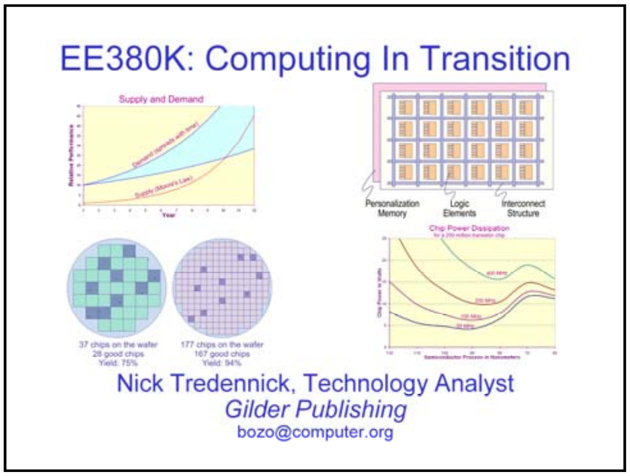
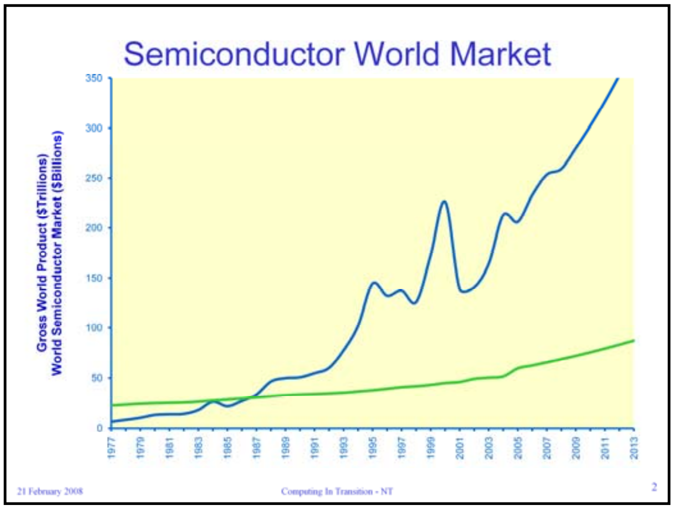


EE380K: Computing In Transition

Nick Tredennick, Technology Analyst
*Gilder Publishing*
bozo@computer.org

Since shortly after its introduction, the microprocessor has dominated the design of electronic systems. The success of the microprocessor, sustained by the march of Moore’s law, stalled innovation in logic design for more than thirty years because programming became a substitute for hardware design. This was possible because the design goal of the personal computer and other microprocessor-based systems, representing the majority of the semiconductor market, was cost-performance. The advent of the value PC and the burgeoning of mobile devices have conspired to change the design goal to cost-performance per watt. Traditional microprocessor-based design cannot meet the challenge of the new design goal, so computing is in transition. A host of multiprocessor configurations and a host of reconfigurable systems vie for control of the next generation of computing applications. Computing is in transition, but the outcome is currently unpredictable.



Semiconductor World Market

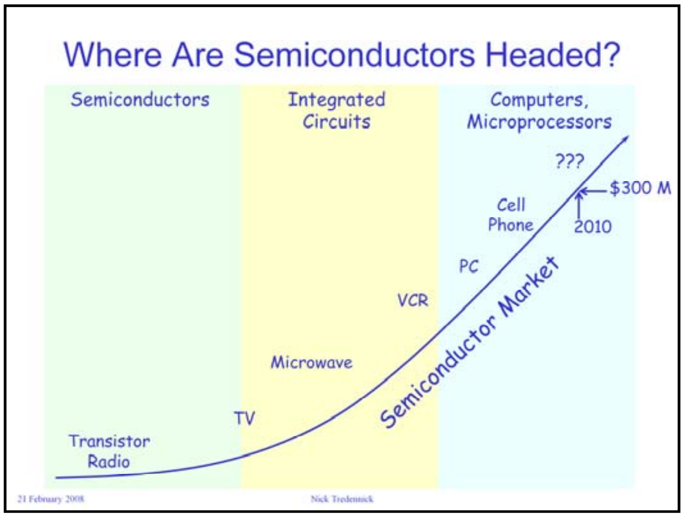
21 February 2008 Computing In Transition - NT 2

The blue line is the specific data for a generalized curve I will use throughout this presentation. It is the worldwide semiconductor market in billions of dollars. It’s easy to see the “.com crash” of 2000. We are well past its peak today. And the market will continue to grow. I’ve put the green line on this chart for comparison. It is the gross world product (GWP). You might think of that as all the goods and services produced in the world annually. The scale for the gross world product is trillions of dollars. There are three things I want you to notice.

First, the gross world product is growing; in fact, it’s growing rapidly. Until recently, only one third of the world’s population was engaged in the global economy; now it’s two thirds. That’s good for all the participants; wealthier is healthier. Instead of spending four hours a day hauling dung and water for daily living, some kid in east Africa may be able to solve the multiprocessor programming problem or, perhaps easier, cure cancer.

Second, the world semiconductor market is nothing relative to the gross world product. That is, it’s less than one percent of the GWP. The semiconductor market isn’t big enough yet to significantly influence the GWP.

Third, notice the slopes of the two lines. The worldwide semiconductor market is growing at approximately twice the rate of increase in the GWP. That’s good news too; that means semiconductors are invading world markets. It’s good news for you if your major is electrical or computer engineering.

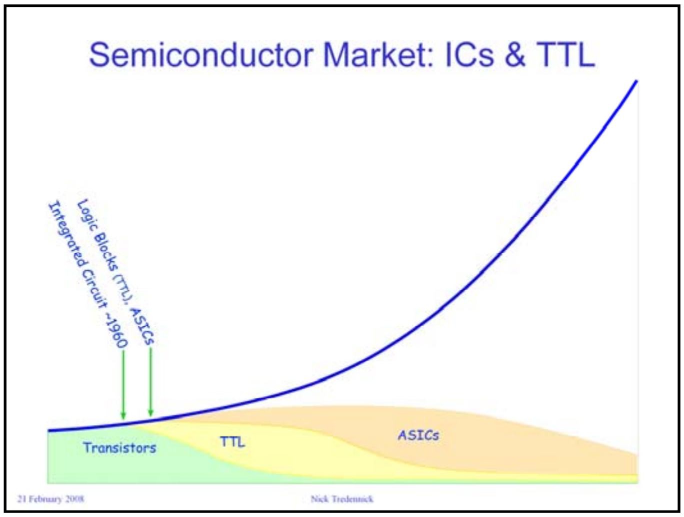


Where Are Semiconductors Headed?

This slide is notional; I just drew it. It’s qualitative, not quantitative. The curve represents the growth of the semiconductor industry in dollars. The colors represent the transition from semiconductors to integrated circuits to computers and microprocessors. None of this is, of course, accurate; I’m here to tell a story. On this background I have added some killer applications as I remember them. I didn’t check for the relative importance of these, but they’re good enough for my purposes. Up through the VCR, the industry didn’t really worry about “killer apps.” That may be because these earlier devices never dominated the industry. With the PC, however, that particular market segment consumed about 40% of the dollar volume of electronics in the industry. That’s a killer app. Likewise, the cell phone, with its unit volumes in the hundreds of millions each year is an obvious killer app. As the PC and the cell phone mature, pundits begin speculating about what will be the next “killer app” to drive growth in semiconductors.

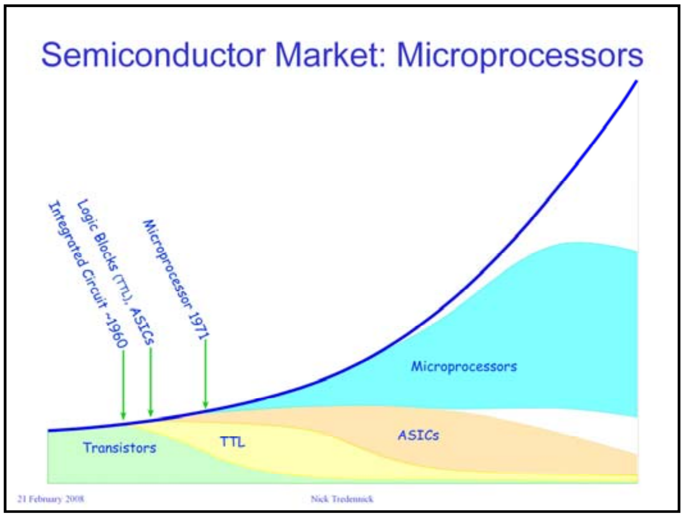
My short answer is: there won’t be one.

Here’s what I think will happen.



