The Lives and Death of Moore's Law

Article *in* First Monday · April 2002

DOI: 10.5210/fm.v7i11.1000 · Source: DBLP

Ilkka Tuomi
Meaning Processing

The Lives and the Death of Moore's Law

Ilkka Tuomi

Visiting Scientist
Joint Research Centre
Institute for Prospective Technological Studies
ilkka.tuomi@jrc.es

1 October 2002

Moore's Law has been an important benchmark for developments in microelectronics and information processing for over three decades. During this time, its applications and interpretations have proliferated and expanded, often far beyond the validity of the original assumptions made by Moore. Technical considerations of optimal chip manufacturing costs have been expanded to processor performance, economics of computing, and social development. It is therefore useful to review the various interpretations of Moore's Law and empirical evidence that could support them.

Such an analysis reveals that semiconductor technology has evolved during the last four decades under very special economic conditions. In particular, the rapid development of microelectronics implies that economic and social demand has played a limited role in this industry. Contrary to popular claims, it appears that the common versions of Moore's Law have not been valid during the last decades. As semiconductors are becoming important in economy and society, Moore's Law is now becoming an increasingly misleading predictor of future developments.

The Lives and the Death of Moore's Law

Moore's Law has been an important benchmark for developments in microelectronics and information processing for over three decades. During this time, its applications and interpretations have proliferated and expanded, often far beyond the validity of the original assumptions made by Moore. Technical considerations of optimal chip manufacturing costs have been expanded to processor performance, economics of computing, and social development. It is therefore useful to review the various interpretations of Moore's Law and empirical evidence that could support them.

Such an analysis reveals that semiconductor technology has evolved during the last four decades under very special economic conditions. In particular, the rapid development of microelectronics implies that economic and social demand has played a limited role in this industry. Contrary to popular claims, it appears that the common versions of Moore's Law have not been valid during the last decades. As semiconductors are becoming important in economy and society, Moore's Law is now becoming an increasingly misleading predictor of future developments.

1. Introduction

In 1965, Gordon Moore, Director of Fairchild Semiconductor's Research and Development Laboratories, wrote an article on the future development of semiconductor industry for the 35th anniversary issue of the Electronics magazine. In the article, Moore noted that the complexity of minimum cost semiconductor components had doubled per year since the first prototype microchip was produced in 1959. This exponential increase in the number of components on a chip became later known as Moore's Law. In the 1980s, Moore's Law started to be described as the doubling of number of transistors on a chip every 18 months. At the beginning of the 1990s, Moore's Law became commonly interpreted as the doubling of microprocessor power every 18 months. In the 1990s, Moore's Law became widely associated with the claim that computing power at fixed cost is doubling every 18 months.

Moore's Law has mainly been used to highlight the rapid change in information processing technologies. The growth in chip complexity and fast reduction in manufacturing costs have meant that technological advances have become important factors in economic, organizational, and social change. In fact, during the last decades a good first approximation for long-range planning has often been that information processing capacity is essentially free and technical possibilities are unlimited.

Regular doubling means exponential growth. Exponential growth, however, also means that the fundamental physical limits of microelectronics are approaching rapidly. Several observers have therefore speculated about the possibility of "the end of Moore's Law." Often these speculations have concluded by noting that Moore's Law will probably be valid for at least "a few more generations of technology," or about a decade. An important example is the International Technology Roadmap for Semiconductors (ITRS), which now extends to 2016. This roadmap is generated by a global group of experts and represents their consensus. Although it notes that within the next 10-15 years "most of the known technological capabilities will approach or have reached their limits," its basic assumption is that Moore's Law, although perhaps slowing down, still provides a good basis for predicting future developments in the semiconductor industry (ITRS, 2001).[1]

Of course, if Moore's Law is valid, independent of the exact nature of physical limits, exponential development means that the limits are only a few technological generations ahead. Order of magnitude errors in our current estimates of the ultimate limits of chip technology will create at most a few months of error in the time when they become bottlenecks in chip technology.[2] As a result, it is easy to predict that Moore's Law will become invalid soon. Speculations on the extended lifetime of Moore's Law are therefore often centered on quantum computing, bio-computing, DNA computers, and other theoretically possible information processing mechanisms. Such extensions, obviously, extend beyond semiconductor industry and the domain of Moore's Law. Indeed, it could be difficult to define a "component" or a "chip" in those future devices.

Although discussion on physical limits, bottlenecks, and alternative information processing models may be important for chip manufacturers, it is, however, not the focus of this paper. Instead, I shall argue that there is no end to Moore's Law in sight simply because it never accurately described developments in microelectronics. It never was valid. Furthermore, it neglected factors that are becoming increasingly visible and important.

The present paper argues that Moore's Law has not been a driver in the development of microelectronics or information technology. This may come as a surprise to some who have learned that Moore's Law has become a self-fulfilling prophecy that semiconductor industry firms have to follow if they want to survive. Instead, I shall argue that technical development in semiconductors during the last four decades has reflected the unique economic and social conditions under which the semiconductor industry has operated. This is important as these conditions are changing. In short, the observed technological trends and their analysis indicate that the semiconductor industry has been a core element in an industry cluster where development to an important extent has been driven by intra-cluster forces. The semiconductor industry, in other words, has been a laboratory of endogenous growth.

This laboratory of endogenous growth has also made many failed experiments. Technology has not evolved in the ways predicted by Moore. Moore's Law has become popular partly because it has allowed great flexibility in interpretation and selective choice of supporting data. In this process, Moore's Law and evidence for it have retrospectively been interpreted to establish the validity of the law. Often the presented evidence has been in visible contradiction with the law.

Technological advances in silicon chips have been relatively independent of end-user needs. During the last three decades, a good approximation in this industry has been that the supply of technology determines development. In economic terms, a key

---

[1] ITRS uses many different variations of Moore's Law in different parts of its documentation. These will be discussed in more detail below.

[2] The continuation of geometrical scaling would mean that by 2020 there would be less than one electron available for switching each transistor on microchips (Birnbaum & Williams, 2000: 39).

factor underlying the rapid development of semiconductor technology has been a continuous imbalance between supply and demand.

It is, however, also clear that no one has exactly been breaking Moore's Law. Strictly speaking there is no such a law. Most discussions that quote Moore's Law are historically inaccurate and extend its scope far beyond available empirical evidence. Indeed, sociologically Moore's Law is a fascinating case of how myths are manufactured in the modern society and how such myths rapidly propagate into scientific articles, speeches of leading industrialists, and government policy reports around the world.

It is therefore useful to revisit the evolution of Moore's Law. During its lifetime, its interpretations have involved mainly technical and economic considerations. The following discussion will therefore combine historical, technical, and economic concepts and data.

The paper is organized as follows. In the next section, I revisit the original formulation of the Moore's Law and describe its basic assumptions. Section 3 discusses the revisions that Moore made to his original formulation during the 1970s. Section 4 describes the extensions of Moore's Law that became dominant in the 1980s and 1990s. Section 5 then evaluates available evidence to see whether any of the proposed formulations of Moore's Law can be justified. Its subsections review evidence on component counts, microprocessor performance, increase in computing power, and quality-adjusted cost of computing. Section 6 then briefly discusses industrial dynamics that underlie technical development in semiconductors and information processing, and points out some reasons why Moore's Law is becoming increasingly misleading in forecasting future developments in information processing technology. The paper concludes with some general observations on the relation between technical, economic, and social development.

2. Moore's original formulation

Accounts of technological development are retrospective stories about continuous advance. In such accounts, historical events often become reinterpreted. This commonly involves modification of the facts and simplification of the "story line" so that specific individuals become represented as main actors in the process of development. As Robert Merton noted, historical acclaim and reputation tend to be allocated to people unevenly. Scientific observations and results are often associated with people who have high visibility and social status. Merton (1973:439-59; 1988; 1995) formulated this as the Matthew Effect: "To those who have will be given more."

