCs, the microprocessor had served only the embedded applications market where its design goal was low cost. PCs and workstations changed the microprocessor's design goal from low cost to performance. Workstations emphasized performance and engineering productivity. Aimed at the value market; PCs emphasized low cost, but they also competed on performance.

In the '80s workstations, from companies such as Hewlett-Packard (HWP), Tektronix (TEK), Digital Equipment, Apollo, IBM (IBM), Intergraph (INGR), Sun (SUNW), MIPS (MIPS), and Silicon Graphics (SGI), competed on performance for a limited number of engineering design seats. Meanwhile, the personal computer, represented primarily by Motorola-based (MOT) Apple (AAPL) and Intel-based (INTC) IBM-compatible entries, strove to penetrate the value-based consumer market.



The electronic systems market is primarily three overlapping segments defined by their ideal target: zero cost, zero power, and zero delay. The zero-cost segment is consumer items, so the design emphasis is low cost. The zero-power segment is portable devices, so the design emphasis is power conservation. The zero-delay segment is performance-oriented systems, so the design emphasis is quick response to requests. Personal computers compete on performance in the consumer market, so they are in an overlapping area between zero cost and zero delay. Workstations compete on performance in the limited engineering-design market, so they are in the zero-delay segment outside of the overlapping area with the zero-cost segment.

In the mid-'80s, Apple executives, in a strategic blunder I will never forgive (since they get paid millions of dollars to make good decisions), forgot that they were in the overlap of the zero-cost and zero-delay markets, and chose to take higher margins in lieu of competing for market share (by way of lower prices). Consumer markets are so sensitive to price that Apple soon irretrievably lost market share and can be ignored for this analysis. That leaves us with servers, workstations, and Intel-based PCs.

It looked as if PCs and workstations were in different market segments, since the PC had to be concerned about cost and workstations did not. Then Moore's law interfered. Workstation makers designed microprocessors for performance and hoped that better performance would lead to more sales. Intel designed for volume manufacturing and rode Moore's law to better performance. High-volume manufacturing and a highly competitive consumer market drove a higher evolution rate for the Intel microprocessor's semiconductor process, so it eventually overtook workstation microprocessors in absolute performance. Workstations supplied performance. PCs supplied price and performance. The higher evolution rate of the PC drove its performance *above the demand* for large segments of the workstation market.

High-volume production of the PC led to its invasion of the zero-delay segment outside the zero-cost segment, where it forced consolidation of workstation vendors. The PC was a profitable business for IBM, but IBM lost control of the system as it became an open standard. As the system and its components became commodities, profits in the PC system business decreased rapidly, except in Microsoft's (MSFT) Windows operating system, Microsoft's Office applications, and the microprocessor. Intel, manufacturer of the PC's microprocessors, grew revenues from \$789 million to \$34 billion on the back of the PC industry and enjoyed twenty years of sixty-percent gross margins in the process.

Microcontrollers and miscellaneous logic

The 16V8 and 22V10 are programmable array logic (PAL) devices. The 16V8 has 16 inputs and 8 outputs; the 22V10 has 22 inputs and 10 outputs. The engineer programs the internal array to construct small, arbitrary logic functions. They hold the miscellaneous logic that ties disparate signals on the board together to complete the system design. *Every* electronic system needs them, so it's a huge commodity market.

Moore's law says the number of transistors on a chip doubles every eighteen months. The function implemented for a fixed cost grows with time or the cost to implement the same function decreases with time. The 16V8 and the 22V10 are fixed functions, so they should get smaller (and faster) with Moore's law progress in semiconductor fabrication. They should, but they don't. Fig. 2 explains why. The circuit and the ring of bonding pads are fabricated on the silicon substrate. Bonding pads are the link between the circuit and the outside world. As semiconductor fabrication improves, the silicon substrate and the circuit shrink and the bonding pads get closer together. But the size and spacing of the bonding pads is limited by the capability of mechanical bonding equipment. As the circuit shrinks, the chip eventually becomes "pad limited," meaning that the size of the silicon substrate is determined by the minimum size of the pad ring. Once this happens, the circuit can continue to shrink, but the chip will not get much cheaper or much faster. Size dominates chip cost, so when the chip stops getting smaller, its cost stops decreasing. Likewise, once the circuit is a tiny speck in the center of the silicon substrate, its performance will be limited by propagation delays associated with the chip's physical packaging. Today's 16V8 and 22V10 are pad-limited chips.