Semiconductor Market: ICs & TTL

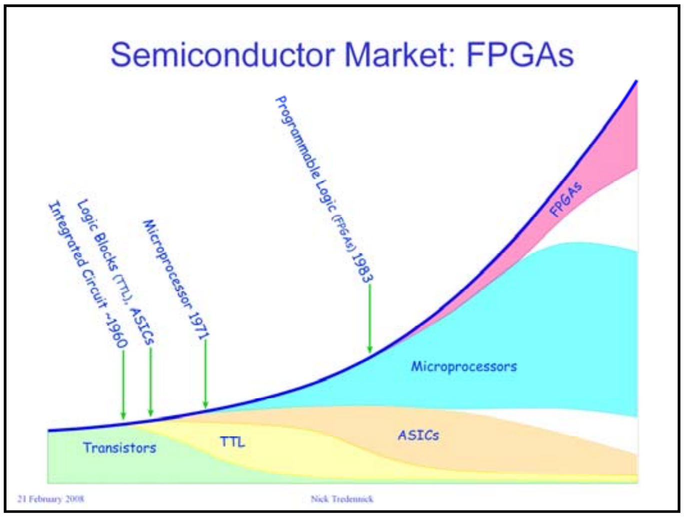
This is the same curve from the previous chart; it shows a stylized version of the semiconductor market. Here, I illustrate which semiconductor components dominated applications in the semiconductor market over time. My experience actually goes back to vacuum tubes, where I saw the transistor wipe out the vacuum tube in short order. With the invention of the integrated circuit, TTL circuits—does anyone remember them?—wiped out the transistor except in some irreducible minimum of applications in such circuits as power amplifiers and analog interfaces. TTL, the Lego blocks of the logic designer's world in the '70s, was wiped out in turn by application-specific integrated circuits (ASICs).



I put the introduction of the microprocessor in 1971 because that's were everyone expects to see it, but the first component that looks to me like a microprocessor was Lee Boysel's AL1 from Four Phase Systems, which appeared in its terminal systems in 1969 (Wikipedia says 1970). An article by Lee Boysel describing the design of the AL1 appeared in the April 1970 issue of *Computer Design* magazine ("Four-phase LSI logic Offers New Approach to Computer Designer," pp. 141-146). Four Phase Systems, which had buildings on the southeast corner of Stevens Creek and de Anza, was bought and destroyed by Motorola. That's a common theme. The effect of that was to leave Intel the liberty to rewrite history. In any event, the microprocessor quickly grew to dominate the market.

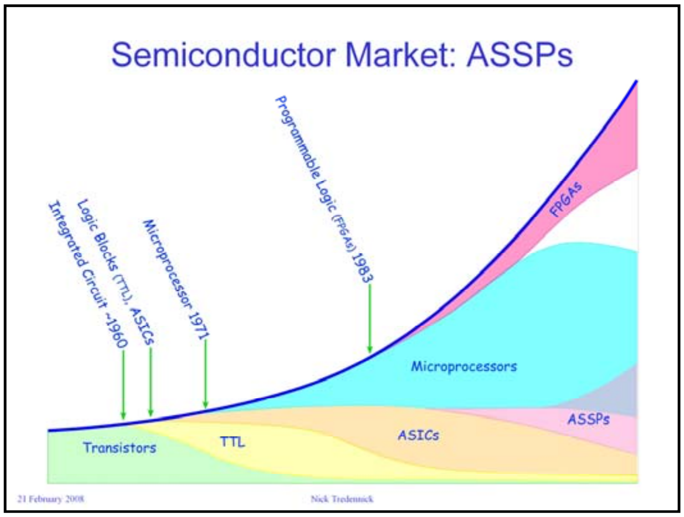
It is easy to look at this chart and say "yes, of course TTL wiped out the transistor." Or "naturally the microprocessor overtook the market for TTL and for ASICs." There are plenty of reasons why what happened in the market is obvious—but only in retrospect. As I'll show in a moment, it matters where you stand when you look at the market.

The introduction of the computer began the end of innovation in logic design; with the introduction of the microprocessor, innovation in logic design came to a standstill. Custom logic design gave way to programming as the way to solve problems. For systems that drew power from wall sockets, it was the most cost-effective method of solving problems. A few standard components, memory, microprocessor, and peripherals, manufactured in huge quantities, sufficed for a broad range of applications.



Semiconductor Market: FPGAs

Programmable logic was introduced to the market in about 1983. I don’t want to get too involved in the details of the various components among GALs, PALs, PLAs, PLDs, CPLDs, and FPGAs, but the two companies that dominate today’s market for field-programmable gate arrays (FPGAs), Altera and Xilinx, were founded in 1983 and 1984, respectively (yes, despite popular notions of programmable logic’s origins, Altera is a year older than Xilinx). In any event, with its introduction , programmable logic began to grow a wedge in the market. I got immersed in the programmable logic market when I became Altera’s Chief Scientist in 1993. From my position deep in the culture and enthusiasm of programmable logic, it looked to me as if programmable logic was poised to take market share from other segments of the industry. Such is the perspective looking forward from within a niche. It retrospect, it looks easy to say why FPGAs didn’t grow to dominate the market.

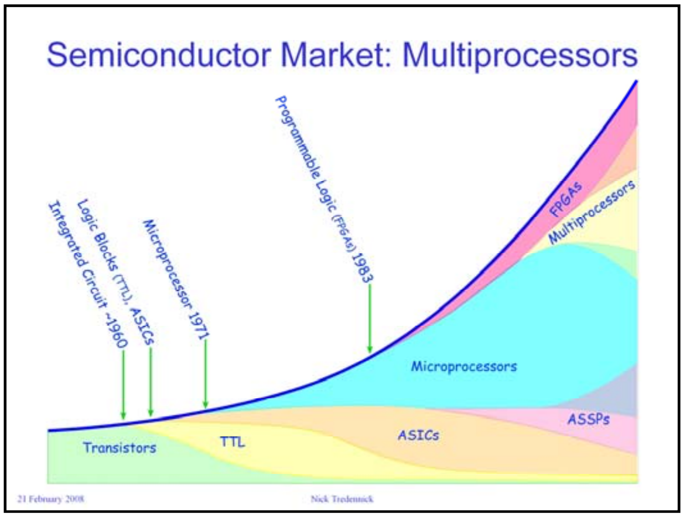


A few more refinements and it will be time to assess today's situation.

An application-specific integrated circuit is a custom design for a particular application for a single original equipment manufacturer (OEM). It has the advantages of excellent performance, unique features, high efficiency, and low per-part cost. The disadvantage is its very high development cost. As components get more complex, running to hundreds of thousands or even billions of transistors, the development cost is astronomical, which limits ASICs to huge dollar-value markets. If the end-system market is ten thousand $1000-list-price units, then it's not cost-effective to spend even a few tens of millions of dollars (a cheap development by today's standards) on custom chip development. As design costs rise, the size of cost-effective end markets rises, narrowing the potential range of applications for an ASIC. As a rule of thumb each semiconductor process generation doubles development cost and therefore implies a doubling of the end market for profitability. This reduces the number of eligible applications with each process generation.

**ASSPs**. One effective way to overcome the design limitation is to design an application-specific standard product (ASSP). A company such as Freescale, Hitachi, or Texas Instruments designs an application-specific standard product that several OEMs use, spreading the high design cost of the custom circuit among several equipment manufacturers. These ASSPs are generally designed to have features that the OEM can set to differentiate the product from competitors that use the same component.

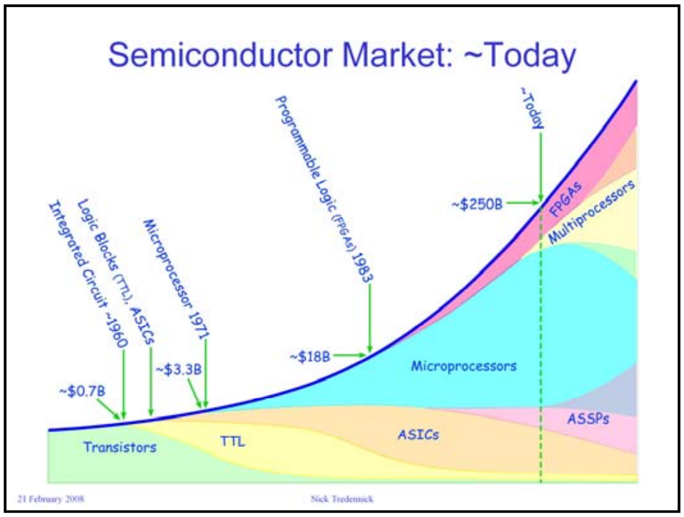
ASSPs are a growing wedge in the market between ASICs and microprocessors—and this is where we begin to be unsure of where the market is headed. There's one more white space to talk about and then we'll begin to assess the situation.



Semiconductor Market: Multiprocessors

21 February 2008 Nick Tredennick

The last wedge is that between microprocessors and FPGAs. I’ve arbitrarily labeled that wedge “multiprocessors.”



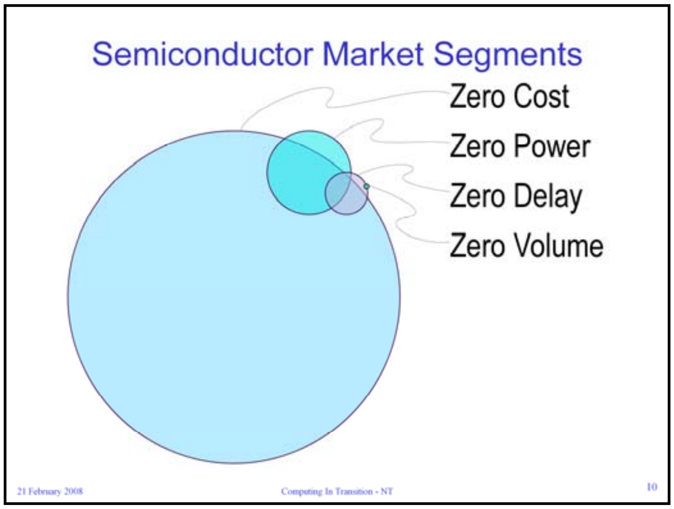
So here's the summary chart for today's situation. It's clearly not on a linear scale, but that's because I wanted to make visual points about the state of the market. We are about at the green vertical line today. That is, the worldwide semiconductor market is about $250 billion and it's on its way to $300 billion. At the invention of the integrated circuit, the worldwide semiconductor market was only $0.7 billion; at the introduction of the microprocessor it was only $3.3 billion; and it was $18 billion the year that Altera was founded.

