Written by Nick Tredennick

DynamicSilicon

Published by Gilder Publishing, LLC

Vol. 2, No. 9 | The Investor's Guide to Breakthrough Micro Devices | September 2002

Lessons From the PC

The worldwide market for personal computers has grown to 135 million units annually. Personal computers represent half of the worldwide revenue for semiconductors. In July of this year, PC makers shipped their *billionth* PC.

I trace the story of the personal computer (PC) from its beginning. The lessons from the PC apply to contemporary products such as switches, routers, network processors, microprocessors, and cell phones. The story doesn't repeat exactly because semiconductor-process advances change the rules.

PC beginnings

Intel introduced the first commercial microprocessor in 1971. The first microprocessors were designed solely as cost-effective substitutes for numerous chips in bills of material. But it wasn't long before microprocessors became central processing units in small computer systems. The first advertisement for a microprocessor-based computer appeared in March 1974. Soon, companies, such as Scelbi Computer Consulting, MITS, and IMSAI, offered kit computers. Apple Computer incorporated in January 1977 and introduced the Apple II computer in April. The Apple II came fully assembled, which, together with the invention of the spreadsheet, changed the personal computer from a kit hobby to a personal business machine.

In 1981, IBM legitimized personal computers by introducing the IBM Personal Computer. Once endorsed by IBM, many businesses bought personal computers. Even though it came out in August, IBM sold 15,000 units that year. Apple had a four-year head start. When IBM debuted its personal computer, the Apple II dominated the market. Worldwide shipments for all personal computers totaled 900,000, so IBM held a tiny fraction of the market. Apple bought ads welcoming IBM to the market, essentially teasing IBM for being late.

Manufacturers cloned the IBM PC within a year of its introduction. A flood of clone makers followed. IBM had to compete with the clone makers on price and on performance. Former Texas Instruments engineers formed Compaq Computer in 1982 and shipped their IBM-PC-compatible Compaq Portable early in 1983. IBM let the PC-clone market develop; Apple kept its designs proprietary by suing clone makers. The IBM-compatible PC was an "open system" because any manufacturer could build one and sell it. Apple's PCs were "closed" systems.

The IBM Personal Computer XT, a PC with a hard disk, shipped in 1984. The same year, Apple shipped Macintosh. Macintosh competed with the IBM PC, as a proprietary design. Since Apple controlled Macintosh design, Apple had a choice of pursuing market share or of forfeiting market share for higher margins. Apple chose higher margins.

In 1987, IBM introduced the Personal System 2 (PS/2) and attempted to regain control of the PC. The PS/2 Micro Channel bus was a higher-performance, but proprietary, replacement for the PC's original ISA (Industry Standard Architecture) peripheral bus. IBM did not share the PS/2's Micro Channel bus with the industry. A consortium of PC makers countered with an open standard, the EISA (Extended ISA) bus. The two standards fought in the marketplace. IBM lost.

The first model of a product is typically made of proprietary components so that it comes together reliably. After experience in the marketplace, off-the-shelf components replace proprietary ones and lower the cost. But IBM built its original PC with off-the-shelf components. None of the *pieces* brought a special advantage. The value was only in the *collection*. As PC sales grew, the number of competitors grew too. The large

In This Issue:

The PC won. Intel and Microsoft won. Apple and the workstation makers—hardware monopolies—all lost. The details belong to the PC, but their significance does not. I tell the story and I apply the lessons to today's situations.

number of PC makers, combined with the high sales volumes, justified manufacturing many electronic components in one step. This component consolidation lowered the cost in the bill of materials for all the PC makers.

As the market grew, the PC standard solidified. Margins migrated from the system makers to the makers of key components—the x86 microprocessor and the Windows operating system.

Intel established the microprocessor as a key PC component. Intel defeated non-x86 microprocessor suppliers because it knew that a successful business had to sustain Moore's law progress, not have the neatest microprocessor. As a microprocessor design, the x86 was garbage, but that didn't matter. Intel knew the money from x86 sales would buy manufacturing plants that would give the x86 marvelous speed. More speed meant more sales, more sales meant more money, more money meant an even better plant, and so on. The key was to start with something that already had lots of buyers and to design the next version with features that would appeal to the largest number of people. Intel *captured* the microprocessor crown by virtue of its belief in the operation of Moore's law. Intel *stayed* on top by competent brand positioning.