Eponymy is a striking example of this phenomenon. Particularly important scientific observations are often associated with a person, as in the case of Gaussian distribution, Halley's comet, and Planck's constant. Historians of science, however, have noted that often the person who is associated with the particular observation, theory, or result was not its original inventor. Based on his studies on the history of statistics, Stephen Stigler (1999: 277) therefore proposed his own "Stigler's Law of

Eponymy." In its simplest and strongest form it says: "No scientific discovery is named after its original discoverer."

When Moore in 1965 observed that the number of components on integrated circuits seemed to double regularly, this fact was known to many people working in the area.[3] Indeed, it took over a decade, during which Moore had become the co-founder of Intel and its President and CEO, before this observation was called Moore's Law. During this decade, the content of the original Moore's Law, however, also started to shift.

In his 1965 article, "Cramming more components onto integrated circuits," Moore discussed the future of integrated electronics. He noted that the object of miniaturization had been in putting increasingly complex electronic functions in limited space with minimum weight. Indeed, the first chip, created in 1959 by Jack Kilby at Texas Instruments, was born as a reaction to the US Army's Micro-Module program that tried to stack individual components as functional modules (Mann, 2000). In his 1965 paper, Moore mentioned the Apollo manned moon flight program as a demonstration that integrated circuits can be as reliable as individual transistors.[4]

Moore went on to describe reduced costs as one main attraction of integrated electronics. According to Moore, manufacturing costs were dominated by two factors. First, the cost to manufacture simple chips was nearly independent of the number of components on the chip. In other words, the cost per component was nearly inversely proportional to the number of components. Second, Moore noted that the increased complexity of chips rapidly lead to decreasing yields in manufacturing. This quickly overwhelmed the cost advantages of cramming more components on the chip. As a result of these two cost drivers, integrated electronics had an optimal complexity of circuitry that led to most cost-efficient chips. As improvements in manufacturing led to better yields, the optimum complexity was growing accordingly.

In his 1965 paper, Moore makes two different arguments, which however become integrated. The first argument was based on the costs of making circuits with different complexities. Moore drew production cost curves for two years, 1962 and 1965, which showed that the cost per component was high if there were too few components on the chip or if there were too many. In 1962 the optimum number of components was about 10 whereas in 1965 it was about 50. Extrapolating this trend, Moore noted that during the next five years the manufacturing cost per component could be expected to drop to one tenth of the present cost and that a minimum-cost chip could have 1,000 components.

Moore then switched to existing chips, presenting them as minimum cost chips. Starting from Jack Kilby's 1959 one-transistor chip, and adding data from a circuit that was under development, Moore observed that the number of components on minimum-cost integrated circuits had increased roughly by a factor of two per year.

---

[3] See, e.g., (Schaller, 1996). Moore's contribution, however, was perhaps original in the sense that it combined economic and technological considerations. This original contribution, however, later become forgotten, as will be discussed below.

[4] Moore has, however, often remarked that military was not, in fact, an important driver in integrated circuit industry, at least after 1962 (e.g. Moore & Davis, 2000:6). Mowery and Rosenberg (1998) have made the same point, noting that consumer chips have been driving technology development, and that military has not played an important role in semiconductor industry.

He further noted that this would mean that by 1975 the number of components per integrated circuit would be 65,000. According to Moore, this would be a minimum-cost chip.

It is useful to note that Moore's estimate was based on his belief that there were no important bottlenecks that would slow down this process. Indeed, he argued that all the required technology was already available:

> "On the silicon wafer currently used, usually an inch or more in diameter, there is ample room for such a structure if the components can be closely packed with no space wasted for interconnection patterns. This is realistic… Such a density of components can be achieved by present optical techniques and does not require the more exotic techniques…" (Moore, 1965)

Similarly, Moore argued that there were no fundamental reasons why the yield could not be improved. According to Moore, it was not even necessary to do any fundamental research or to replace present processes. "Only the engineering effort is needed."

Moore concluded: "Clearly, we will be able to build such component-crammed equipment." The question to ask, then, was under what circumstances we should do it. In his 1965 paper, Moore deals with this question in only a couple of sentences. First, he notes that the total cost of making a particular system function must be minimized. One way of achieving this would be to mass-produce the chips so that the engineering costs could be amortized over many identical chips. Another way would be to develop flexible design techniques that could be used for many different chips. Moore, however, noted that increase in chip complexity did not necessarily lead to most cost-efficient solutions:

> "It may prove to be more economical to build large systems out of smaller functions, which are separately packaged and interconnected. The availability of large functions, combined with functional design and construction, should allow the manufacturer of large systems to design and construct a considerable variety of equipment both rapidly and economically." (Moore, 1965)

In its original form Moore's Law, then, says that the number of components on chips with the smallest manufacturing costs per component doubles roughly every 12 months. Moore further noted that existing technologies could keep this trend going for the next decade if engineering efforts would lead to improved yields in the manufacturing process. The fundamental assumption was that the total manufacturing costs are practically independent of the complexity of the chips. For this to be the case, the engineering and design costs had to be so small that they could be neglected. Indeed, Moore noted that the costs of integrated circuits were totally dominated by packaging costs. In other words, the costs of silicon was fixed and knowledge was free and the only limiting factor in manufacturing costs was the rapidly increasing waste created by deteriorating yields. Moore's discussion did not explicitly take into account investment costs. The costs of building and financing semiconductor manufacturing facilities were therefore implicitly assumed to be fixed and negligible.

From an economic point of view, Moore's Law was a rather fascinating law. It implied that the development of integrated circuits was completely determined by manufacturing costs. Moore's Law, therefore, defines a completely new economy. In this economy, demand is infinite.

3. Reformulations of Moore's Law

Around 1965, the integrated circuits manufacturing industry run into trouble in trying to build larger systems out of simpler components. The components tended to become unique and this lead to an explosion of costs. As Moore later noted:

> "Efforts were made to confront the part-number explosion directly. I remember at that time having discussions on how to design, manufacture, and test several hundred new part types every week in volumes of perhaps only 10 to 100 of each type." (Moore, 1979: 32)

It didn't, therefore, prove to be more economical to build large systems out of smaller functions. So, Intel took the other route discussed in Moore's 1965 paper.[5] The essence of Moore's argument had been that it was becoming possible to manufacture increasingly complex integrated circuits and that the price per component was dropping radically. The limiting factor would be efficient amortization of design investments. This could be done in two basic ways: either by making high volumes of single function or by making designs that could be used for many different chips. The first path led to Intel's focus on memory chips and the latter, a couple of years later, to microprocessors.

The microprocessor proved to be a major innovation. It combined the benefits of both high volume manufacturing and reuse of design work in high-volume multifunctional chips. It therefore made possible the amortization of design costs over not only existing markets but also emerging new markets.

Microprocessors also involved an important business innovation. The costs of semiconductor manufacturing dropped radically as universal microprocessors made the application developers pay for most of the design costs. Much of the difficulty and cost of designing complex systems was off-loaded to system and software designers. Similar logic, of course, worked with memory chips. The manufacturer did not have to know or care how the chips were used. In other words, the semiconductor industry solved its biggest problem by making it someone else's problem.

In 1975, Moore gave a presentation at the IEEE International Electron Devices Meeting in which he reviewed his earlier extrapolation.

---

[5] It is interesting to note that Moore's 1965 article contains the core elements of the business plan that was used by Moore and Robert Noyce to set up Intel in 1968. Indeed, the business logic discussed by Moore has been driving Intel's strategy for several decades since then. The Intel business plan was a one-page document written by Robert Noyce. According to Moore, its core idea was to make complex chips that could be manufactured in large volumes (Gibbs, 1997a). Intel, therefore, started its business with memory chips and then entered the calculator market, which led to the development of the first microprocessor.