So, imagine yourself sitting at 2008 looking back at the development of the market. It all looks pretty certain: of course TTL wiped out the transistor, of course ASICs displaced TTL; and of course microprocessors took over the market. Everything has been measured and all the results are in. OK. Now turn and look the other way. What will happen at the border between FPGAs and multiprocessors? Will FPGAs grow market share? Will multiprocessors take market from the microprocessors? What about the ASSPs? Will they take market share from the microprocessors? From the ASICs?

What's the difference between an ASIC and a microprocessor anyway? Do you imagine that there are any ASSPs in the market that don't have one or even several microprocessors embedded in them?

Before I cleverly avoid answering any of those questions, I'll introduce a couple of ways that I look at the market.

A few years from now, when the "Today" line is more to the right, and the unresolved boundaries of 2008 have been measured and quantified, we'll all look back at the hard lines in the figure and it will be obvious to us that the way the market resolved was inevitable.



Semiconductor Market Segments

Zero Cost

Zero Power

Zero Delay

Zero Volume

21 February 2008 Computing In Transition - NT 10

I call this figure the zeroes model of the semiconductor market. It’s meant to illustrate unit volumes or the dollar value of the market for various applications sorted according to the engineer’s design goal. It’s meant to be conceptual and not quantitative.

The zero-cost segment is consumer applications such as automobile transmissions, hair dryers, blenders, and electric toothbrushes. If you are a designer for Delco or Braun, your design goal for a consumer product will be a zero-cost BOM. The consumer market is so intensely competitive that every extra dollar of cost in the BOM translates to significant loss of market share.

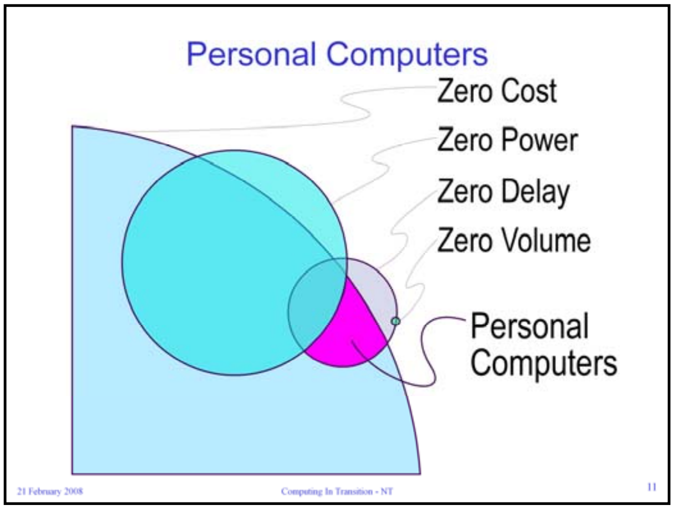
The zero-power segment is applications that want to run forever on weak ambient light. That would be smoke alarms, cell phones, flashlights, wristwatches, and electric toothbrushes. The engineer’s design goal is zero power.

The zero-delay segment is performance-based systems. Anything that wants zero delay from request in to answer out. Head-up displays, interactive displays, and weather-modeling computers fit this design goal.

There’s overlap among these segments, of course. Because they are consumer items, smoke alarms and wristwatches, for example, must meet zero-cost and zero-power design objectives. The cell phone, with its enormous computational requirements, fits in the overlap among zero cost, zero power, and zero delay.

At this point in the design of my model, I thought I was finished. I sent it to John Wharton, who taught EE380K for about fifty years. He sent back a short note saying that I forgot the zero-volume segment. What’s the zero-volume segment? It’s the product for which there will be no demand. How many people think Budweiser needs Clydesdales to deliver beer? Companies design some products to capture headlines instead of market; that’s the zero-volume segment. The designer’s objective is to make trade-press headlines and conference keynotes across the industry.

This diagram shows why the microprocessor has been so successful. The vast majority of applications are in the zero-cost segment. That means that they are sensitive to cost, but require only adequate performance and are essentially indifferent to power. Products can be built of standard components, microprocessor, memory, and peripheral chips, giving these components the enormous volumes that lead to low cost.

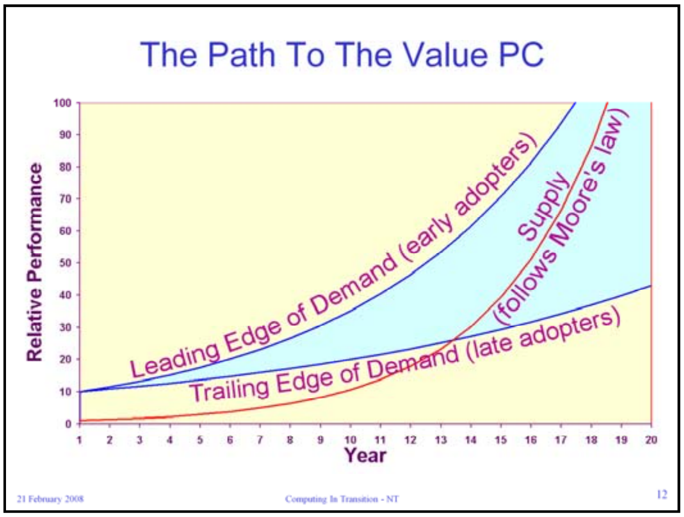


Personal Computers

21 February 2008 Computing In Transition - NT 11

Here’s a figure that zooms in on the zero-cost circle so we can see what the overlap looks like. As I hope to show, this will be important to us in guessing where the industry is headed. Personal computers are the pink wedge in this diagram. They’re consumer items, so they are in the zero-cost segment, but they advertise (or at least they did advertise) and compete on performance, so they are in the zero-delay segment as well. This is important because the personal computer came to dominate the worldwide semiconductor market, at one time holding about 40% of the dollar volume of the market.

When the PC was the killer application, the design goal was cost-performance.



The Path To The Value PC

21 February 2008 Computing In Transition - NT 12

Here’s the reason the PC is no longer the killer app that it once was. It’s in the margins and it requires a short lesson in economics and markets.

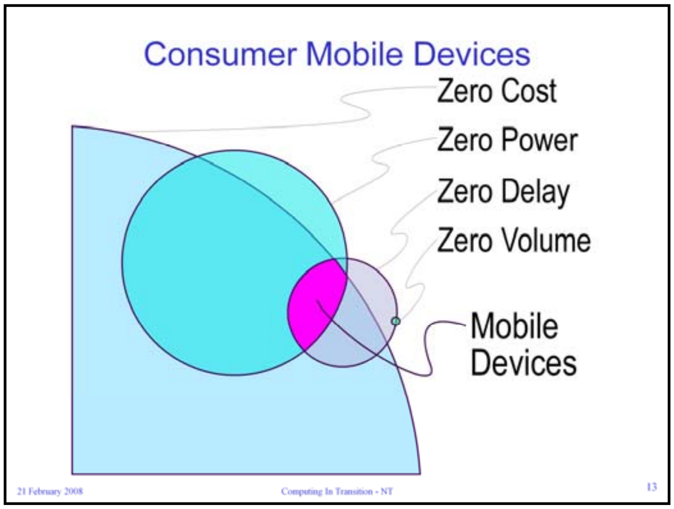
For a long time, the PC didn’t enough performance to satisfy anyone.

Industry inertia is on performance and assumes that function comes in the wake of delivering performance.

The big change is that for the first time we can see a point when there’s enough performance for the vast majority of users.

The quest for performance will no longer drive development.

When the PC became a commodity, profits evaporated. When profits evaporated, it was time for the industry to move to something new and more profitable. That’s when it’s time to look for the next killer app.



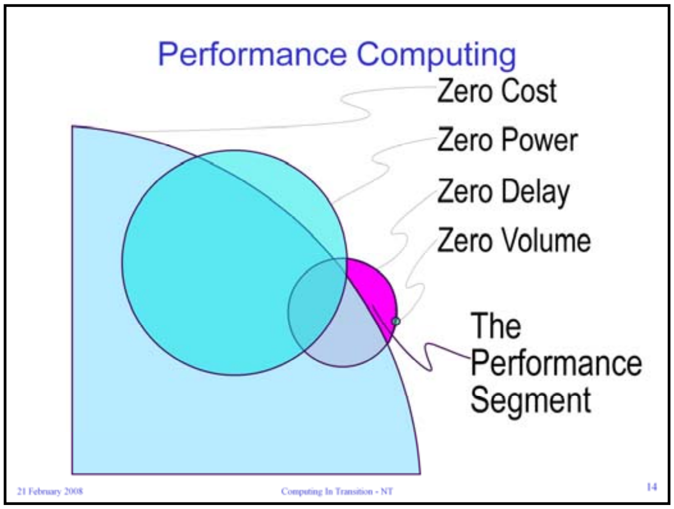
Consumer Mobile Devices

21 February 2008 Computing In Transition - NT 13

As the PC became a commodity, logic designers began the transition from low-margin, cost-performance systems to higher margins in mobile systems with a design goal of cost-performance per watt. This segment is represented by the overlap of zero cost, zero delay, and zero power in this diagram.