The general-purpose microprocessor displaced IC macro functions and, in doing so its market has grown rapidly. Manufacturers ship about seven billion microprocessors annually. About 150 million microprocessors ship as the CPU in a computer system; but the rest of the seven billion microprocessors go into embedded applications. Microprocessors for embedded applications are mostly destined for the zero-cost consumer market. They have to be cheap and they generally contain other functions such as timers, counters, and input/output registers. Since they contain logic other than just the central processing unit, they are called microcontrollers. They run washing machines and hair driers; they don't need much computing horsepower, so they are primarily limited-function, 8-bit microprocessors with a variety of on-chip peripheral functions. Many are either pad limited or they are several generations behind the leading edge of semiconductor fabrication. The microcontroller market is a commodity market. The microcontroller market has grown to billions of units as it displaced IC macro functions and as it grew its own application base. These microcontrollers may become victims of their own success.

Microprocessors displaced IC macro functions because they offered a few standard parts that could displace a wide variety of parts in a range of applications. As microcontrollers have become more successful, their variety has risen rapidly. Motorola's microcontroller web site (begin viewing at http://e-www.motorola.com/webapp/sps/prod_cat/index_pc.jsp), for example lists three 8-bit, two 16-bit, and three 32-bit microcontroller "families." Within each of these families, there are between two and thirty-eight different chips listed. For each chip, there may be five or so ways that the chip can be packaged and there may be several speed grades. This amounts to hundreds of different parts and this doesn't include Motorola's microprocessors, communications processors, digital signal processors, security processors, coprocessors, or the unique microcontrollers it makes for companies such as General Motors (GM).

Motorola is one of dozens of microcontroller manufacturers. The engineers trying to find the right microcontroller for a particular application can choose from thousands of parts. In a typical application, the engineer might use a microcontroller that has most of the desired peripherals, functions, and interfaces and might compensate for missing functions and interface mismatches by employing several 16V8s. The 16V8s sit between the microcontroller and other parts of the circuit (such as sensors and actuators). In addition to the function it contributes to the circuit, the 16V8 has one interface circuit to talk to the microcontroller and another to talk to the sensor or actuator.

Rather than slogging though thousands of microcontroller configurations to find the appropriate part for a particular application, it would be better if the engineer could begin with a general-purpose microprocessor and a menu of peripheral functions that could be configured and added as needed. If the microcontroller's interfaces could be configured to match the particular application's requirements, the 22V10s and 16V8s could be eliminated in the design. The application's design is more reliable since it has fewer components and it is more efficient since it doesn't require interface translation between the microcontroller, the miscellaneous logic, and the sensors and actuators.

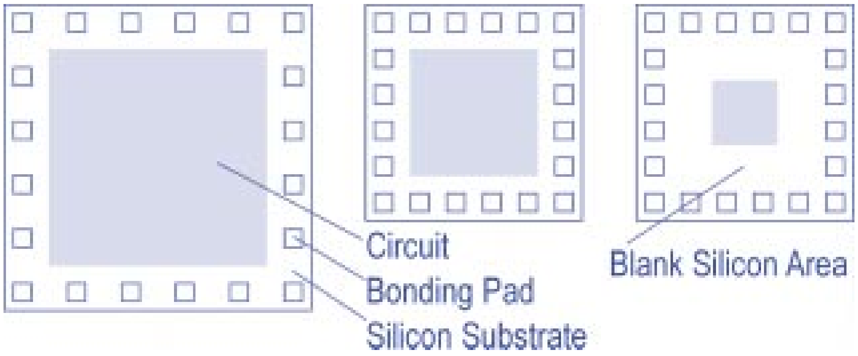


Fig. 2. As semiconductor fabrication improves, the substrate and circuit get smaller and the bonding pads get closer until the bonding pads limit the size of the silicon substrate.