Surprisingly enough, the integrated circuits introduced during the 1959-1975 period had followed the predicted trend relatively well. A new charge-coupled-device memory to be introduced in 1975 was to contain 16,384 bits.[6]

In his presentation, Moore analyzed the different causes of the exponential development. First, the physical size of the chips had been growing approximately exponentially. In 1975, chip sizes of the most complex chips were about 20 times larger than in 1959. Second, the miniaturization of component dimensions had also evolved at roughly exponential pace. This miniaturization had led to about 32-fold increase in component density in 15 years. The combination of increased chip size and component miniaturization therefore seemed to explain about 640-fold increase in the number of components. According to Moore's prediction, however, in 1975 chips were supposed to contain more than 640 components. The remaining 100-fold increase Moore associated with "circuit and device cleverness."[7] New technology, such as better isolation of components and innovations in circuit layouts had made it possible to pack more components closer to each other. (Moore, 1975)

Moore, however, estimated that the possibilities of circuit and device cleverness were already becoming exhausted. The future developments, therefore, were to be based on bigger dice and finer dimensions. Accordingly, Moore revised his original growth rate estimate and proposed that by the end of the decade, the number of components on the most complex chips would double about every two years. Soon after, this prediction became known as "Moore's Law." According to Moore, the name was coined by Carver Mead (Yang, 2000).

In 1975, Moore implicitly changed the meaning of Moore's Law. As he had done ten years before, he was still counting the number of components on semiconductor chips. Instead of focusing on optimal cost circuits, however, he now mapped the evolution of maximum complexity of existing chips. Indeed, in an article written a few years later, the famous growth curve is explicitly called "Moore's Law limit" (Moore, 1979). At that point the growth estimate is presented as the maximum complexity achievable by technology. In Moore's 1979 paper, which shows a picture with component counts of Intel chips manufactured in 1977 and 1978, most chips fall one, two, or even three orders of magnitude below this limit.

In the 1975 article, Moore's key message was that the development in semiconductor chips is still going to be fast "for the next several years" but that it was slowing down. The data that Moore uses to make his point is, however, somewhat confusing. He mentions that the last data point in his drawing corresponds to year 1975 and represents a charge-coupled device with 16,384 bits. In his drawing this point is located roughly at the 32,000-component complexity level. Moore then goes on and

---

[6] This 1975 data point is often presented as evidence that Moore's Law had been accurate. For example, Schaller (1996) notes: "16k charge-coupled-device (CCD) memory, indeed contained almost 65,000 components." I will discuss the characteristics of this chip below.

[7] Schaller (1996) notes: "Combining the contributions of larger die sizes and finer dimensions clearly helped explain increased chip complexity, but when plotted against the original plot by Moore, roughly one-third of the exponential remained unexplained." This is based on misreading of Moore's chart, which uses a logarithmic axis for component numbers. The missing contribution from "circuit cleverness" is two thirds from the total growth, not one third. In 1975, the factors that Moore had originally discussed in his 1965 paper had produced two orders of magnitude less components on a chip as originally predicted.

does his calculations as if the chip would have 64,000 components. Indeed, as noted above, he plots the increase in chip size for the most complex chips and observes that it has grown approximately 20 fold since the first planar transistor of 1959. This corresponds to chip size doubling in about 3.5 years. Then Moore plots the increase in device density due to decreased line widths and line spacing. Leaving out the first prototype chip, and averaging the decrease in line width and line spacing, he finds that these have decreased to about one fifth of their 1961 values. Assuming that component density grows as the square of this linear improvement, we could come up with 25-fold increase. Moore uses the value 32 for the 1959-1975 period, corresponding to a 3-year doubling time. As noted above, together these improvements imply a 640-fold increase and a 640-component chip in 1975. But then Moore makes a logical hat-trick. Without much advance warning he pulls out a magical 64,000 component chip that would almost fit his 1965 prediction of 65,000 components. He then calculates that the missing improvement factor needs to be 100. Moore notes that this contribution from "circuit cleverness" has actually been the most important factor in achieving the fictive 64,000-chip.

It is not completely clear how much magic is involved here, as it is not clear what was the most complex or cost-efficient chip introduced in 1975. At that year Intel introduced the 16-kilobit CCD, the Intel 2416. In the same year Intel also introduced its 2116-chip, a 16-kilobit dynamic random-access memory (DRAM) chip. Such a chip would have contained somewhat over 16,384 transistors, including some control circuitry, and about 16,384 capacitors. Since the mid-1970s, complexity has been counted based on the number of transistors. Moore's earlier calculations, however, were based on the total number of components.

In any case, the CCD circuit used by Moore as an example of current chip technology was not a very typical chip. It was used as a memory system under two different product names for a few months before its production was discontinued. Intel's first 64-kilobit device was the 2164, introduced in 1979.[8]

In 1979, Moore revised his analysis of development of semiconductors. He noted that in the 1965-1968 period the semiconductor industry had not been able to produce chips with the maximum complexity determined by the Moore's Law. There were basically two reasons for this. The first was that the increased complexity of chips typically required that the chip had to be connected in increasingly complex ways with other chips and components. The available packaging technology limited the number of connections that could be used for linking components. Although in theory it might have been possible to create complex chips, in practice the number of wires that came out from the chip package limited the complexity. (Moore, 1979)

The other limiting factor had been product design. Indeed, Moore presented a new exponential growth curve in his 1979 paper. According to it, the man-hours per month required for integrated circuit production was also growing exponentially. Moore went on to note:

> "If we assume that the cost in man-hours per month is inflating at 10 per cent per year (a conservative figure considering the need for increased computer

[8] http://intel.com/intel/intelis/museum/arc_collect/TimelineChron.pdf

> support, software, etc.), then the costs double every two years…This cost can be contrasted with manufacturing costs, which are largely independent of device complexity. Whereas manufacturing costs were once dominant and exceeded those of design, the situation is now reversing, with design costs becoming dominant.” (Moore, 1979: 35-6)

Moore also noted that the problems that slowed down the growth of semiconductor complexity in the 1965-1968 period had not been solved. Engineers were still unable to design and define products that would have used silicon efficiently. Instead, the industry was saved by inventing two product categories where problems could be avoided:

> “In general, the semiconductor industry’s efforts to solve its problems in the 1965-1968 era were not successful. The product definition crisis persisted and limited IC complexity through the mid-60s. Two things broke the crisis for semiconductor component manufacturer, though not necessarily for the mainframe computer manufacturer: the development of the calculator and the advent of semiconductor memory devices.” (Moore, 1979: 33)

The calculator was an important product as it was a relatively simple system. Connecting four integrated circuits that had about 40 pins could make a calculator. The interconnection problem, therefore, was tractable. As calculators were produced in high volumes, the design costs could be justified. Memory chips, in turn, were easy to design and universal in their functionality, and therefore also high volume products with low design costs. Moore went on to note:

> “Thus the interconnections and product-definition problems of the past were not necessarily solved; they were simply circumvented. The semiconductor industry developed a different set of markets to keep itself busy, postponing the solution of its previous problems.” (Moore, 1979: 34)

Moore’s 1979 paper, therefore, makes an important amendment to the 1975 Moore’s Law. Moore still claims that the complexity of the most complex commercially available chips had for the first 15 years been doubling roughly every year and, after that, doubling every two years. Now he however explicitly notes that the curve represents the limit of achievable complexity and that most chips fall far beyond the curve. Whereas the reason for this slowing down was in 1975 described as disappearing “circuit cleverness,” in 1979 Moore noted that the essential problem was in designing products that would benefit from the maximum complexity. This was becoming the dominant concern as the industry was attempting to move from large-scale integrated circuits, LSI, to very large-scale integration, or VLSI. As Moore put it:

> “As for my original question—is the semiconductor industry ready for VLSI?—the conclusion is that for maximum advantage, both component and system suppliers must address the problems of product definition and design. In fact, unless both industries—the semiconductor component as well as the systems company—address and solve these problems, as we look back on the VLSI era we may only be able to say ‘thanks for the memories.’”(Moore, 1979: 37)

Two years later the semiconductor industry went through a contraction, recovered, and then hit a major crisis in the 1984-1985 period (Gibbs, 1997b). Intel withdrew from the DRAM memory business where it had been a technology leader for 15 years. The increasing competition and over-capacity meant that demand had become relevant again. Moreover, financing started to play an important role. As Moore recalled later:

> "Intel had just developed the next-generation product and process. It was the one-megabit DRAM at that time and we had a leading-edge product. But in order to put that in production and become a significant market share participant, we estimated we'd have to spend $400 million on new facilities…And with nobody making money and no obvious prospect of making money, the decision not to invest $400 million was very easy, and the consequence of that was we had to drop out of the DRAM business." (Schmitt, 2000)

4. Losing the memory

Since the 1980s, Moore's Law has been used often and in many different ways. For example, Intel described Moore's Law on its web site in April 2002 in the following way:[9]

> "Gordon Moore made his famous observation in 1965, just four years after the first integrated circuit was discovered. The press called it "Moore's Law" and the name has stuck. In his original paper, Moore predicted that the number of transistors per integrated circuit would double every 18 months. He forecast that this trend would continue through 1975…Moore's Law has been maintained for far longer, and still holds true…"

Moore himself has noted:

> "I never said 18 months. I said one year, and then two years…Moore's Law has been the name given to everything that changes exponentially. I saw, if Gore invented the Internet, I invented the exponential." (Yang, 2000)

The historically inaccurate 18 months doubling time has been extremely widely used. It is possible even to find fictive quotes of Moore's 1975-presentation saying: "The number of transistors per chip will double every 18 months." The 2001 edition of the International Technology Roadmap for Semiconductors makes this claim, defining Moore's Law as doubling of components per chip every 18 months (ITRS, 2001:1) and arguing that the industry will continue to stay on the trend. A recent IBM press release makes the same claim:

> "The evolution of semiconductor technology has traditionally followed a trend described by Moore's Law, an industry axiom that predicts that the number of

---

[9] http://www.intel.com/research/silicon/mooreslaw.htm. After April 2002, Intel changed the text on its web site by removing references to specific doubling times.

> transistors on a chip will double every 18 months, largely due to continued miniaturization known as scaling." (IBM, 2001)

A Scientific American article on Moore's Law (Stix, 2001) starts by saying:

> "When Gordon Moore, one of the founders of Intel, plotted a growth curve in 1965 that showed the number of transistors on a microchip doubling every 18 months, no one had any idea that his speculations would not just prove prescient but would become a dictate—the law by which the industry lives or dies."

A widely quoted version of Moore's Law was provided by R.X. Cringely (1992) in his Accidental Empires. According to this version, Moore's Law states that the number of transistors per square inch on integrated circuits doubles every eighteen months. This version combines two historical errors, first, by associating Moore's observation with the 18 month doubling time and, second, by forgetting that Moore also included increases in chip size in his calculation. Eric Raymond (1999) makes a similar conversion from components to component density and gives an exact mathematical form for Moore's Law, defining it as "The observation that the logic density of silicon integrated circuits has closely followed the curve (bits per square inch) = 2^(t - 1962) where t is time in years." Raymond further notes that this relation was first uttered by Moore, in 1964, and that it held until the late 1970s, at which point the doubling period slowed to 18 months. According to the most detailed historical study focusing on the issue (Schaller, 1996), Moore found in 1975 that circuit density was doubling every 18 months.[10]

It is therefore not surprising that less technology oriented authors have been confused. A World Bank strategy report notes: "Gordon Moore's Law - microcomputer chip density doubles every year or so - has held for 30 years"(Talero & Gaudette, 1996). The European Competitiveness Report (EC, 2000), prepared by the European Commission, notes: "According to the so-called Moore's law, microchip capabilities double every 18 months."

Several economists who have specialized on analyzing the economic impacts of information technology make similar mistakes. For example, Jovanovic and Rousseau (2002) start their article on Moore's Law by stating: "In 1965, the co-founder of Intel, Gordon Moore, predicted that the number of transistors per integrated circuit would double every 18 months." DeLong (2001), in turn, argues that despite the recent collapse in internet-company stocks, there are good reasons to expect that a new economy is, indeed, emerging, driven by Moore's Law. DeLong defines the law as "the rule of thumb that Intel cofounder Gordon Moore set out a generation ago that the density of circuits we can place on a chip of silicon doubles every eighteen months with little of no significant increase in cost," and notes that "Moore's Law has held for thirty years; it looks like it will hold for another ten at least."

As noted above, Moore never claimed that the number of components would double every 18 months.[11] The first version, the doubling of components on a chip every

[10] In fact, as noted, Moore found that circuit density had doubled in about 36 months.

[11] He mentions 18 months in a keynote speech given in 1997 (Moore, 1997), saying: "We can expect to see the performance of our processors double every 18 to 24 months for at least several years."

year, would mean that the number of components would increase 1024-fold per decade. The second version, doubling every two years, would translate into a much more modest increase of 32 per decade. In fact, the International Technology Roadmap for Semiconductors (ITRS, 2001) uses as the basis of its near-term microprocessor forecasts three year doubling time.[12] A three year doubling-time means that the number of transistors on a chip increases about 9-fold in a decade.

Over several decades the differences obviously increase dramatically. During the four decades of validity often claimed for Moore's Law the difference between one year and three year doubling time means about 8 orders of magnitude. In other words, to get the same number of transistors, the slower growth path would require us to buy 100 million chips, instead of one. So, although a few months more or less in the doubling rate might not appear to be a big deal, actually it is.

Whereas Moore's observation was originally about the number of components on a lowest-cost chip, it was, however, quickly extended outside this well-defined area. There have been three main extensions, which all transform Moore's Law qualitatively. First, Moore's Law has been defined as the "doubling of processing power on a chip every 18 months" (e.g. Gates, 1997). Second, it has been defined as "doubling of computing power every 18 months" (e.g., Gore, 1999). Third, it has been defined as "price of computing power falling by half every 18 months" (e.g., Gordon, 2000).

Although these extensions are often freely mixed, they imply quite different claims. The first extension adds two new elements to Moore's original considerations. Processing power depends fundamentally on the amount of information that can be handled within a unit of time. This introduces the dimensions of time and functionality into Moore's law. Whereas the Moore's original formulations only counted components, this revised version already has an implicit theory about their use.

The second extension adds the assumption that computing power evolves at the same pace as processing power of individual chips. It extends Moore's Law to include computer systems that comprise of multiple different hardware technologies, as well as software.

The third extension adds to these an economic dimension. It argues that the market cuts the price paid for computing power at the same rate as "Moore's Law." A simpler variation of this reformulation is that the cost per component halves roughly every 18 months.

One should note that any of these claims might be right, independent of the fact that they historically misrepresent Moore's Law. Moreover, although the logic may be disconnected between any two steps in the chain of extensions from component counts to new economy, any of the claims might still be empirically valid. It is therefore interesting to see to what extent this is the case.

---

[12] The ITRS editors apparently find no contradiction between this estimate and their definition of Moore's Law.