One problem with reaching the design objectives for this segment is that instruction-based processing simply isn't efficient enough.

Another problem is that leakage currents have become significant. When custom chips were a few thousand transistors, active currents were in milliamps and leakage currents were in nanoamps and could be safely ignored. Active currents depend on the small percent of transistors that are switching at any time; all transistors contribute to leakage currents. When active and leakage currents were six orders of magnitude apart, leakage current could be ignored. As chips grew in complexity from a few thousand transistors to a few billion, leakage currents grew from nanoamps to the same order as active current.



Performance Computing

Zero Cost

Zero Power

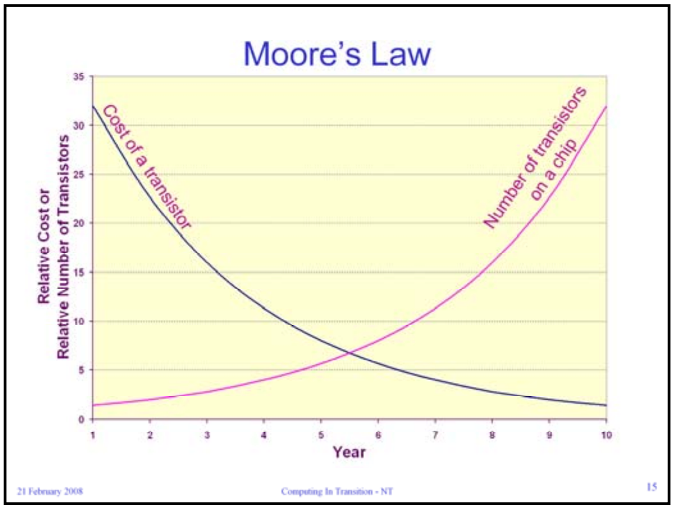
Zero Delay

Zero Volume

The Performance Segment

21 February 2008 Computing In Transition - NT 14

Just a short example here of a use for this model. I used this slide at a recent supercomputing conference. The pink wedge represents the performance segment occupied by supercomputers; supercomputers are in the part of the zero-delay segment that lies outside the consumer (zero-cost) segment. That's not good news for supercomputer enthusiasts. It says that they had better learn to build their systems from components that come from developers working inside the zero-cost segment because that's the only segment with enough volume to support the high cost of custom chip design.



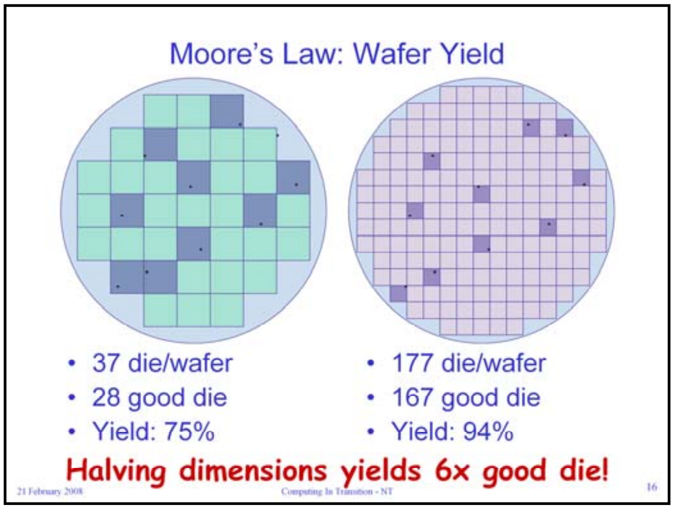
Moore’s Law

21 February 2008 Computing In Transition - NT 15

At the beginning of the presentation, I showed a slide that illustrated the growth of the worldwide semiconductor market and the growth of the gross world product. It showed that the semiconductor market’s growth rate is twice that of the world’s GWP. Moore’s law offers one explanation for why that’s so. This graph illustrates the dual nature of the benefits offered by Moore’s law improvements in semiconductor process.

Each generation of Moore’s law expands applications at the high end and at the low end. With each generation, more transistors fit on a chip, so applications that were out of reach for complexity or for performance are now tractable. With each generation, the cost of an individual transistor falls, so that low-cost applications that were out of reach for cost are now affordable. If the function you wanted for your design of the next-generation electric razor is too expensive this year, it’ll be half as much in two years and it’ll be time to implement new features in the razor.

Think of a graph with performance on the horizontal axis and cost on the vertical axis. The range of applications is an oval. With each successive chip generation, the oval expands in all directions. At the low end, cost decreases to reach new applications; at the high end, performance increases to reach new applications.



Moore’s Law: Wafer Yield

- 37 die/wafer
- 28 good die
- Yield: 75%

- 177 die/wafer
- 167 good die
- Yield: 94%

**Halving dimensions yields 6x good die!**

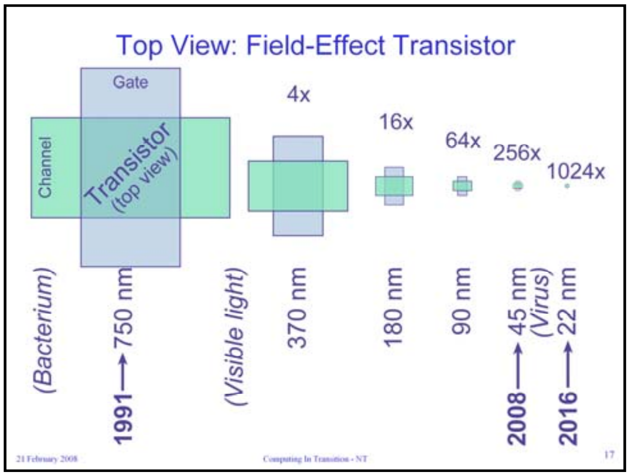
21 February 2008 Computing In Transition - NT 16

Since I’m sure you are not only familiar with Moore’s law, but probably sick of it by now, I’ll go through these slides quickly. I’m showing them because they give a quick visual display of why shrinking transistors works so well and they illustrate a little about just how difficult it is.

This figure illustrates that the advantage of shrinking the line width by half is much more than the expected value of four.

Specialization is one of the less-appreciated aspects of economic advance and this illustrates it well. There isn’t a one of us that could so much as manufacture a pencil, yet you are on your way to mastering a skill that will pay you an excellent living by mastering the minute details of a technical specialty.

A lot of detail is missing in this simple model. For example, shrinking the transistors means that smaller contaminants can destroy chips, so the clean room must get cleaner.



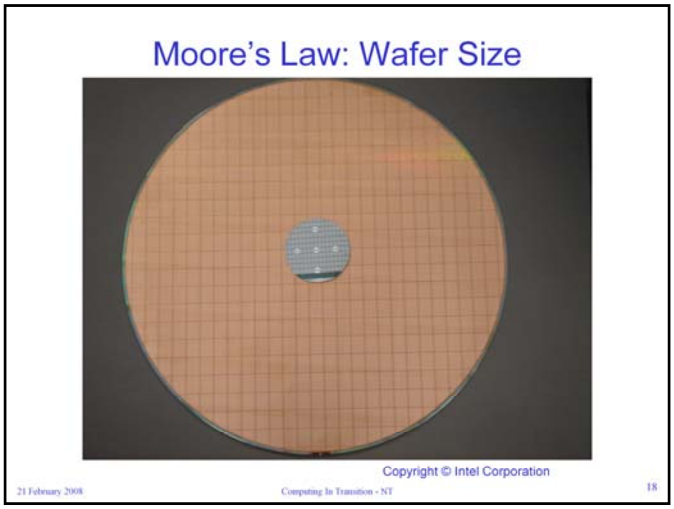
Top View: Field-Effect Transistor

21 February 2008 Computing In Transition - NT 17

This figure is more background for conclusions I’ll draw later. It illustrates where we’ve been in semiconductor processing and where we’re headed. I wanted to remind you of the dimensions semiconductor companies work with. Looking at the 1991 transistor on the left, it is made up of a “channel” and a “gate.” Current flows through the channel. The gate, which sits on top of the channel, controls the current flow. The width, or “line width,” of the channel is 750 nm, the line width of the gate is also 750 nm.

Because making transistors is essentially a two-dimensional process, cutting the line width in half enables four times as many transistors in the same area. Four of the 370-nm transistors fit in the area occupied by the 750-nm transistor. By 1991, transistors were already smaller than bacteria. By the 370-nm generation, transistors were smaller than the wavelength of ordinary light. You cannot see these transistors with an optical microscope. This year, companies will build chips with 45-nm transistors. That’s about the size of a *virus*. At 45 nm, almost half a million transistors fit on a small grain of sand! That’s why big chips can have a few billion transistors.

I don’t want to dwell on this too long, but think about nanotechnology and bioengineering for a moment. These fields didn’t come from nowhere; their emergence was enabled by the march of semiconductor process advances. The tools and techniques developed for integrated circuit manufacturing marched right down to the molecular resolutions that enable advances in bioengineering and nanotechnology. The semiconductor industry will soon be co-opting nature’s solutions.