Triscend's E5 is just such a component (http://www.triscend.com). The blank silicon area in a pad-limited design is free. Triscend's engineers chose a popular 8-bit microcontroller for the core, threw out the peripherals, and then crammed programmable logic into the remaining area. The chip isn't much larger than a standard pad-limited microcontroller, so its production cost could be similar. The programmable logic allows the engineer to build peripherals and interfaces to suit the application. Triscend supplies a few unique part numbers rather than hundreds, so the range of applications for each part is vastly increased as are the production volumes.

Triscend has the potential to consolidate microcontrollers. I add Triscend to the list of Dynamic Silicon Companies for its early position in on-chip programmable logic for the zero-cost segment, but with the warning that its executives should not forget the mistake made by Apple executives in setting prices for this consumer-driven market segment.

PLDs displace ASICs, DSPs, and microprocessors

Texas Instruments leads the world in optics based on microelectromechanical systems (MEMS) with its digital micromirror display (DMD). TI is building MEMS-based components for hard disks. It has announced a direct-conversion receiver chipset for cell phones. It is also the world leader in digital signal processors (DSPs). It's a world leader in MEMS in optical applications and in components for untethered devices. Wouldn't it be the perfect Dynamic Silicon Company? No. I'm not likely to add companies such as Texas Instruments (TXN), Motorola, IBM, HP, Siemens (SIE), and STMicroelectronics (SMT) to my list. These companies, however well run and however wonderful their technology in a particular area, are too large and too diversified to benefit substantially from leadership in dynamic silicon's segments.

By contrast, **Altera** (ALTR) and **Xilinx** (XLNX) belong on my list. Altera and Xilinx have built billion-dollar businesses with a narrow focus on programmable logic devices (PLDs). Between them, they dominate the programmable logic market. Programmable logic devices support today's Telecosm and they will be key components in tethered devices for the fibersphere's future. Altera and Xilinx are on the leading edge of a long-running wave of success as PLDs displace fixed-function logic. Fixed-function logic is IC macros and application-specific integrated circuits (ASICs).

Altera and Xilinx have been growing as fast as they can to stay abreast of customer demand. The danger is that the company supports its customers to the best of its ability until the company and the customers have painted themselves into a corner. That is what Altera and Xilinx are doing. They have built their businesses on prototyping and on displacing ASICs. Think of PLDs and ASICs as suppliers of circuits. The supply of circuits is growing at a Moore's law rate. The supply of circuits is growing faster than the demand for circuits. PLDs, because of the overhead in transistors that support their personalization, are well below ASICs in the number of usable circuits that they can supply. But, because both the PLD and the ASIC grow rapidly in the number of circuits they can supply, *the PLD is crossing the demand line for market segments held by ASICs*.

The PLD has an advantage over the ASIC in production volume. A single PLD fits a vast range of applications while each ASIC fits one application. The PLD has an advantage in flexibility; the designer easily corrects errors in a PLD-based design, but the ASIC has to be scrapped. The ASIC enjoys advantages in circuit speed and in gate capacity (PLDs have a lot of overhead transistors). The ASIC has high fixed cost and low per-unit cost (if production volumes are high enough); the PLD has no fixed cost and high per-unit cost. For high-volume production, an ASIC is more cost effective; for low-volume production, the PLD is more cost effective. That's the way it's supposed to be, but Moore's law interferes again. As semiconductor fabrication improves, the PLD's cost decreases and its speed and capacity increase. PLDs displace ASICs at the low end of the ASIC's application range. The average ASIC design gets bigger each year (more logic circuits per ASIC), but the size of designs isn't growing as fast as PLDs are growing, so PLDs are eating into the market for ASICs.

That's the way the PLD business has worked since it started: it's been prototyping and it's been eating into the low end of the ASIC market. It's a great business and it will continue to grow. It fits with my vision for the future and it fits George Gilder's Telecosm vision. Altera and Xilinx are among my Dynamic Silicon Companies because Moore's law is on their side, while it is working against the ASICs. But, to support their growing customer base, Altera and Xilinx have a strategy that supports the ASIC and prototyping model. It will eventually paint them into a corner as untapped markets go to more aggressive companies. More potent than PLDs' advantages in prototyping and in competing with ASICs are their advantages in such larger markets as DSPs and microprocessors.

The DSP and the microprocessor are general-purpose components. They displaced direct hardware implementations with programmed solutions, even though they are inefficient and slow in comparison with a direct hardware solution. The DSP and the microprocessor displaced direct hardware solutions because they had adequate performance, they increased the engineer's productivity, and they were cheaper. They were cheaper by virtue of high-volume production of a limited number of standard components applied across a vast range of applications.