Microsoft established the Windows operating system as a key PC component. Microsoft's success stemmed from its belief that independent software vendors (ISVs) were king. Courting and supporting ISVs was all that mattered. Customers (end users) didn't matter. Customers are drawn by applications. Applications come from ISVs. If you have the ISVs, you have the customers. Microsoft induced a large number of the world's ISVs to write software that is dependent on Microsoft operating systems. Linux is a threat to Microsoft because it is an *ISV* movement.

The PC is a general-purpose system. The PC is difficult to use and it doesn't do anything particularly well. It's easy to beat its performance or its price with an application-specific system. The PC's killer feature, and the key to its longevity, is its versatility. The key to its future is that it built the Web. The simplest road to compatibility with file formats, browsers, browser plug-ins, and applications is through an x86-based system.

When the PC entered the market, the response (access) time of its memory chips was about the same as the request time (clock speed) of its 4.77-MHz 8088 microprocessor. For a few years, memory chips kept pace as microprocessors got faster. These were critical years for PC manufacturers, as these were the years that they trained their customers to buy computers by noting the *microprocessor's* clock speed. But designs for memory chips and designs for microprocessors optimize differently. Designers build memory chips to hold a lot—designers take what they can get in speed. Designers build microprocessors for speed, period. Today's memory chips hold 4,000 times more than 1981's memory chips. But today's memory chips are only seven times faster than 1981's. Today's microprocessors are 500 times the speed of the 8088. Unfortunately, memory chips haven't gotten faster as fast as they have gotten bigger. In 1981, a memory access took one microprocessor clock cycle. Today, a memory access takes 70 microprocessor clock cycles (up substantially from just a few months ago as Intel and AMD continue their clock-speed race).

The rift between the speed of the microprocessor and the speed of its memory system grows, requiring increased sophistication in the memory hierarchy (expensive caches). The overall result is diminishing returns in performance for microprocessor clock speed advances. Hard-disk design also chooses capacity over speed, adding to the problem.

When the PC came out, it was better than nothing and its performance was lacking. But with each new PC generation, the PC got faster. For years that still wasn't enough, because users' expectations grew too. Supply and demand. The PC supplied performance attempting to meet the market's demand. While the demand and its rate of increase are hard to measure, here's my view on the nature of demand for PC performance. First, early adopters have higher performance expectations than late adopters (nerds expect more from their PCs than office workers). Second, as time goes by, performance expectations are not likely to be rising as fast as Moore's law improves the PC's components. This is because late adopters constitute the growing segment of users. Third, unlike what engineers like to think, there's no necessary connection between the rate of improvement in system features (e.g., capability, performance) and the rate of increase in demand.

DynamicSilicon

Editors	**Nick Tredennick** **Brion Shimamoto**
Art Director	**Charles Bork**
Managing Editor	**Marie Lavinio**
Chairman	**George Gilder**

Dynamic Silicon is published monthly by Gilder Publishing, LLC. Editorial and business address: 291A Main Street, Great Barrington, MA 01230. Editorial inquiries can be sent to: bozo@gilder.com.

Intel

When the PC market was young, Intel didn't pay much attention. The Intel 8088 microprocessor, used in the original IBM PC, was designed for embedded applications. Intel had no interest in building computers. More than two years elapsed between the introduction of the Intel 8088 and the introduction of the IBM PC. Intel announced the 286 microprocessor in 1982; IBM's 286-based PC AT shipped in 1984. Intel announced the 386 microprocessor in 1985; this time, Compaq shipped the first 386-based PC in 1986. By 1986, the PC was becoming important to Intel. Waiting a year, from the introduction of a new microprocessor to the development of new motherboards designed around the new microprocessor, became unacceptable. Although the margins for motherboards are low relative to the margins for PC microprocessors, Intel entered the motherboard business. Intel did this to shorten the time between the announcement of a new PC microprocessor and the availability of systems based on the microprocessor. But Intel found it could now set the direction for the evolution of PC systems. By 1989, the year Intel brought out the 486 microprocessor, PCs based on the new microprocessor shipped the same year. Today, Intel and its manufacturing partners announce microprocessors and systems at the same time.