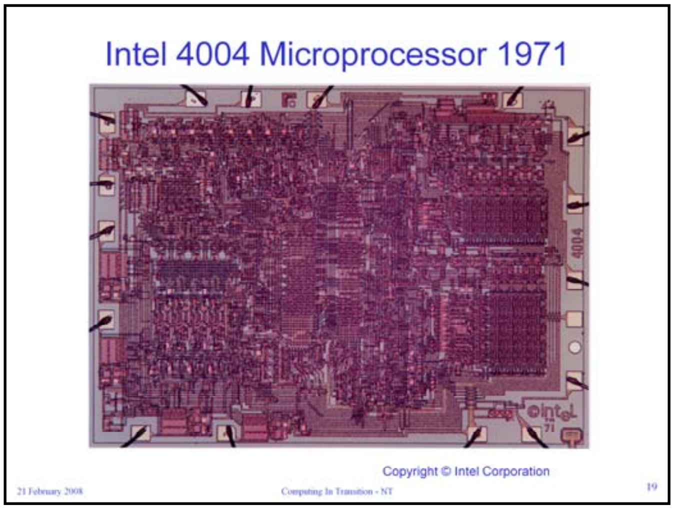


Moore’s Law: Wafer Size

Copyright © Intel Corporation

21 February 2008 Computing In Transition - NT 18

A 2” 4004 wafer on top of a 12” Pentium wafer. We tend to forget that part of the progress in semiconductor processing involves making larger wafers.

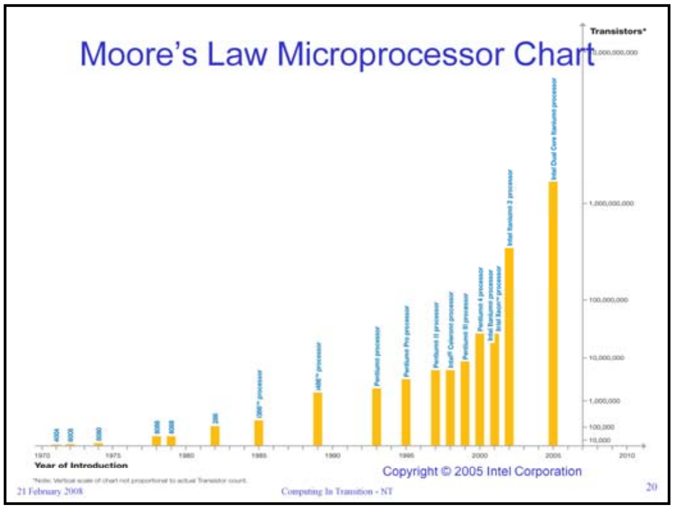


## Intel 4004 Microprocessor 1971

Copyright © Intel Corporation

21 February 2008 Computing In Transition - NT 19

In a photo of the Intel 4004, it’s possible to see the connections, buses, and even the individual transistors.

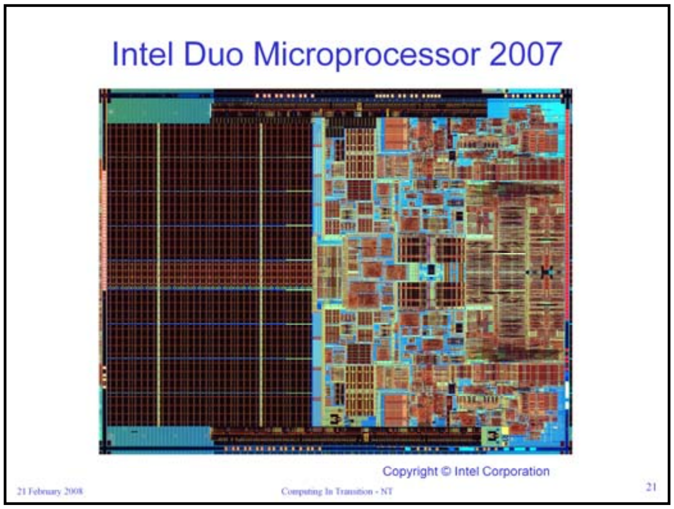

Moore’s Law Microprocessor Chart

Copyright © 2005 Intel Corporation

*Note: Vertical scale of chart not proportional to actual Transistor count.

21 February 2008 Computing In Transition - NT 20

This chart, borrowed from Intel, shows the march of microprocessor complexity. Note that the vertical scale is logarithmic. There’s exponential advance in the complexity of microprocessors even on a logarithmic scale! On a linear scale every design worked on before 2002 would look as if it had zero transistors.



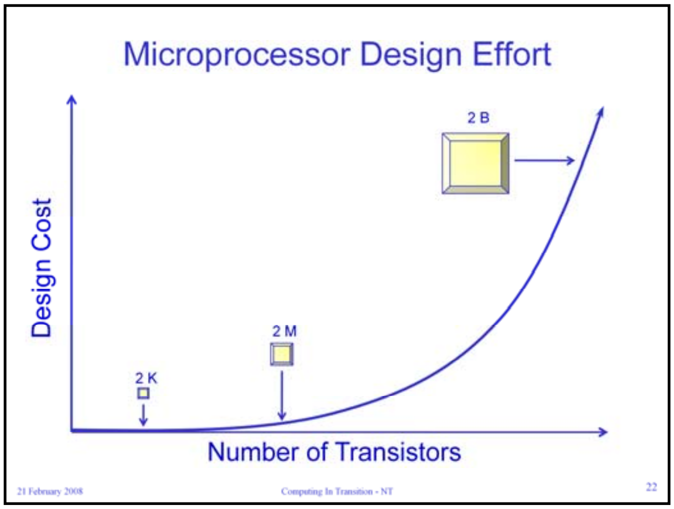
Intel Duo Microprocessor 2007

Copyright © Intel Corporation

21 February 2008 Computing In Transition - NT 21

The Intel Duo microprocessor is more than a billion transistors. That compares with the 4004, which had, I think, 2300 transistors. The 4004 was an autonomously functional state sequencer that could run programs. It was a microprocessor. And here’s the Duo, a single chip, that’s got enough transistors to build more than 400,000 primitive microprocessors.

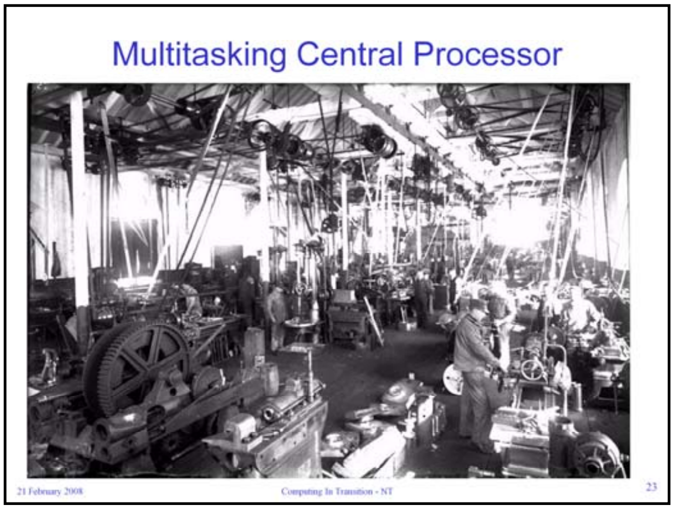
Unlike the transistors on the 4004, that can be seen in an ordinary photograph, the individual transistors on this chip are less than a quarter of the wavelength of visible light. Not only is it not possible to resolve individual transistors in a photograph, they cannot even be resolved with a high-power optical microscope.



Microprocessor Design Effort

21 February 2008 Computing In Transition - NT 22

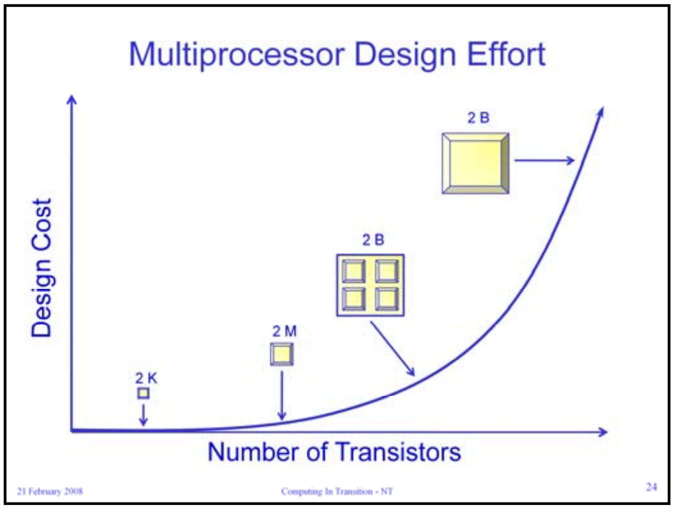
One problem with the rising complexity of designs is that design cost is rising at least as fast as complexity. This figure would also have a logarithmic vertical axis. 2K is about the level of the Intel 4004 in 1971, 2M is close to the Intel Pentium in 1993, and 2B is today’s design point.



Multitasking Central Processor

21 February 2008 Computing In Transition - NT 23

Here’s an old machine-shop photo that illustrates the concept behind central processing. This is how machine shops were organized in the days of the central processor—that is, a single power source (steam turbine, waterwheel, giant electric motor, or whatever) in the days before the invention of the fractional-horsepower motor. The introduction of the fractional-horsepower motor revolutionized not only the organization and operation of machine shops, but manufacturing businesses around the world. It enabled huge advances in productivity. Central control is the situation with giant multitasking central processors today. So they’re trying to do everything from a single processing core and they are climbing the escalating cost curve on design. It’s not a situation that can continue.