Moore's law has given the DSP and the microprocessor a long honeymoon. This couple has been riding a wave of speed and capability improvements that has let them dominate the application space. As a result the engineering community, educational system, and development tools look first to microprocessor-based solutions. The low efficiency of their solutions hasn't been an issue because performance has been *adequate* and power has been *free*. Now power and efficiency are becoming important. For mobile applications, power and efficiency are obviously important, but they are becoming important in tethered applications too. Ask the manager of a server farm if it's easy to supply 5 KW per rack to several hundred racks of computers. The average server farm in Silicon Valley asks the power company to deliver twelve megawatts (that's fifteen to twenty times what a typical commercial office building uses).

Direct solutions implement the application's algorithms in hardware, rendering engineering equations representing the problem solution directly into logic circuits. It is a high-performance and efficient solution, but it requires considerable engineering effort. Microprocessor-based solutions program the problem solution on a computer's instruction set. It is an inefficient and indirect solution to the problem, but programming the solution increases the *engineer's* productivity. PLDs lie in the middle between the direct solution and the programmed solution. They are more efficient than the programmed solution, but require more engineering effort as well.

Network performance is increasing on one side of the server and the hard disk's interface is getting faster on the other side. The server in the middle is the bottleneck. It's not efficient for the server to handle all of the data transfers through the CPU. As BlueArc has shown with its file server, it's more efficient to direct traffic with Altera PLDs than for the CPU to futz with the protocols and data buffering at both ends. BlueArc has implemented the data flow equivalent of the file server with Altera PLDs. Its performance and efficiency allow it to handle 2 Gb/s at the network's Ethernet interface and 1.7 Gb/s at the hard disk's Fibre Channel interface. In the file server, the efficiency of the PLD-based implementation knocks the socks off the microprocessor-based implementations. BlueArc offers proof of concept for the notion that PLD-based solutions can be substantially better than microprocessor-based solutions. The file server is here; web servers, base stations, switches, and routers will follow.

While Altera and Xilinx diligently position themselves for prototyping and for displacing ASICs, they have been overlooking a huge untapped market: positioning themselves against DSPs and microprocessors. The PLD's advantage in efficiency and in power conservation could win huge chunks of the market from the dinosaurs. It's time to end the honeymoon.

Dynamic logic applications

I proposed a direct assault on the application space held by DSPs and microprocessors. It's a good place to start because there are cases that prove the concept. It will be difficult to dislodge the microprocessor because microprocessor-based solutions are ingrained. It will take time, but it can and will be done.

Beyond the direct assault lie more sophisticated dynamic-logic solutions. A dynamic-logic implementation takes advantage of the PLDs ability to "page" logic into the system as it is needed, further boosting efficiency and conserving power. The first proof of concept dynamic-logic implementation will be a high-volume, portable, consumer device. It will have to be because a dynamic-logic implementation requires engineering effort so intense that it must be amortized across large volumes. QuickSilver Technology, is building a dynamic-logic chipset for mobile devices.

PCs displace proprietary network processors

Cisco (CSCO) didn't make George Gilder's list of Telecosm companies. He doesn't like Cisco's future. George thinks Cisco may be hollowed-out—the way the PC got hollowed-out—with profits going to the manufacturers of the network processors. I agree with George's conclusion, but I arrive by a different route.

Cisco's business is in some ways similar to the PC business, and in some ways, different. Networking equipment is microprocessor-based and it is software intensive, just like the PC business. In the PC's case, the manufacturers didn't have control of the key system components. Microsoft owned the applications and the operating system and Intel owned the processors. Profits went to Microsoft and to Intel. Since Cisco owns the systems, the software, and network processor development, key components are concentrated and profits cannot escape. In the networking business, Cisco is Microsoft and Intel and the system manufacturer all in one.

The PC and the router. If your company has offices in Sunnyvale, Austin, and Fargo, a virtual private network (VPN) ties your offices together securely over the Internet. A VPN extends your local-area network securely to remote locations and it ties a mobile client to the home network. A VPN forms a secure, encrypted "tunnel" between the local network and the remote network.