Since the PC's introduction, Intel has been the dominant supplier of x86 microprocessors. For seventeen years, Intel offered a single line (same basic design) of microprocessors from the low end to the high end. The line would be six or eight microprocessors differentiated by speed (like six or eight models of the same car with different engines and trim). As Intel introduced each new microprocessor or each faster microprocessor, the new high-end microprocessor would occupy the top price position. Other family members slid down one position in cost, with the cheapest microprocessor exiting at the bottom. Intel typically priced its top-of-the-line processor at $600. Prices ranged down to $100, giving the line a six-to-one price range for a performance range of about two-to-one. Customers were willing to pay six times as much for twice the performance of the low-end microprocessor.

Beginning in March of 1998, Intel repositioned its microprocessor line. It split its microprocessor line into Celerons and Pentiums. Celeron microprocessors, with lower prices and lower performance, served the low end of the desktop market. In July, Intel split its microprocessor line again, this time with the introduction of Xeon microprocessors, for high-end servers. Intel has since added a line of microprocessors for mobile applications (laptops). Instead of a single line spanning a price and performance range for all applications, Intel now has four microprocessor lines. Introducing a new high-end Xeon microprocessor no longer forfeits margins in desktop, laptop, or low-end computers as prices for these microprocessors are no longer tied to one set of uses. Intel could now charge thousands of dollars for its Xeon microprocessors while it priced Celeron microprocessors under a hundred dollars, to compete in the "value PC" market against VIA Technologies and AMD.

What is an x86?

"x86" refers to the precise collection of one-and-zero sequences that the microprocessor has as its repertoire of commands (instructions). Example instructions are "Add," "Compare," and "Store." Each is represented by unique patterns of ones and zeroes. The same patterns (instructions) are understood by Pentium, by 486, by 386, by 286, and by 8086 microprocessors. PC software consists of millions of lines of such instructions, expressed in the x86-required patterns. Different (e.g., non-x86) microprocessors have different sets of instructions and have different one-and-zero representations, hence the incompatibility among software written for different systems.

RISC workstations

In 1981, Dave Patterson and Dave Ditzel at the University of California, Berkeley started what became a fad in microprocessor design. It began with their paper on "reduced instruction-set computing" (RISC) at the Eighth Annual Symposium on Computer Architecture.

The paper said that how the information was organized in a microprocessor—how the river of information flowed through it—mattered more than the electrical properties of the silicon that made up the chip itself. Since silicon real estate was valuable (transistor densities were nowhere near today's), this was an important statement. Designers could make up for a lack of space on the chip by being clever. Designers split into RISC and CISC (complex instruction-set computing) factions. The CISC camp felt that by targeting a high-volume platform like the PC—one that *already* had customers—they would sell a lot of microprocessors. This would get them the money to build a more sophisticated chip-making plant—one that turned out silicon chips with better electrical properties—enabling faster and denser circuits. This would result in faster PCs. Faster PCs meant more sales, therefore, more microprocessor chips sold, and so on. The RISC camp felt that being clever—by applying two-score of the latest ideas in computer science—would yield the fastest microprocessors. And even though the RISC computers wouldn't run PC software—and the software they did run was hugely expensive by PC standards—the performance would compel people to switch. A way to state the two arguments is to ask a question. Does (high sales) volume beget performance or does performance beget volume?

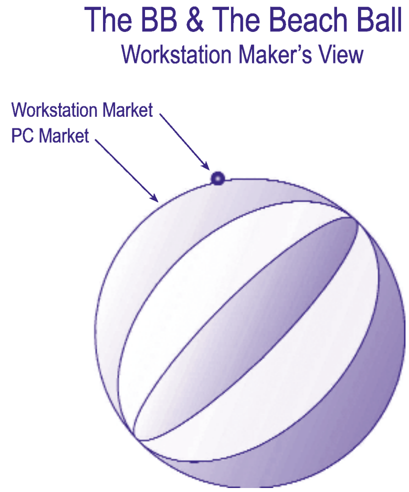
Thus, microprocessor design became a hot topic at universities and triggered an avalanche of papers on the RISC idea. The RISC mire caught the microprocessor manufacturers. IBM (801, POWER, PowerPC), Intel (i860), Motorola (88000, PowerPC), AMD (29000), DEC (Alpha), Sun (SPARC), MIPS (MIPS), HP (PA-RISC), and others built RISC microprocessors. The battle in the marketplace became RISC vs. CISC. CISC was represented by Motorola's MC68000-family microprocessors and by Intel's x86-family microprocessors. PowerPC (RISC) displaced the MC68000 (CISC) in Apple's Macintosh. SPARC (RISC) displaced the MC68000 in Sun's workstations. Sun and other workstation makers and Apple could change the microprocessors in their closed systems; the PC's open system retained the x86.