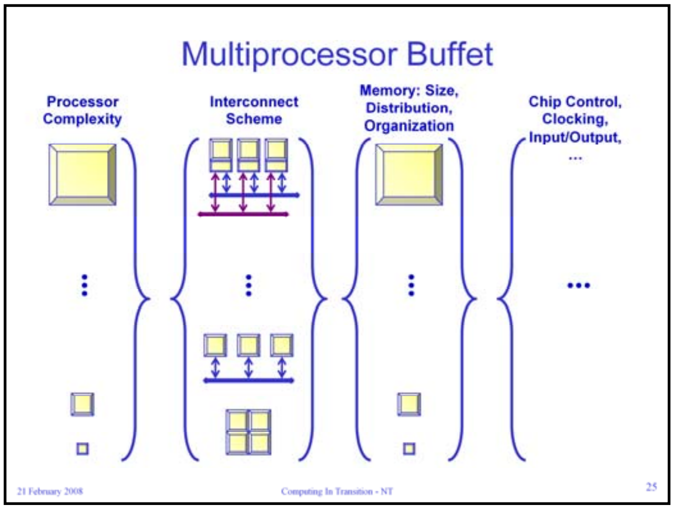
Multiprocessor Design Effort

21 February 2008 Computing In Transition - NT 24

IBM’s z6 Mainframe processor is 991M transistors. Hot Chips 2007

The accompanying SMP Hub Chip is 1.6B transistors, w/ 242Mb SRAM.

It’s obvious that single-processor designs are just too expensive, so many companies have noticed that design cost is much lower for multiprocessors. They incur approximately the design cost of the processor core’s transistors, but have aggregate compute capacity according to the number of cores on the chip. The problem with this is in extracting the latent performance from the design.



Multiprocessor Buffet

Processor Complexity

Interconnect Scheme

Memory: Size, Distribution, Organization

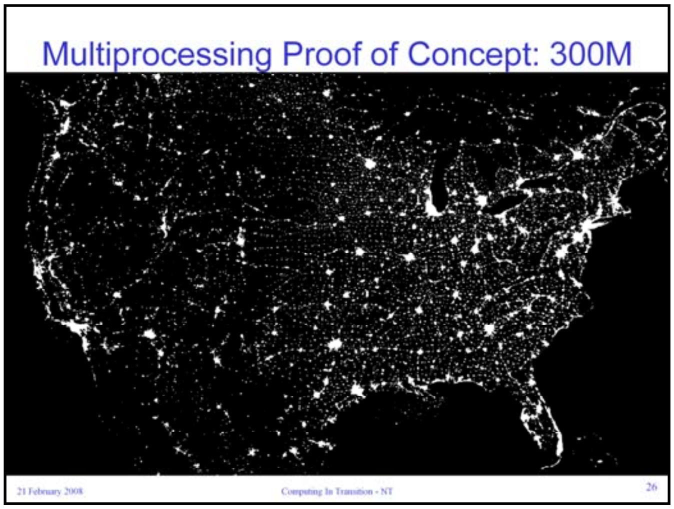
Chip Control, Clocking, Input/Output, …

21 February 2008 Computing In Transition - NT 25

The industry has not yet figured out what works. Today’s designs are a smorgasbord of choices: one from column A, one from column B, and so on. They vary in processor complexity, in interconnect scheme, in memory organization, in memory distribution, in clocking, and in control. Should the processors be homogeneous or heterogeneous? Should the memory be per-processor, shared, or a mix? Should there be two processors or two thousand? Should there be separate control and data buses? Should busing be hierarchical or flat? Dozens of variables just in the design; never mind what it will take to program these things.

We don’t yet know what will work. I’ve seen impressive demonstrations, but it’s usually one labor-intensive effort to show one optimally suited application.

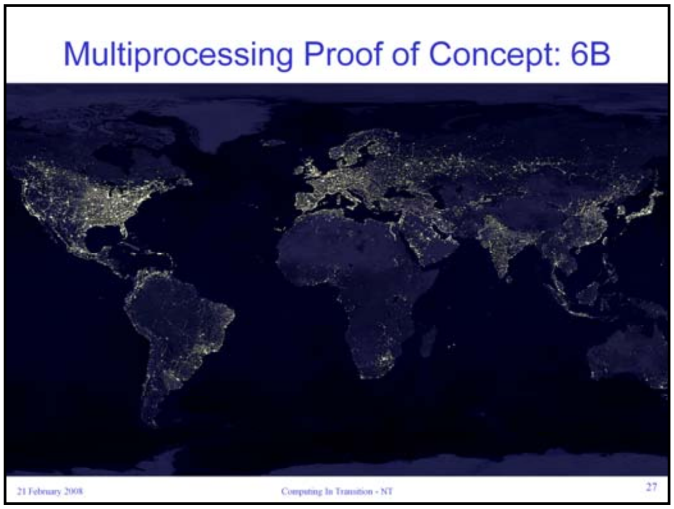
Perhaps it will turn out that numerous application areas will have particular organizations. I don’t know, but I do know that multiprocessors work.



## Multiprocessing Proof of Concept: 300M

21 February 2008 Computing In Transition - NT 26

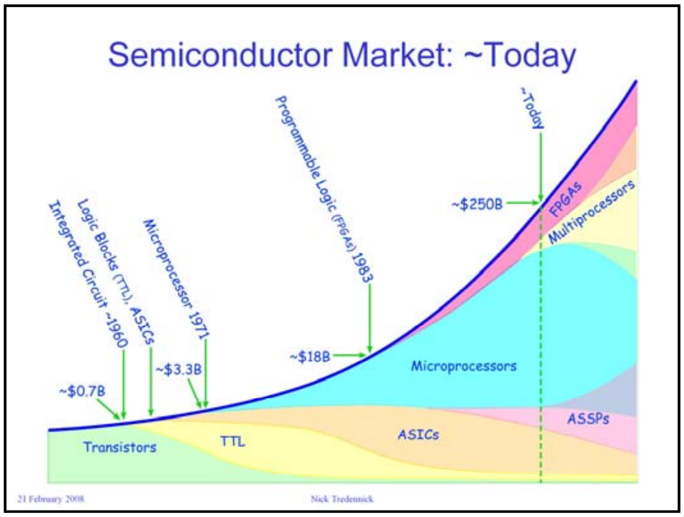
Here’s a proof-of-concept photograph of a 300-million-unit loosely coupled multiprocessor. This particular multiprocessor, with approximately 5% of the world’s population and 6.2% of the world’s land area, generates 28% of the gross world product.



Multiprocessing Proof of Concept: 6B

21 February 2008 Computing In Transition - NT 27

And here’s proof that it scales.



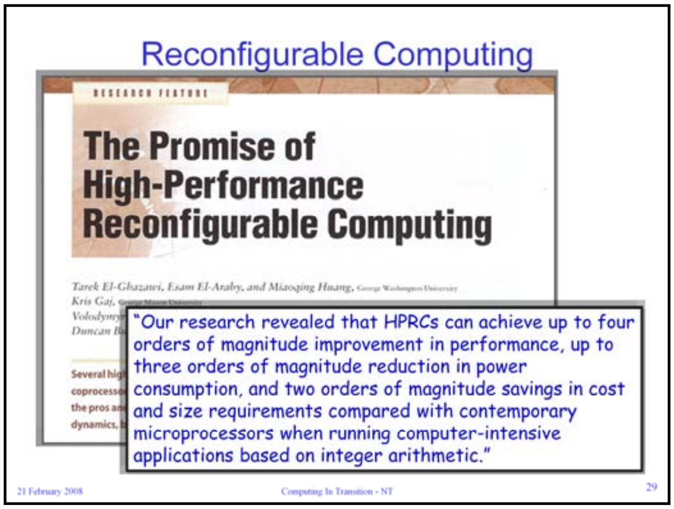
Semiconductor Market: ~Today

21 February 2008

Nick Tredennick

Finally, we arrive back at this culminating slide. Looking forward from today from both a career and dollar investment perspective, the question is: how will the future unfold? What will happen at the boundary between multiprocessors and FPGAs? Will the FPGAs gain market share? Will multiprocessors gain market share from microprocessors? Will ASSPs encroach on the territory now owned by ASICs? Will ASSPs encroach on territory held today by microprocessors? Will some entirely new technology or possibly some hybrid of the others begin a new wedge?

I don’t have the answers; it’s your career, your decision, and your money. But I’ll give you a hint or two.



Reconfigurable Computing

RESEARCH FEATURE

The Promise of High-Performance Reconfigurable Computing

Tarek El-Ghazawi, Esam El-Araby, and Miaoqing Huang, George Washington University
Kris Gaj, George Mason University
Volodym
Duncan B

Several hig
coprocesso
the pros an
dynamics, b

"Our research revealed that HPRCs can achieve up to four orders of magnitude improvement in performance, up to three orders of magnitude reduction in power consumption, and two orders of magnitude savings in cost and size requirements compared with contemporary microprocessors when running computer-intensive applications based on integer arithmetic."