OpenReach offers virtual private networking based on open-source software running on a PC (Brion Shimamoto is chief technology officer at OpenReach and I am on its board of directors so you know we're biased). OpenReach does this by dedicating a cheap PC at each location to manage VPN connections to other locations. The PC can displace the local router. You don't need a router for the local network's connection to the Internet; the PC can do the necessary packet processing for a T1 line. In fact, today's PC can do the packet processing for a T3 line. For OpenReach, it was simpler to duplicate the firewall, packet processing, and routing functions in the PC than it was to figure out how to achieve compatibility with dozens of proprietary routers, gateways, and firewalls.

Fig. 3 plots PC speed in megahertz against time beginning with Cisco's 1990 IPO. Cisco delivered its first product in 1986, when the PC was too wimpy for network processing. In fact, Cisco had twelve years to establish its business based on proprietary networking gear before the PC was capable of packet processing for a T1 line. It is the story of workstations and the PC all over again with Cisco in the role of the workstation manufacturer and the PC in its usual role. In this case, what is being supplied is performance. The PC and proprietary network gear are improving at a Moore's law rate, but this time the demand function is a series of horizontal lines representing required packet processing rates (fig. 3 shows T1 and T3 packet processing rates for the PC). OpenReach and the PC are coming up from the bottom to displace networking gear at the (high-volume) edge of the network.

While the PC attacks the low end, lambda (wavelength) routers attack the high end, and PLD-based implementations attack the middle. Cisco will soon be under siege by the PC. Cisco, like the workstation companies before it, has an engineering culture based on building performance-oriented, proprietary designs. It is culturally and technically equipped to deal with threats from proprietary designs (the lambda routers and the PLD-based implementations), but it is not equipped to deal with an attack by an open platform (the PC). Moore's law and open systems will show no mercy.

Lessons of Moore's law

In the early years, IC macro functions benefited from Moore's law. As the market grew, IC macros became more complex and they proliferated. Eventually Moore's law enabled the microprocessor; thereafter, Moore's law worked *for* the microprocessor and *against* IC macro functions. IC macros had proliferated; the microprocessor consolidated (many IC macros consolidated into the microprocessor and a few companion parts).

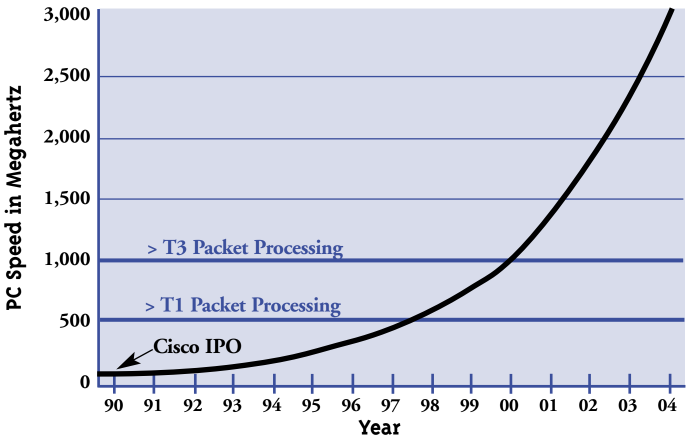


Fig. 3. At Cisco's IPO in 1990, the PC wasn't capable of any significant packet processing. Eight years after Cisco's IPO, the PC became capable of T1 packet processing.

Moore's law supported the microprocessor and the DSP. As the market grew, microprocessors and DSPs got more complex and they proliferated. But Moore's law now works for the PLD as its capacity and performance come up to displace microprocessors and DSPs, first in mobile applications and later in tethered applications.

Moore's law supported the proliferation of proprietary computers and minicomputers, until it enabled the workstation. Then, Moore's law worked *for* the workstations and *against* the minicomputers and mainframes as the workstation's performance came up from below to displace the mainframe and the minicomputer. Moore's law abandoned workstations for the PC as the PC's performance came up from below the workstation to steal the workstation's markets. Proprietary workstation designs proliferated and the PC consolidated.

Cisco is repeating the experience of the workstation manufacturers. Cisco builds proprietary networking gear. The PC's performance will come up from below to steal its markets.