Workstation makers showed the power of fads in the semiconductor industry.

When the IBM PC came on the scene, microprocessors had been evolving for more than ten years. Moore's law had had ten years to increase microprocessor circuit speeds and circuit densities. Microprocessors became powerful enough to challenge the custom processors used at the core of the minicomputer business. Sun Microsystems introduced its Sun "workstation" in 1982. Workstations did not run PC software because workstations were not x86 compatible.

For engineering design, microprocessor-based workstations were more cost effective than minicomputers. Workstations, from manufacturers such as Sun, Apollo, DEC, HP, Silicon Graphics, MIPS Computer Systems, and Intergraph, proliferated; minicomputer manufacturers began their decline.

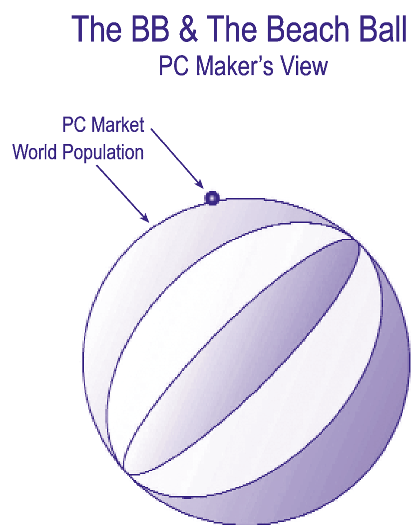
Workstation manufacturers began designing and manufacturing their own microprocessors, thinking that was the path to high performance and the path to independence from the large microprocessor suppliers. But the workstation manufacturers couldn't sell enough workstations to justify the cost of a proprietary microprocessor. In the diagram I call the "BB and the Beach Ball," workstation manufacturers held the BB-sized market and looked with envy upon the beach-ball-sized PC market. Beginning with significantly higher performance than PCs, workstation makers thought they could cost-reduce their systems by using PC-compatible hardware components (video cards, hard disks, floppy drives, etc.). These cost-reduced systems would have better price-performance than the PC and would, therefore, take market share from the PC.



Meanwhile, PC makers saw the "BB and the Beach Ball" diagram, but they viewed it with different labels. In the PC-makers' version, the BB represented the PC's installed base, and the beach ball represented the *world population*. PC makers eyed this "total available market" with envy and plotted to penetrate it. One strategy for penetrating the available market reduces cost to reach the most customers. This is building for volume. Since the PC market, even as represented by the BB, was a hundred times the size of the workstation market, PC makers ignored the workstation market.

Perhaps ten years ago, a workstation company invited me over for a consulting interview. If it went well, I'd consult regularly with the company's strategic planners. The planners thought they could build systems with PC hardware—disk drives, video controllers, adapter cards, keyboards, etc. "We can build RISC-based systems that are as cheap as PCs and that have twice the performance. What do you think?"

"I don't think so. PC makers like Dell and Gateway ship so many systems that they negotiate much better deals with their suppliers than you will ever get." That interview ended my consulting for the company. Successful consultants divine and endorse their employers' thinking.