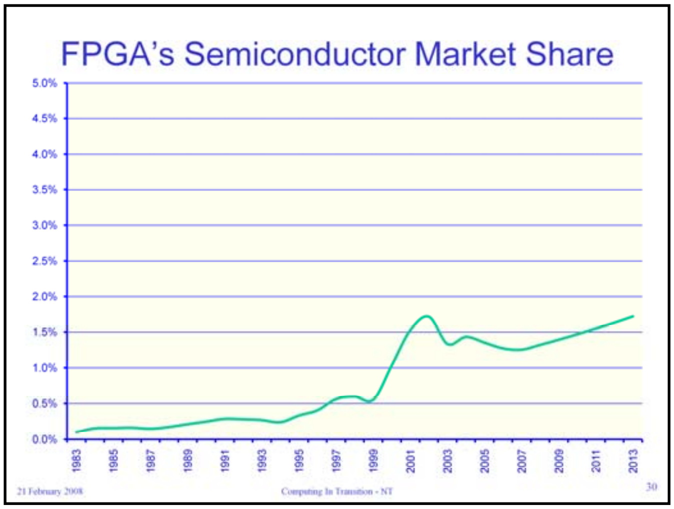
21 February 2008 Computing In Transition - NT 29

Here's the headline on the first page of an article in this month's IEEE *Computer* magazine. The article surveys large computing applications in remote sensing, molecular dynamics, bioinformatics, and cryptanalysis using microprocessor-based computers with FPGA accelerators. The microprocessors run gigahertz clocks, while the FPGAs run at 100-200 MHz, yet some applications are accelerated by factors of hundreds or thousands. That's better efficiency than instruction-based processing can achieve. These systems, as the paper shows, are still in their infancy, so they display application-processing bottlenecks that significantly reduce the hardware's efficiency. These systems will evolve jsuts as the microprocessor has and will achieve even better results in the near future.

Reconfigurable systems are going through the same investigative experimentation that today besets multiprocessor systems. The two main categories of reconfigurable systems are uniform-node non-uniform systems (UNNS) and non-uniform-node uniform systems (NNUS). UNNS have a variable number of FPGAs per microprocessor, while the NNUS fix the ratio of FPGAs per microprocessor.

Here's one of the concluding paragraphs from the article:

> "Our research revealed that HPRCs can achieve up to four orders of magnitude improvement in performance, up to three orders of magnitude reduction in power consumption, and two orders of magnitude savings in cost and size requirements compared with contemporary microprocessors when running computer-intensive applications based on integer arithmetic."

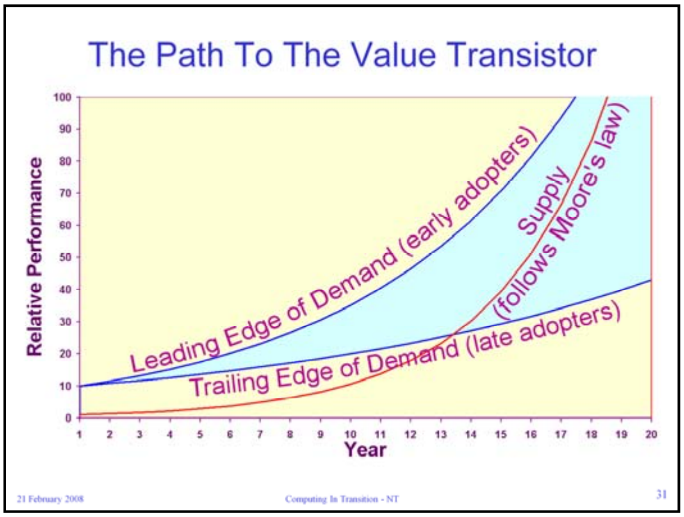


As I said earlier, fifteen years ago I was convinced that the FPGA market would grow faster than the rest of the semiconductor market. Here's what happened. This should be a hint to you not to trust any other predictions I might make. In any event, what this chart is showing is the percent of the worldwide semiconductor market held by the FPGA companies over time. So this is the first curve I presented (the world semiconductor market in dollars, by year) divided by some number representing the revenues of the FPGA companies. It shows that they have grown at approximately the rate of the semiconductor market's growth.

Why have they not done better? I think it's because they are component companies dependent upon the number of logic designers. They are capacity limited by the available expertise in the design community. I have been badgering them for years to change their customer focus from logic designers (the limiting resource) to programmers, because programmers outnumber logic designers by at least ten to one. (Some say this is twenty to one or more.) But they are slow to do so because the companies were begun and run by circuit designers who took twenty years to make the transition to management with logic design expertise. Programming expertise and an understanding of the market is still on the distant horizon.

A second factor holding back the development of the FPGA market is the lack of a good non-volatile memory cell.

Why am I even showing FPGAs, given that they are such a small percent of the market (again the figure is not to scale if there's even enough wedge to accommodate a label)? FPGAs are there for their potential and for reconfigurable computing (a topic not covered in this talk).


The Path To The Value Transistor

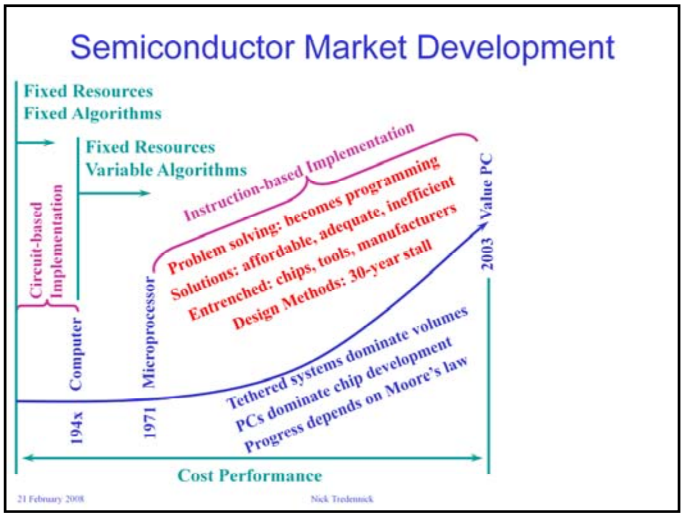
21 February 2008 Computing In Transition - NT 31

I’m using the transistor to illustrate supply and demand. The concept of supply and demand is underappreciated in the semiconductor business because the industry has been growing at the leading edge of what Moore’s law could supply that we have come to take it for granted that supply and demand are synonymous for our industry. It isn’t so.

When the transistor came out, it wasn’t good enough for any of its applications. The transistor improved at some rate. The demand for transistor performance grew also, but at a different rate. There’s no necessary correlation between the rate of improvement in supply and the rate of increase in demand for what is being supplied.

These phony curves illustrate the point. The transistor began with performance well below demand. Over time, the transistor improved. Demand for performance rose, but it also spread—and it increased at its own rate that was independent of the rate of improvement in transistors. After a few years, the transistor had improved enough and demand had spread enough that there were some transistors that were good enough for some applications.

The same arguments apply to integrated circuits. There’s a huge difference in the performance demanded by leading-edge digital transceivers and by consumer washing machines, for example. Washing machines don’t need leading-edge ASICs.



Semiconductor Market Development

Fixed Resources
Fixed Algorithms

Fixed Resources
Variable Algorithms

Instruction-based Implementation

Problem solving: becomes programming
Solutions: affordable, adequate, inefficient
Entrenched: chips, tools, manufacturers
Design Methods: 30-year stall

Tethered systems dominate volumes
PCs dominate chip development
Progress depends on Moore's law

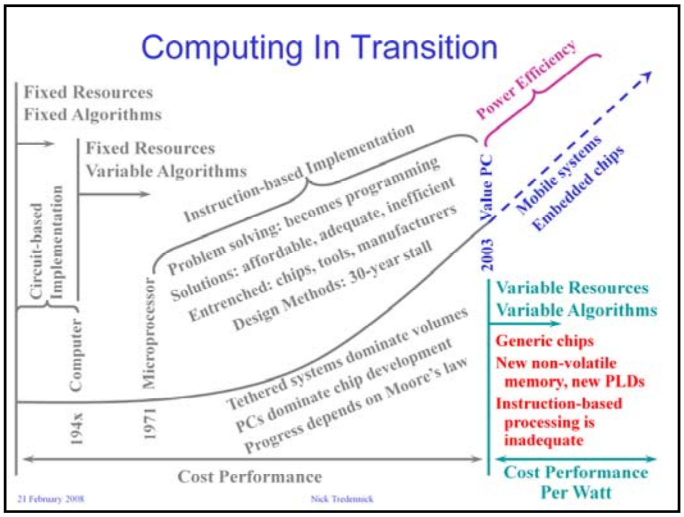
I have two summary charts on where the market has been and where I think it’s headed. In the abstract I said that logic design innovation had stalled for thirty years and, so far in the talk there hasn’t been a word about that seemingly controversial statement.

Before the invention of the computer, all hardware was custom. The resources were fixed and the algorithms were fixed.

With the introduction of the computer, programming enabled fixed hardware to support a range of algorithms: fixed resources and variable algorithms.

The introduction of the microprocessor accelerated the trend to substitute programming for custom hardware design. Problem solving became programming. Solutions were affordable, adequate, and inefficient. That was all that was necessary because the design goal for systems that drew power from wall sockets was cost-performance.

The personal computer was the first “killer app” because its unit volumes were so large that, at one time, it encompassed 40% of the dollar value of the worldwide semiconductor market. But as the PC’s performance passed the minimum requirements of the majority of users, the PC became a commodity, shrinking margins. The PC that meets the needs of most users is the “value PC.” The emergence of the value PC sent electronics firms in search of higher-margin opportunities.