Moore's law benefits both general-purpose solutions (the microprocessor, the PLD, and the PC) and proprietary solutions (IC macros, minicomputers, workstations, network processors), but it works best for general-purpose solutions. Proprietary solutions have better initial performance, but general-purpose solutions evolve faster. General-purpose solutions evolve faster because higher unit volumes attract more competitors.

Nick Tredennick and Brion Shimamoto
March 7, 2001

Dynamic Silicon Companies

The world will split into the tethered fibersphere (computing, access ports, data transport, and storage) and the mobile devices that collect and consume data. Dynamic logic and MEMS will emerge as important application enablers to mobile devices and to devices plugged into the power grid. We add to this list those companies whose products best position them for growth in the environment of our projections. We do not consider the financial position of the company in the market. Since dynamic logic and MEMS are just emerging, several companies on this list may be startups. We will have much to say about these companies in future issues.

Altera and Xilinx (ALTR http://www.altera.com) (XLNX http://www.xilinx.com)
Altera and Xilinx together dominate the programmable logic business, with almost seventy percent of the CMOS PLD market. Both companies are aggressive and competitive. Sixty-six percent of Altera's revenue comes from the rapidly growing communications segment (Telecosm companies) and an additional sixteen percent comes from the electronic data processing (EDP) segment. Altera and Xilinx are positioned to be major suppliers in tethered applications such as the base stations that support mobile devices.

Analog Devices (ADI http://www.analog.com)
Analog Devices is a leader in analog electronics for wireless RF and communication, MEMS for automotive applications (accelerometers, pressure sensors, transducers), and in DSPs.

ARC Cores (ARK (London) http://www.arccores.com)
ARC Cores makes configurable processor cores. Configurable processors allow the application engineer to adapt the processor's instruction set to the requirements of the problem. Conventional microprocessors have fixed instruction sets.

Cypress (CY http://www.cypress.com)
Cypress Microsystems builds components for dynamic logic applications. Cypress also builds MEMS and is a foundry for MEMS.

QuickSilver Technology, Inc. (* http://www.qstech.com)*
QuickSilver has the potential to dominate the world of dynamic logic for mobile devices (untethered). While many companies work on programmable logic and on "reconfigurable computing" for tethered applications, QuickSilver builds adaptive silicon for low power mobile devices.

SiRF (* http://www.SiRF.com)*
SiRF builds RF GPS chips for the mobile market. It is a world leader in development of integrated GPS receivers.

Transmeta (TMTA http://www.transmeta.com)
Transmeta makes new generation microprocessors that use closed-loop control to adapt to problem conditions in an x86-compatible environment. This enables Transmeta's microprocessors to save power over conventional microprocessors from AMD and Intel. The base instruction set is not available to the application engineer.

Triscend (* http://www.triscend.com)
Triscend builds microcontrollers with configurable peripheral functions and with configurable inputs and outputs. Triscend helps consolidate the microcontroller market into high-volume, standard chips.

Technology Leadership	Company (Symbol)	Reference Date	Reference Price	2/28/01 Price	52-Week Range	Market Cap.
General Programmable Logic Devices (PLDs)	Altera (ALTR)	12/29/00	26.31	23.13	19.62 - 67.12	11.7B
Dynamic Logic for Mobile Devices	QuickSilver Technology, Inc. (none*)	12/29/00				
MEMS Foundry, Dynamic Logic	Cypress (CY)	12/29/00	19.69	19.58	16.55 - 58	2.6B
RF Analog Devices, MEMS, DSPs	Analog Devices (ADI)	12/29/00	51.19	41.45	36.71 - 103	15.2B
Configurable Microprocessors	ARC Cores (ARK**)	12/29/00	£3.34	£1.71	£1.21 - 4.58	£791.6M
Field Programmable Gate Arrays (FPGAs)	Xilinx (XLNX)	2/28/01	38.88	38.88	35.25 - 98.31	14.6B
Configurable Microcontrollers (Peripherals)	Triscend (none*)	2/28/01				
Silicon for Wireless RF, GPS	SiRF (none*)	12/29/00				
Microprocessor Instruction Sets	Transmeta (TMTA)	12/29/00	23.50	20.38	17 - 50.88	3.0B

*QuickSilver and SiRF are pre-IPO startup companies.

** ARK is currently traded on the London Stock Exchange