5. Empirical evidence on Moore's Laws

It is important to note that even in its simplest form Moore's Law actually may be interpreted in several different ways. Moore's 1965 prediction that in 1975 the most cost efficient chips would incorporate 65,000 components was, in fact, wrong. The chip that Moore used in his 1975 presentation to show that his earlier prediction had been accurate, was a charge-coupled memory device that was just about to be introduced and it didn't have 64,000 components. Such a chip, therefore, strictly speaking did not exist. Furthermore it probably did not represent the most cost efficient chip, as it was quickly withdrawn from production. Indeed, the cost structure of different chip types had already become differentiated. In 1975 Intel introduced its first general purpose 8080 processor that started the PC revolution. It had 4,500 transistors.

The empirical support for Moore's Law strongly depends on two important factors. First, there exist many types of silicon chips and they have historically evolved differently. For example, the number of transistors has typically grown much faster on memory chips than on microprocessor chips. Second, each chip usually has a life cycle of about a decade, during which its manufacturing costs and selling price radically change. It is easy to create approximately linear growth trends on semi-logarithmic charts if the data points can be moved on the linear axis. Indeed, most graphical representations of Moore's Law do not specify how they locate different chips in time. Moore's original claim was that the number of components on chips with lowest manufacturing cost was doubling yearly. This allowed for the fact that there were other chips that were more expensive, for example, because they had more components. Moore's observation, however, was not only a claim about the number of components but also explicitly a claim on the timing of their optimum cost.

There are many ways to locate specific chips in time. This can be seen in Figure 1, which shows the shipping volumes and prices of 16-kilobit DRAM memory chips as a function of time. Using the availability of 16-kilobit DRAMs, we can say that DRAM chips had 16 kilobits in 1976. If we use the shipped volume of these chips, which presumably reflects optimal marginal costs of production, we can say that 16-kilobit chips perhaps should be located in 1983.

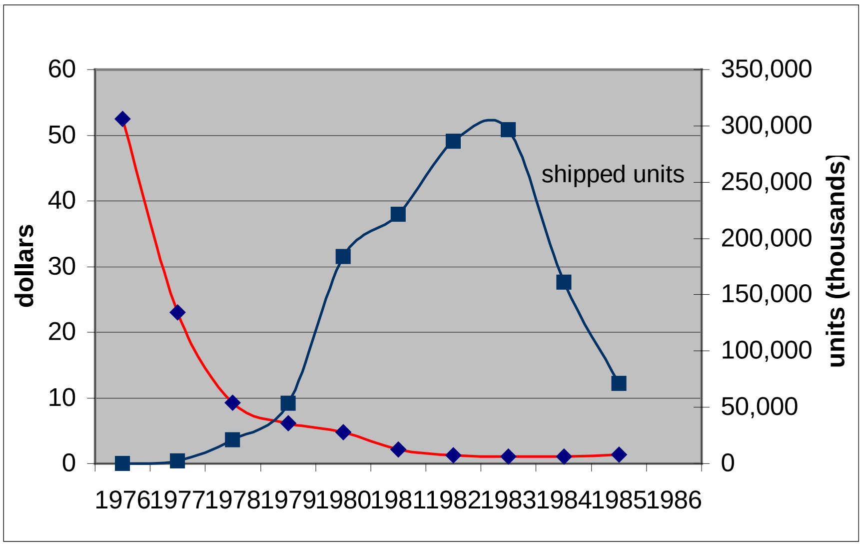

Figure 1. Prices and Quantities of 16 kilobit DRAM chips. (source: Grimm, 1998)

As specific chip types usually have a long lifetime during which the costs and other parameters of the chip change, the ITRS roadmap differentiates four main life-cycle phases. The first is the *year of demonstration*. This is the year when the feasibility of a specific characteristic, for example the number of transistors on a single chip, is first demonstrated. The second phase, *market introduction*, usually two or three years later, is defined to occur when the leading manufacturer ships small quantities of engineering samples. The third phase, *production*, is defined to occur when the leading manufacturer starts shipping the chip in volume quantities and a second manufacturer follows within three months. The *lowest cost* phase emerges when the production processes have been optimized and competition does its work. For example, the 1-gigabit DRAM was demonstrated in 1997, introduced in 1999, and is expected to be in volume production in 2003 (ITRS, 2001:56). Similarly, the Intel Itanium processor was announced in 1994, was originally planned to be on market in late 1997, but was delayed and became commercially available in 2001. Market researchers currently project that Itanium will garner less than 10 per cent of the market for server computing in 2007 (Markoff & Lohr, 2002).

Transistor count

With these reservations in mind, a simple test of Moore's Law would be to see whether the number of transistors actually follow an exponential growth curve. In other words, we can ask whether the number of components on a chip double regularly. A first approximation can be found using data provided by Intel. We can simply use data published on the Intel web page that describes Moore's Law.[13] According to the provided data, the first microprocessor 4004 was introduced in 1971, with 2,250 transistors. The last available data point at the time of writing was Pentium 4, with 42,000,000 transistors. Without doubt, technology had advanced rapidly.

[13] http://www.intel.com/research/silicon/mooreslaw.htm

Using this data we can fit an exponential growth curve and see how good the fit is. Figure 2 shows the percentage differences in number of transistors from their predicted values. According to this simple fit, there were some 5 million transistors missing from Pentium II in 1997, representing about 70 per cent, and some 7.5 million, or some 18 per cent, too many in the Pentium 4 in 2000. The estimate for Pentium 4 comes relatively close to its actual value as the data, indeed, has been used to fit the exponential, and the final point is heavily weighted. The error would be greater if we were to use previous points to fit the curve and then predict the number of transistors of Pentium 4.

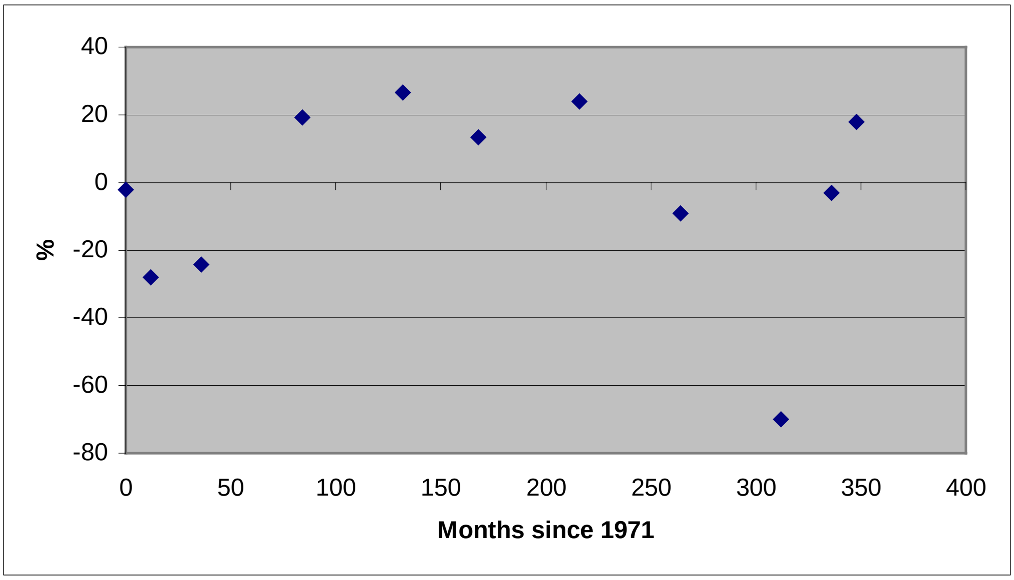

Figure 2. Percentage difference in expected number of transistors in Intel microprocessors.

An alternative simple test for Moore's Law would be to see whether the number of transistors actually have been doubling every 18 months or every two years. One would expect that many people who refer to Moore's Law would have done this exercise. Actually, it seems to have been done relatively rarely. Indeed, one of the few exceptions is a comment written by Brenner (2001). It notes that a figure published in an earlier Physics Today article (Birnbaum & Williams, 2000) shows a doubling time of about 26 months, whereas the caption text claims that "Moore's Law is the empirical observation that the number of transistors on a single integrated-circuit chip increases by a factor of four every three years," implying that the data shows doubling time of 18 months.