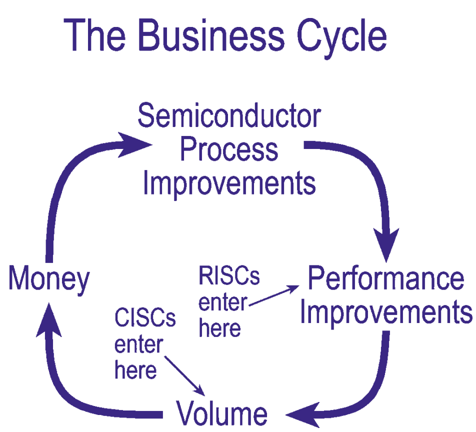
The strategy for Intel's x86 microprocessor designs and for the PC makers: build for volume and let Moore's law yield performance. The strategy for workstation makers with their proprietary microprocessors: build for performance and price-performance will capture market share. Intel controlled the microprocessor's cost by setting a chip-size limit for the design team; workstation makers designed for maximum performance and let the result determine chip size. The build-for-volume strategy of Intel and of the PC makers trounced the workstation makers' build-for-performance strategy. Workstation makers did not encroach on the PC market. Instead, PCs squeezed workstation makers into even smaller niches as PCs increased in performance.

As PC sales soared, PC makers proliferated. "Screwdriver shops" assembled PCs from readily available components. High-volume manufacturers, such as IBM, HP, Compaq, Dell, and Gateway, emerged. As the PC market grew, economies of scale overwhelmed the screwdriver shops. Large PC manufacturers, buying components by the *trainload*, paid much lower prices than screwdriver shops, which bought components by the hundreds. Large PC manufacturers demanded that their suppliers' warehouses be close to the manufacturing plant—reducing the manufacturer's inventory, lowering shipping costs, and shortening the time from order to component delivery. I'm the ultimate screwdriver shop; I build my own PCs from components I buy from Fry's Electronics or through Pricewatch on the Web. Even with my free labor, Dell's PCs are cheaper.

Generic systems, such as the IBM-compatible PC, eventually displace proprietary systems. Here's how it happens. The PC proliferates while the market for proprietary systems, such as workstations, grows slowly. Software developers look at the effort and at the installed base to decide what systems to support. If the installed base is small, the development is too costly to justify. As PCs invade workstation applications, it becomes more appealing for developers to write for the PC than it does to write for the proprietary workstation. Even Apple, which still has a measurable share of the personal computer market, will lose developers because Apple's share of the market is declining.

Some years ago, I talked to a software developer who had been to see Apple when it had a larger share of the PC market than it does today. He waited in an outer office for an audience. As he sat there, he realized that his software was installed on more IBM-compatible PCs than were represented by Apple's share of the personal computer market. Why should he develop for Apple computers when he could develop product enhancements for his software's installed base and could reach a larger and more receptive audience? The community of software developers for the PC is large and growing; the community of software developers for workstations and for other proprietary designs is small and fragmented. This situation makes it easier to develop software for the PC and it makes it difficult and expensive to develop software for workstations.

Manufacturers built RISC microprocessors for performance and hoped for volume. Intel built for volume (to serve the growing PC market) and did what it could for performance since it worried about competition from RISC microprocessors. By building for volume, Intel amortized research and development cost over more microprocessors. Intel could spend more on development, and it could develop and move to new semiconductor processes sooner. The PC's volume strategy, based on Moore's law improvements in its microprocessor, defeated the workstation's performance strategy. Intel's volume-based strategy thoroughly defeated the RISCs' performance-based strategy.



Taking an ironic turn, Intel, still worried about the RISC camp, developed a RISC-style chip, the 64-bit Itanium microprocessor. Itanium is a *non-x86* design that has an "x86 mode." When the Itanium runs PC software, it is trying to be something that it is not. It is inefficient at running x86 software compared to an x86 microprocessor. AMD, in contrast, staying with an x86 base, has defined 64-bit extensions to the x86 instruction set. High-end servers use 64-bit features. AMD should win this contest handily.

Lessons from the PC

A new use for a chip can change its design objective. Before the PC, microprocessors were designed for embedded applications, which meant that they were first, cheap, that they worked with a variety of support chips, and that they performed only adequately. After the PC, speed became the primary design goal in microprocessor design.

New areas require backing from a market leader. The PC didn't take off until IBM endorsed it. IBM's imprimatur meant it was safe to build upon.

"Open" works better in the marketplace than "proprietary." With open systems, competition decides winners. With proprietary systems, corporate executives make self-destructive mistakes. Open systems and proprietary systems are like market-driven economies and centrally planned economies, respectively. Market forces drive the evolution of open systems, while fallible central planners drive proprietary systems. In addition, development costs for open-system components are shared across companies, while suppliers of proprietary systems shoulder all development costs.