Computing In Transition

Fixed Resources
Fixed Algorithms

Fixed Resources
Variable Algorithms

Circuit-based Implementation

Computer

Microprocessor

194x

1971

Instruction-based Implementation

Problem solving: becomes programming
Solutions: affordable, adequate, inefficient
Entrenched: chips, tools, manufacturers
Design Methods: 30-year stall

Tethered systems dominate volumes
PCs dominate chip development
Progress depends on Moore's law

Cost Performance

2003

Value PC

Power Efficiency

Mobile systems
Embedded chips

Variable Resources
Variable Algorithms

Generic chips
New non-volatile memory, new PLDs
Instruction-based processing is inadequate

Cost Performance Per Watt

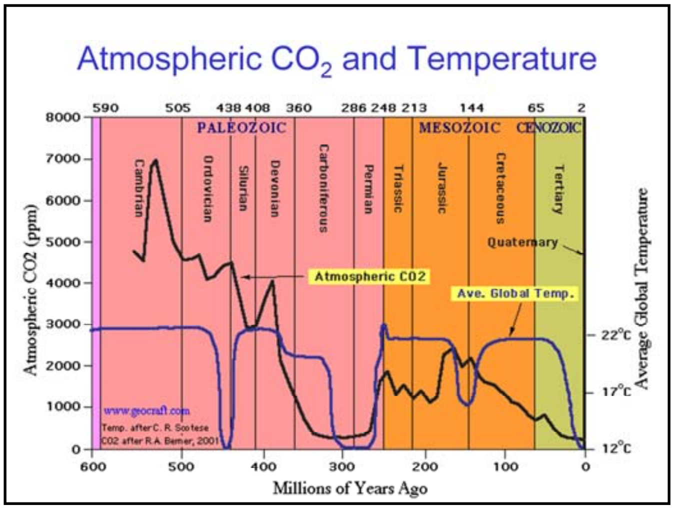
21 February 2008

Nick Tredennick

I've arbitrarily placed the commoditization of the PC (the "value PC") at 2003. When the PC became a commodity, that started the transition in design from cost performance to cost-performance per watt. For that transition, which will take years because designers have to change their design methods, instruction-based processing simply isn't efficient enough. One way to improve efficiency is to page custom logic implementations into the hardware in the way that software is paged into microprocessors today. The processor in your mobile device would have a different logic implementation for a phone call than it has as it plays a movie.

## Questions?

21 February 2008 Computing In Transition - NT 34



Here’s the 600-million-year history of the earth. The only time the earth’s CO2 levels have been as low as they are today was during the Carboniferous period.

For a short paper on this topic, please see “Carbon Crazies” at:

http://www.tredennick.com/ftp/pub/documents/Publications

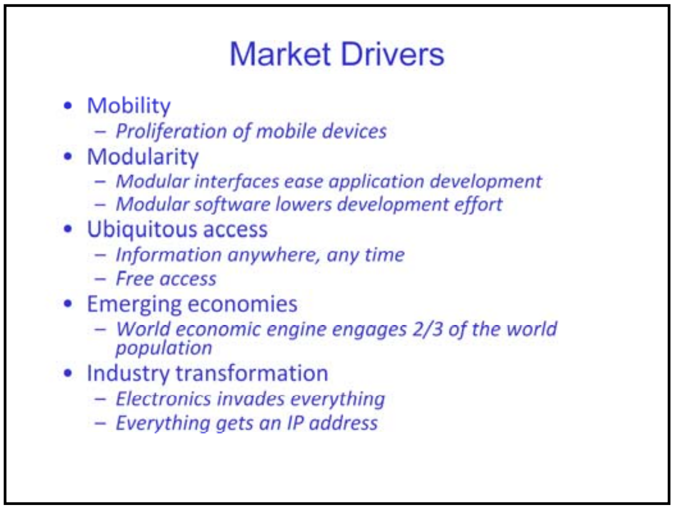
Here are some conclusions I draw from this figure:

1.] There is no correlation between levels of atmospheric $CO_2$ and average global temperature.

2.] There’s no cause and effect relationship between levels of atmospheric $CO_2$ and average global temperature.

3.] It’s not reasonable to make millions of measurements at the right-hand axis (all the ice-core samples ever taken are in the right-hand quarter of the vertical axis (Quaternary period) on the right) and to attempt, with that data, to infer a relationship between levels of atmospheric $CO_2$ and average global temperature.

4.] Human activity is irrelevant in its influence on levels of atmospheric $CO_2$ and on average global temperature. (This is because the absolute level of atmospheric $CO_2$, at 380 ppm, is historically low and because the human contribution to the increase of this already-low number is insignificant. $CO_2$ is 0.04% of the atmosphere; the human contribution is ~3% of that.)



### Market Drivers

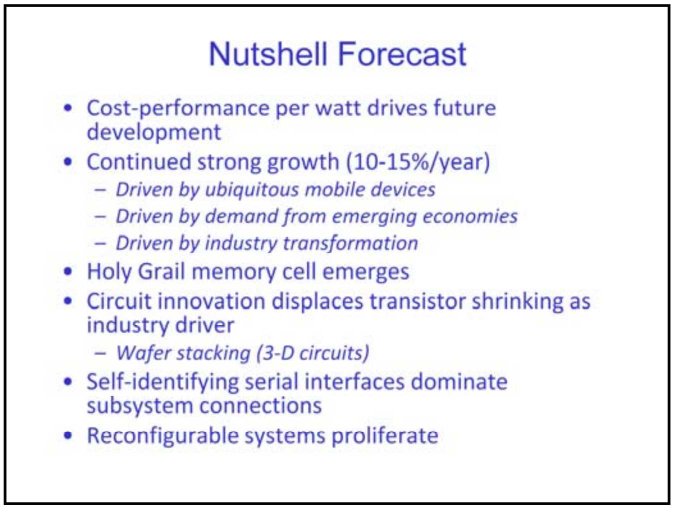
- Mobility
  - *Proliferation of mobile devices*
- Modularity
  - *Modular interfaces ease application development*
  - *Modular software lowers development effort*
- Ubiquitous access
  - *Information anywhere, any time*
  - *Free access*
- Emerging economies
  - *World economic engine engages 2/3 of the world population*
- Industry transformation
  - *Electronics invades everything*
  - *Everything gets an IP address*

**Trend one: device mobility**. My brother, John, builds houses. In the old days, he had three choices at a new job site. He could wait for the local utility's line power installation, he could bring a generator, or he could work with unpowered tools. Today, he uses battery-powered hand tools that are powerful and efficient and are good enough to last all day. He'd never go back to tripping over tangled power cords. The convenience of portable devices that is compelling for construction work applies across a wide range of occupations and leisure activities. The convenience of mobile handhelds extends to toothbrushes, hair dryers, telephones, laptop computers, PDAs, and remote controls. The consumer market for mobile devices is growing rapidly.

**Trend two: Modularity**. Modular interfaces ease application development. Sophisticated electronic systems can be assembled with the latest components, spreading sensor and processor development cost across many applications. This makes many previously unaffordable applications cost effective.

**Trend three: ubiquitous access**. Almost an extension of device mobility is the demand for ubiquitous access: voice, video, and data. Not so long ago, I visited a library for information, made calls when I got to a convenient land-line telephone, and played music or watched movies at home. No more. Today, I get impatient and frustrated when I can't answer a question by immediate access to information on the Internet, when I can't make or take a call anywhere, and, probably soon, when I can't see or hear whatever I want at any time or place. I expect any information I want, whether it's historical facts about Civil War battlefields or specific product and pricing information about items in a nearby store, to be immediately available—and I expect it to be free. Such access will be so common and so pervasive that everyone will take it for granted.

**Trend four: emerging economies**. Humans, as Julian Simon demonstrates compellingly in his writing, are the ultimate resource; they are the cylinders of the world's wealth-creation engine. The world economic engine has been running on less than a third of its cylinders. With China, India, Eastern Europe, and others joining the world economy, that engine will soon engage two thirds of the world's population. We are entering a period of unprecedented wealth creation. Demand is the flip-side of wealth creation; as more individuals produce more, demand rises as newly created wealth is spent. The world's producers will scramble to meet the demands of emerging economies for household appliances and for other goods that the developed world takes for granted.



## Nutshell Forecast

- Cost-performance per watt drives future development
- Continued strong growth (10-15%/year)
  - *Driven by ubiquitous mobile devices*
  - *Driven by demand from emerging economies*
  - *Driven by industry transformation*
- Holy Grail memory cell emerges
- Circuit innovation displaces transistor shrinking as industry driver
  - *Wafer stacking (3-D circuits)*
- Self-identifying serial interfaces dominate subsystem connections
- Reconfigurable systems proliferate

Cost-performance per watt and time to market will dominate design for the consumer systems that make up the bulk of semiconductor demand.

The semiconductor market will continue to grow in the range of ten to fifteen percent per year as electronics invades everything from carpeting to car bumpers. Demand from emerging economies will spur growth rates of the traditional semiconductor market.

At least one Holy-Grail memory cell, with the speed of SRAM, the density of DRAM, and the non-volatility of flash memory, will emerge and gain commercial importance within five years.

Wafer stacking and other techniques will displace transistor-shrinking as the means to continue historical performance improvements and price declines in semiconductors.

Self-identifying serial interfaces will become the way to connect disparate subsystems in single-chip designs, simplifying integration of functional blocks.

Programmable-logic derivatives will become pervasive in the form of systems that house physical circuits in the way that today’s memory devices house programs. These programmable-logic chips will be generic at manufacture and will be customized in the field.