Leaving aside for a while the question how well the above mentioned graph with 26 months doubling period reflects available evidence, it is evident that Moore's Law has become so commonly accepted that it is possible to publish articles that contain both its historically incorrect version and data that contradicts the provided version. The discrepancy between the standard version of Moore's Law and reality, however, should not be particularly difficult to see. Starting from 1971 with 2,250 transistors and using 18 months doubling time leads to about 1.4 billion missing transistors in year 2000. This represents about 3,400 per cent error. If one uses the 2-year doubling period the error is less dramatic. Only about 10 million transistors are missing from Pentium 4. Pentium II had somewhat more missing transistors in 1997, corresponding to about 150 per cent too few transistors on chip.

The real story is, of course, more complex. During the last three decades, processor architectures have changed considerably. Starting with Intel's 486 processor series, so-called cache memory began to be included on the same silicon die as the processor. Processor chips, therefore, became a combination of processors and memory. As memory chips have a considerably higher density of transistors than microprocessor chips, this combination of memory with processors led to rapid increase in the number of transistors on such integrated processor chips. The fast memory that was included on the chip was called L1 cache. For almost a decade, this cache memory was supplemented by fast external memory chips, called L2 cache. Due to the hight cost of fast L2 cache memory chips, Pentium Pro and Pentium II chips were packaged in a multichip packages that contained both the processor and L2 cache memory. Starting from the second-generation Celeron processors, this additional memory was integrated with the same silicon chip as the processor. This meant that whereas the 300 MHz Celeron chip, introduced in June 8, 1998, had 7.5 million transistors, the second generation 300A-version of the processor, introduced a couple of months later, had 19 million transistors. The added transistors were on-chip memory. There has also been an increasing variety of processors within each processor family. For example, Pentium II was released in 18 different variations, including mobile versions.

A simple test for the validity of the Moore's Law can be done by charting the number of transistors on microprocessor chips. This chart is, basically, the one that is usually shown with the claim that it proves that the number of transistors doubles roughly every 18 months. The results can be seen in Figure 3. The data for the figure comes from Intel's "Microprocessor Quick Reference Guide" that lists all processors with their basic technical specifications.[14] In contrast to most presentations of transistor counts, Figure 3 includes all Intel microprocessors, until the Pentium III and Celeron processors. The figure also shows three exponential trends that have been fitted to the data.

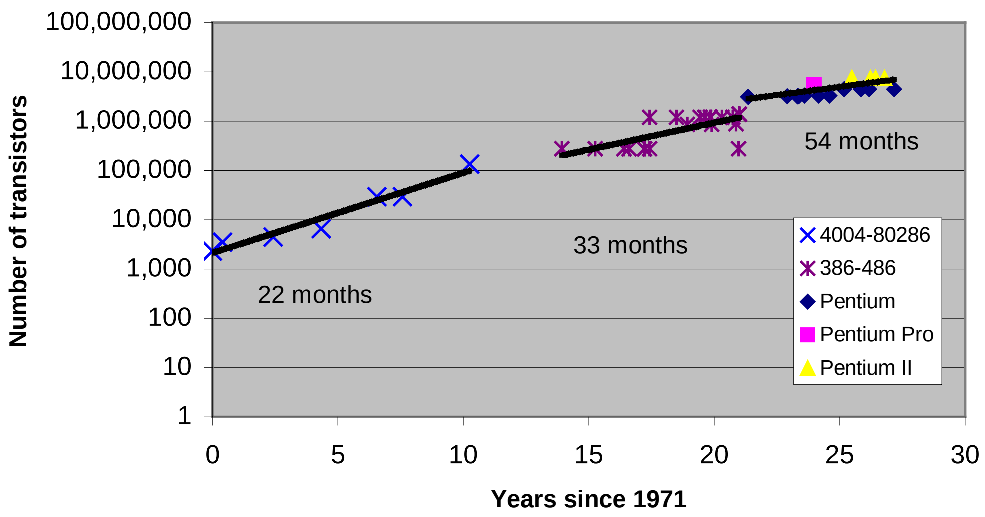

Figure 3. Number of transistors on Intel microprocessors.

[14] http://www.intel.com/pressroom/kits/quickrefyr.htm

As can be seen from Figure 3, the number of transistors has not doubled very regularly. During the first decade of microprocessors, the doubling rate was approximately 22 months but also very irregular. After the introduction of the 80386-processor family, the doubling speed was closer to 33 months. During this period, the number of transistors jumped from 275,000, on the Intel 80386-chips in 1988 to 1.4 million transistors on the 80486 SL-chips, at the end of 1992. During the Pentium, Pentium Pro and Pentium II processor families the transistor count doubled roughly at a rate of 54 months. Strictly speaking, the transistor counts, however, have changed irregularly and the mentioned doubling times are based on statistical trends.

Since late 1999, Intel has not included transistor counts in its processor summaries. In October 1999, Intel Pentium III Xeon and Mobile Pentium III processors had some 28 million transistors. In July 2001, Pentium 4 had about 42 million transistors. Most of these transistors were cache memory. During the last couple of years, the transistor counts on microprocessors have increased very rapidly due to the fact that they have closed their earlier gap with memory chips. Indeed, it seems that Intel is again in the memory business.

Processor performance

The first qualitative revision of Moore’s Law is affected by similar concerns. In its simplest form it says that the number of instructions per seconds, expressed as MIPS (millions of instructions per second) doubles every 18 months. Indeed, the famous 18-month doubling time may have its origin in such calculations. Moore has noted that a colleague at Intel came up with this number by combining Moore’s 2-year rule and increases in processor clock frequency (Yang, 2000). The increase in clock frequency means that the chip can do calculations more rapidly. Other things equal, the combination of increase in chip complexity and clock speed might, therefore, lead to a doubling of processing capability in less than two years.

One problem with this approach is that the clock speed is not directly related to the amount of information that is processed. For example, the original Intel 8088 PC microprocessor took on average 12 clock cycles to execute a single instruction. Modern microprocessors can execute three or more instructions per clock cycle.

Another problem is that there is no simple way to measure processing power. In the early 1970s, computer processing power was often expressed as the cycle time of its memory. Towards the end of the 1970s, Vax MIPS became a common benchmark for integer operations and Vax MFLOPs for floating point operations. Towards the end of the 1980s, standard test program suites were developed to measure processing speed. During the revisions of these program suites they have intentionally been made incompatible with earlier benchmarks.

The reason why the benchmark programs have been revised is that the definitions of processing power constantly change. Processor architectures are designed for specific types of problems. As the use of microprocessors change, the types of programs that are actually run on them change as well. This means that there is no direct link between processor clock time, number of instructions executed, and processing power. Indeed, Intel’s competitor AMD has repeatedly made the point that the benefits from increasing the clock speed have rapidly slowed down. According to AMD the accelerating increases in clock frequency in Intel's chips mainly reflect architectural problems and diminishing returns for processing power (e.g., AMD, 2001). In other words, AMD says that Intel has to tick its clocks faster and faster to keep up with the architectural performance improvements made by its competitors.

With the understanding that the clock frequency does not really measure processor performance, one may, however, check how the clock frequency has evolved during the lifetime of microprocessors. Information on average processor clock speeds has been collected by Berndt et al (2000). This is shown in Figure 4. The data covers distinctly identified personal computers that have been marketed and sold in the US. Processor speeds for mobile computers are excluded from Figure 4.