As the market grows, margins migrate from the system to key components in the system. In the PC, the key components are the x86 microprocessor and the Windows operating system. These components became key *because of who was doing them, not because of what the components were*. The microprocessor and the operating system did not become key components because of something inherent. Rather, it was insights and beliefs that Intel and Microsoft held. It's market positioning and not technology that's important (see below).

Unforeseen applications can grow to dominate markets. PCs grew from "Why would anyone want one?" to half of the world's semiconductor revenue.

Supply and demand grow at different rates. For semiconductors, what is being supplied (transistors, function, capacity, performance) grows with Moore's law. Demand is unlikely to grow as rapidly, so supply will cross the demand line, precipitating change. Today, for example, the PC's (supply of) performance has crossed the demand line for most users. This means longer upgrade cycles and buyer preference for value PCs over high-end PCs.

Market success may lead to unexpected business opportunities. The desire to shorten the time from microprocessor announcement to system availability drove Intel into the PC motherboard business.

Brand positioning can establish a system's key components. "Intel Inside" established the Intel x86 as the real thing for PCs, keeping other x86 microprocessors out. Finally, brand positioning with Xeon, Pentium, and Celeron enabled Intel to collect high margins on its Xeon and Pentium microprocessors while it fought x86-compatibles for market share at the low end. While Itanium is a technical mistake, brand positioning will protect the other Intel microprocessor lines by containing the damage to Itanium-branded microprocessors.

The semiconductor industry is as subject to fads as are the fashion and toy industries. The RISC fad diverted microprocessor design resources across the industry for twenty years.

Moving up to more demanding customers is easier than moving down to less demanding customers. As minicomputers moved upmarket to more sophisticated customers, they abandoned their low-end customers to microprocessor-based workstation manufacturers. The microprocessor's performance grew more rapidly than the minicomputer makers could move up. Workstations thus overtook and wiped out minicomputers. Few minicomputer makers made the transition from custom, many-chip central processors to microprocessor-based workstations.

Growing the customer base beats competing for market share of a fixed base. Workstation makers focused on taking market share from PCs. PCs focused on increasing their penetration among nonusers. The PC's strategy won.

Build for a volume market, and semiconductor manufacturing progress (Moore's law) supplies performance; build for performance, and there is no one to supply volume.

For consumer systems, price-performance matters only at low price points. Price-performance is price divided by system performance and has units of dollars per unit of performance. For a time, workstations were price-performance leaders. It didn't matter. In competition with PCs, price-performance matters, but only at price points too low for workstations to reach. Thousand-dollar PCs at two dollars per performance unit would outsell twenty-thousand-dollar workstations at one dollar per performance unit.

The market expects costs to decrease with time for electronic systems. Escalating microprocessor development cost (design, chip layout, semiconductor process development, mask cost, and fabrication) changed the economics of the microprocessor-based systems business to favor high-volume production (e.g., x86-based systems). That is, costs escalated faster than the market grew for workstations, making amortized development cost an ever-larger portion of the microprocessor's cost. That meant that the system cost rose with time rather than fell, which countered the market expectation that costs for electronic systems decline with time.

Applying the lessons

The purpose of extracting the lessons is to apply them to current situations. Here are a few examples: integrated device manufacturers, foundries, network processors, and cell phones.

Integrated device manufacturers are like the workstation makers. Design cost, semiconductor-process-development cost, mask cost, and fabrication-equipment cost are rising rapidly. Skyrocketing costs are fragmenting the integrated device manufacturers (*Dynamic Silicon*, Vol. 2, No. 4). Similar escalating costs will force changes among workstation makers. While Sun and IBM both sell workstations based on proprietary microprocessor designs, IBM is large enough to subsidize losses in microprocessor design with profits from other businesses. Sun isn't.