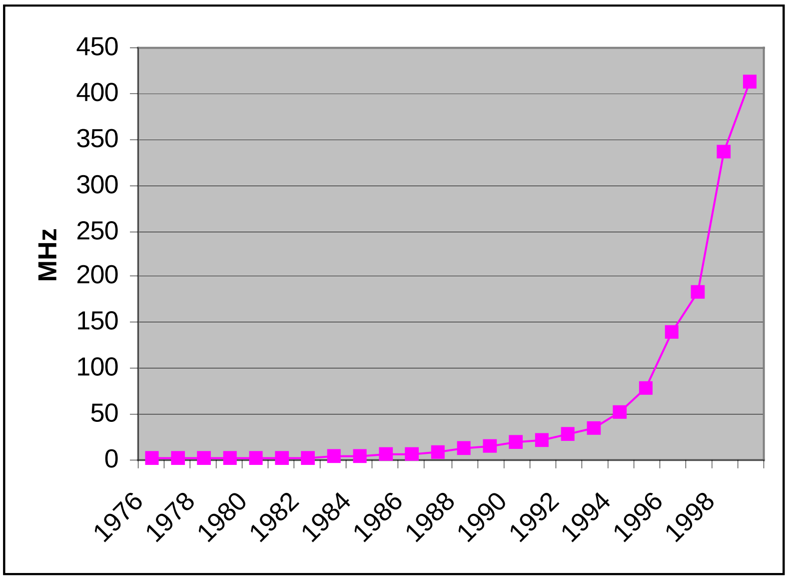

Figure 4. Desktop computer processor speed. (source: Berndt et al, 2000, Table 1).

Figure 4 has several interesting characteristics. First, it should be noted that the data shown do not directly represent advances in microprocessor technology. It is based on computers that have been marketed to end customers. In that sense it does reflect changes in the actually available processing power. As can be seen from the figure, until the end of 1980s the increase in the reported processor speed was quite modest. During the first decade, processor speed grew about 4-fold and between 1986-1995 somewhat less than 10-fold. In about 1994, the clock speed started to grow much more rapidly.

It is therefore clear that a simple combination of increasing clock speed and number of transistors on chip could not have produced the famous 18-month doubling constant. The 18-month constant was commonly used already in the 1980s when clock speed was increasing only slowly.

Second, the rapid increase in processor speed at the end of the 1990s is an interesting phenomenon in itself. Indeed, it indicates that the microprocessor industry was not driven by its own technical advances anymore. The fact that processor speeds started to grow rapidly around 1994 may be related to the increased demands of operating systems and office applications used with Intel processors. The continuation and acceleration of this trend may be related to increased consumer interest in multimedia and audio applications.

During the first two decades of microprocessor history, processing power was often expressed as millions of instructions per second (MIPS). As the actual processing power depends on the real throughput of information processing, which depends on the amount of data processed as well as the number of processing steps, there is no direct link between MIPS and processing power. This is easy to see. One million instructions per second lead to very different processing power if each instruction can handle one bit or a thousand bits at a time. Processing power depends on the amount of data that can be accessed and operated on, the set of instructions available, and the complexity of operations that each instruction can accomplish.

Intel microprocessors used for many generations so-called complex instruction sets. As long as the basic architectural principles remained relatively stable, it was possible to characterize processing power using MIPS. For Intel processors this data is available until June 1995. Figure 5 shows the processor MIPS ratings as a function of processor introduction date.

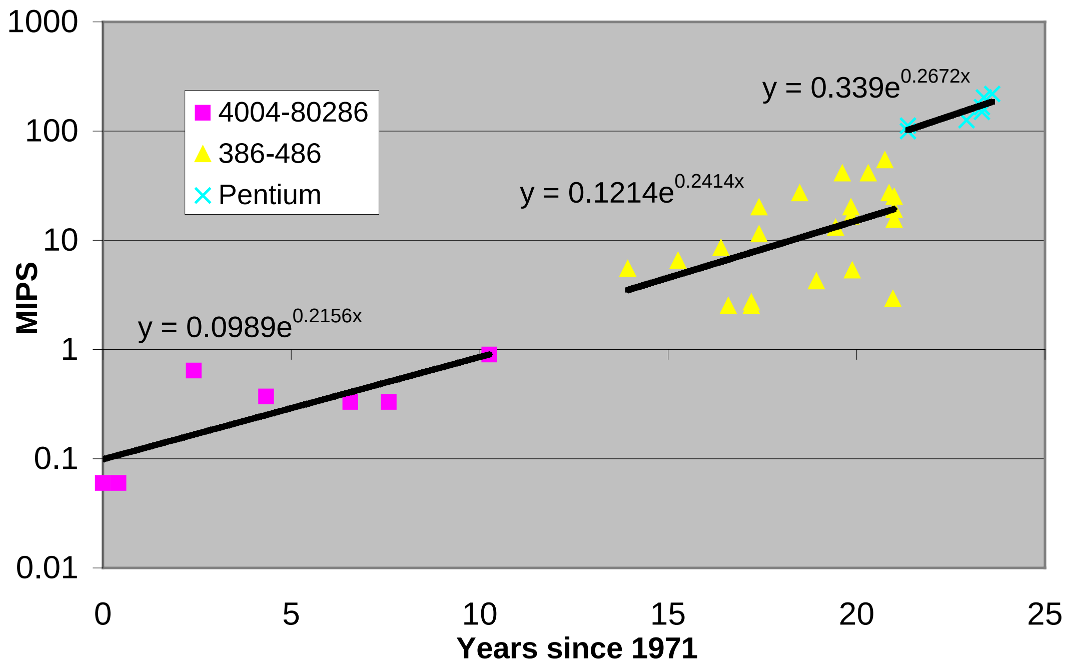

Figure 5. Processor performance in millions of instructions per second (MIPS) for Intel processors, 1971-1995.

As can be seen from Figure 5, MIPS ratings of Intel processors have not increased exponentially in time. There has been considerable variation of MIPS ratings across different processors. This is, of course, related to the fact that processors have been often released in many different performance and price categories. For example, the Intel 80386 processor was during its lifetime sold at least in 10 different varieties. Some of these maximized processor performance, others optimized for power consumption or cost.

Fitting exponential trends to the data, it seems that MIPS ratings for Intel processors grew about ten-fold during the first ten years, in other words, doubling in somewhat more than 3 years. Due to the large variety of the 80386 and 80486 processors, exponential fit is not very good for the next ten-year period but the best fit has about the same rate of doubling. The small sample of data available for the Pentium processors indicates a somewhat faster rate of increase.

Figure 5 also shows a gap after the introduction of the 80286-chip in 1982. Until that time, microprocessors were only a small fraction of Intel's revenues, which mainly came from memory chips. According to Moore, microprocessors were slow to gain traction in the market and volume sales only started after the introduction of the 80286:

> "1983 was when the volume really started to develop, but that was when the world thought their usage was going to go through the moon. 1983 was one of these wild expansion years for our industry, and everybody was telling us what they were going to need, and it was three times what we ever thought we could make." (Moore, 1999:13)

Preparing for the rapid expansion of markets, Intel licensed the 80286 microprocessor to other manufacturers, including AMD, Fujitsu, Siemens, and IBM. "So it was a big boom in 1983, into the middle of 1984—and then the semiconductor world collapsed in late 1984 and 1985. It was a big recession." (Moore, 1999:13) According to Moore, the entire industry put in capacity as fast as it could and the demand turned out be something like a third of what the customers had predicted. "So with huge over-capacity, the prices just collapsed and we had to get rid of a third of our work force, shut about a dozen factories. Those were traumatic events because they impacted people's lives so strongly."(Schmitt, 2000)

By 1985, then, demand had started to be less than infinite and the semiconductor industry was not endogenously driven by technology. In 1982, the increase in MIPS ratings stopped for about three and half years instead of following an exponential trend. As Moore had noted in his 1979 paper, there was an increasing mismatch between technological possibilities and the ways technology could be beneficially used.