A microprocessor has two costs: the cost to manufacture the chip and the cost to develop the chip. Assume the cost to manufacture is $100 and that the cost to develop is $100 million. Selling a hundred million chips means the chip's price includes $101 of cost, which is $100 of manufacturing cost and $1 of amortized development cost. Selling a hundred thousand chips means the chip's price includes $1,100 of cost, which is $100 of manufacturing cost and $1,000 of amortized development cost. The PC market is large and monolithic, which means that manufacturing cost dominates the microprocessor's cost and that amortized development cost is small. The workstation market is about a hundred times smaller, and it's shared among the workstation manufacturers. For workstation makers, the contribution of amortized development cost to the microprocessor's price will be at least a hundred times what it is for PC makers.

In addition to higher cost for its microprocessor, the manufacturer of a proprietary workstation pays the whole cost of board and system development, of BIOS development, and of operating system development. The PC market shares these costs across manufacturers. Since workstation development costs rise faster than the market grows, the problem gets worse with time. PCs will continue to force workstations into smaller niches, and workstations will become more expensive and less competitive with time. Apple, which makes proprietary PCs, has a larger market than the workstation makers, but Apple's competitive position will follow that of the workstation makers.

Foundries of today are like the PC clone makers. The PC market was never vertically oriented to the extent that integrated device manufacturers are because the original PC was built with off-the-shelf components. To the extent that it was vertically organized, however, increasing unit volumes have fragmented the business horizontally. System makers no longer design and build their own motherboards or develop their own BIOS versions.

Network processors are repeating the experience of manufacturers of workstations and of RISC microprocessors. About fifty companies are attempting to build proprietary processors for networking. It's not as if the switch and router companies don't see them coming. Cisco, which dominates the market for switches and routers, isn't about to give margins in its business to an outside company by outsourcing a key component. Meanwhile, PC performance is coming up from low-end routing applications. The "dumb" network will come down from the top. Network processors will squeeze into a niche in the middle.

The PC changed the course of microprocessor design from cost to performance. Mobile devices are changing it again—from performance to efficiency.

Cell phones and PDAs are converging. PDAs connect to the Internet and are telephones. The cell phone market is a hundred times the size of the PDA market; cell phones will force PDAs into smaller niches.

NICK'S SCORECARD: WHO WINS, WHO LOSES

COMPANY	TYPE OF COMPANY	FUTURE POSITION	THE WAY I SEE IT
Altera, Xilinx	Fabless	Excellent	Supply and demand: PLDs come up from below to take applications from ASICs. Build for volume: high volumes for generic components pay for leading-edge semiconductor process.
Chartered, TSMC, UMC	Foundry	Excellent	Open vs. proprietary: foundries share equipment costs and process development costs across a customer base.
Legend Group Limited	PCs	Good	Build for volume: build for volume markets and move upmarket with Moore's law.
National Semiconductor	Integrated device manufacturer	Good	Open vs. proprietary: open systems based on x86 will proliferate in embedded applications. Build for volume: high volumes for x86-based applications lead to higher performance.
QuickSilver Technology	Fabless	Good	Build for volume: high volumes for generic components pay for leading-edge semiconductor process.
Transmeta, VIA Technologies	Fabless	Good	Open vs. proprietary: open systems based on x86 will proliferate. Build for volume: high volumes for x86 components lead to higher performance.
Triscend	Fabless	Good	Build for volume: build generic microcontrollers for volume markets and move upmarket with Moore's law.
LSI Logic	ASIC supplier	Struggle	Supply and demand: ASIC performance and capacity are moving up faster than demand for performance or for capacity grows.
Intel, Motorola	Integrated device manufacturer	Struggle	Open vs. proprietary: integrated device manufacturers shoulder the entire burden of semiconductor plant and of process development.
Apple, Sun Microsystems	Proprietary computer systems	Fail	Open vs. proprietary: development costs escalate too rapidly to support proprietary systems.

The "position for the future" and "the way I see it" apply only to the topic of the issue. Possible positions for the future are: excellent, good, OK, struggle, and fail. A company that is "excellent" with respect to horizontal fragmentation of an integrated business may, for example, "struggle" with cultural obstacles in another technical transition. A company listed as "struggle" in another issue could be listed as "good" in this issue since issues cover different topics.

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, some companies on this list are startups.

Company (Symbol)	Technology Leadership	Reference Date	Reference Price	8/30/02 Price	52-Week Range	Market Cap.
Altera (ALTR)	General Programmable Logic Devices (PLDs)	12/29/00	26.31	10.71	10.19 - 31.05	4.01B
Analog Devices (ADI)	RF Analog Devices, MEMS, DSPs	12/29/00	51.19	24.10	19.57 - 52.74	8.81B
ARC Cores (ARK**)	Configurable Microprocessors	12/29/00	£0.34	£0.25	£0.20 - £0.64	£0.73M
ARM Limited (ARMHY***)	Microprocessor and Systems-On-Chip Cores	11/26/01	16.59	7.23	5.55 - 19.20	2.43B
Calient (none*)	Photonic Switches	3/31/01				
Celoxica (none*)	DKI Development Suite	5/31/01				
Cepheid, Inc. (CPHD)	MEMS and Microfluidic Technology	12/17/01	4.73	3.69	1.48 - 11.48	113.2M
Chartered Semiconductor (CHRT)	CMOS Semiconductor Foundry	7/31/01	26.55	11.85	12.79 - 30.36	1.64B
Coventor (none*)	MEMS IP and Development Systems	7/31/01				
Cypress (CY)	MEMS Foundry, Dynamic Logic	12/29/00	19.69	10.53	9.45 - 26.20	1.30B
Cyrano Sciences, Inc. (none*)	MEMS Sensors	12/17/01				
Energy Conversion Devices (ENER)	Ovonic Unified Memory	6/18/02	27.69	12.07	9.47 - 25.73	264.3M
Flextronics International (FLEX)	Contract Manufacturing	8/6/02	7.68	9.47	5.85 - 29.99	4.89B
Foveon (none*)	CMOS Imaging Chips	6/18/02				
Legend Group Limited (LGHLY.PK)	PCs and Consumer Electronics	8/6/02	6.63	7.25	N/A	N/A
Microvision (MVIS)	MEMS-based Micro Displays, Nomad Head-Worn Display, Scanners	6/18/02	6.80	4.50	2.64 - 17.59	65.1M
National Semiconductor (NSM)	Geode x86 Microcontrollers, Consumer Orientation, 51% Ownership of Foveon	6/18/02	32.30	15.99	15.44 - 37.30	2.89B
QuickSilver Technology, Inc. (none*)	Dynamic Logic for Mobile Devices	12/29/00				
SiRF (none*)	Silicon for Wireless RF, GPS	12/29/00				
Taiwan Semiconductor (TSM†)	CMOS Semiconductor Foundry	5/31/01	14.18 ††	8.17	7.05 - 19.08	30.19B
Tensilica (none*)	Design Environment Licensing for Configurable Soft Core Processors	5/31/01				
Transmeta (TMTA)	Microprocessor Instruction Sets	12/29/00	23.50	1.21	0.85 - 4.47	161.8M
Triscend (none*)	Configurable Microcontrollers (Peripherals)	2/28/01				
United Microelectronics (UMC†)	CMOS Semiconductor Foundry	5/31/01	10.16	4.47	3.70 - 10.02	11.92B
VIA Technologies (2388.TW)	x86 Microprocessors for "Value" PCs	6/15/02	78.00	58.50	58.00 - 156.00	N/A
Wind River Systems (WIND)	Embedded Operating Systems	7/31/01	14.32	5.11	4.01 - 20.14	403.7M
Xilinx (XLNX)	General Programmable Logic Devices (PLDs)	2/28/01	38.88	19.32	15.77 - 47.16	6.52B

† Also listed on the Taiwan Stock Exchange

†† TSM reported a stock split on 6/29/01. The Reference Price has been adjusted for the split.

* Pre-IPO startup companies.

** ARK is currently traded on the London Stock Exchange

*** ARM is traded on the London Stock Exchange (ARM) and on NASDAQ (ARMHY)

NOTE: This list of Dynamic Silicon companies is not a model portfolio. It is a list of technologies in the Dynamic Silicon paradigm and of companies that lead in their application. Companies appear on this list only for their technology leadership, without consideration of their current share price or the appropriate timing of an investment decision. The presence of a company on the list is not a recommendation to buy shares at the current price. Reference Price is the company's closing share price on the Reference Date, the day the company was added to the table, typically the last trading day of the month prior to publication. The authors and other Gilder Publishing, LLC staff may hold positions in some or all of the companies listed or discussed in the issue.