Increase in computing power

During the last decades, computer clock frequency and the number of instructions per second have become very inaccurate indicators of processor power. Since the Intel 80286 processors shipped in1982, microprocessors have utilized parallel processing in many alternative forms. By parallelism, more operations can be accomplished within a time unit. For example, the processor can be loaded with instructions that contain several operations, it can have several execution units that process multiple instructions within one cycle, or the processing of operations can be started before the previous operations have finished. All these forms of parallelism have commonly been used since the mid-1980s. Moreover, since the 1990s, processor architectures have increasingly relied on program compilers that detect and optimize parallelism in the source code programs. Indeed, the innovations in compiler technology have been a main driver in processing power improvements. Processing power, therefore, is increasingly determined by software that compiles computer programs into machine code. In short, today it is impossible to say what the processing power of a given microprocessor is by studying the characteristics of the chip itself. The MIPS ratings, for example, have become quite irrelevant for measuring processing power.

When Moore's Law has been extended from processor performance to computing power, its scope has been extended by yet another radical step. Despite a common misconception, microprocessors are not single-chip computers. A functional computer needs input and output devices, memory, data to work on, and a program that defines how the data are handled. Microprocessor architectures have constantly been revised to optimize the division of labor between the different computer system components. Indeed, in their 1989 paper on the future evolution of microprocessors, Gelsinger and his colleagues (1989) argued that microprocessors have internalized many of the architectural ideas first developed for mainframe computers.

Computing power is rarely determined by the capabilities of microprocessors. Usually, the microprocessor is connected to external memory and input and output devices with links that are an order or magnitude slower than connections within the chip. Moreover, microprocessor architecture has a major impact on the speed by which different types of computing problems can be handled. For this reason, several different types of processors have emerged. In the 1980s, personal computers, for example, often contained separate processors for floating-point operations and integer operations. Today, a personal computer may contain multiple device controllers, digital signal processors, and main processing units. Main processing units, in turn, may combine large amounts of memory, hard-wired programs, as well as logic circuitry within a single chip. A functional computer also contains different types of memory and programs that have been compiled and configured for the machine in question.

This complexity is reflected in computer processing power measurement. One important computer processing measurement initiative has been organized around the non-profit Standard Performance Evaluation Corporation (SPEC). It was set up in 1988 by a group of workstation vendors "who realized that the marketplace was in desperate need of realistic, standardized performance tests."[15] The first SPEC measurement suite was called SPECmarks and published in 1989. The suites were revised in 1992, 1995, and 2001. A separate suite has been used for integer operations and for floating point operations. The CINT92 suite for integer operations contained 6 programs, written in C language, and the CFP92 for floating point operations contained 14 programs, mainly written in FORTRAN. The 1992 revision was partly done because computer manufacturers had started to skillfully write compilers that created machine code optimized for the benchmark programs. Some programs in the test suite were observed to measure mainly the skill of compiler programmers and had little to do with processor performance. The controversies around the SPEC benchmark suites are still going on. In general, the main use of benchmarking results have been in marketing. In contrast to the SPEC benchmarks, the Business Applications Performance Corporation (BAPCo) has developed benchmark suites that imitate typical business uses of computers. One of the most popular BAPCo

[15] SPEC's background: http://www.specbench.org/spec/

benchmark suites, is the SYSmark.[16] It measures computing performance by running specified tasks with Windows applications.

The attempt to develop measurement systems for computer processing power have made it clear that the definition of computing power depends on the tasks for which the computer is used. Therefore there is no well-defined criterion or data for arguing that computer power would have increased exponentially. On the contrary, it has frequently been argued that most of the increase in computer capabilities has been consumed by software. This is often formulated as Wirth's law: "Software gets slower faster than hardware gets faster."[17]

Moore himself has noted:

> "Fortunately, the software industry has been able to take advantage of whatever speed and memory we could give them. They taken more than we've give, in fact. I used to run Windows 3.1 on a 60 megahertz 486, and things worked pretty well. Now I have a 196 megahertz Pentium running Windows95, and a lot of things take longer than they used to on the slower machine. There's just that much more software, I guess." (Gibbs, 1997c)

Price and cost of computing

Many versions of Moore's Law avoid technical details and try to make the law understandable for general audience and policy makers using economic terms. Strictly speaking, it is not clear why an average person should be interested in the number of transistors on silicon chips. The extensions that translate the number of components into "processing power" perhaps help in clarifying that transistors mean progress. Eventually, however, the value of technical change is supposed to be visible in our everyday life. Economic considerations have historically been an important benchmark for such progress.

The economic versions of Moore's Law, therefore, essentially claim that "you get more by less." They also claim that this change is exponential, usually specifying 18 months as the time it takes for computing power to double at a constant price. This extension, therefore, brings in theoretical and empirical elements that did not exist in the previously discussed versions of the law. Instead of counting transistors on a chip or benchmarking the number of instructions that a microprocessor or a computer system handles in a unit of time, the cost of computing involves prices.

The basic problem in measuring prices in computer technology is technical advance. A personal computer that sells for 1000 dollars or euros today has much more functionality than a personal computer five years ago, sold at the same price. Technical advance, therefore, has to be accounted somehow. When we compare computer prices from two time periods, we have to adjust for qualitative changes before we can say how the price has changed.

---

[16] http://www.bapco.com/
[17] See, e.g., Metcalfe (2000).

This problem is evident when we compare the prices of desktop computers over the last three decades. The median value of desktop computer prices is shown in Figure 6. Without quality corrections, we could say that the price of computing roughly doubled in the 1976-1990 period, and then dropped back close to its 1978 value in 1999. This, obviously, could not be used as an argument to support Moore's Law.

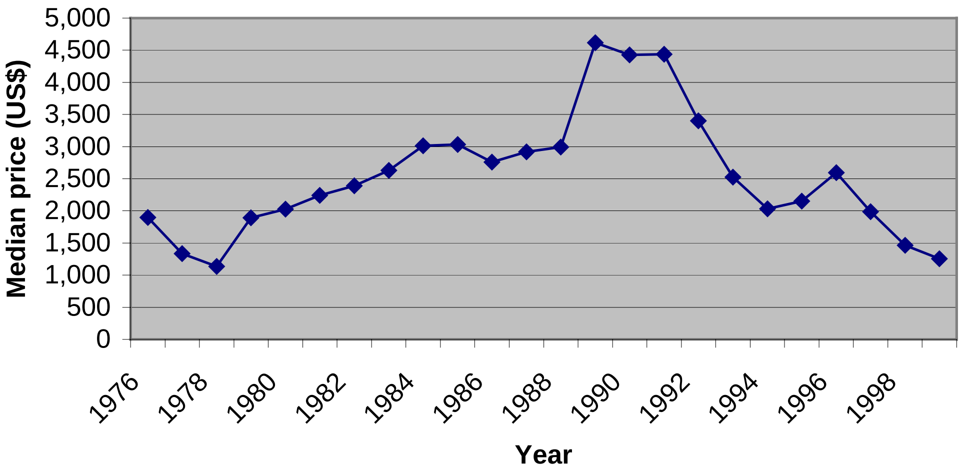

Figure 6. Median price for desktop computers sold in the US. (source: Berndt et al., 2000, Table 1)

Clearly, there have been huge qualitative changes in desktop computers during this time. The problem, then, is how to take them into account.

One approach is to create so-called matched-model price indexes. It is possible to measure price changes of a given computer type across several years and deduce from this the actual price change. So, instead of looking yearly changes in the list price of desktop computers, as in Figure 6, we can look how the price for a given PC configuration has changed. Using such data, we can fit a statistical model that estimates the price changes associated with technical change in the various computer components, for example, in memory capacity and processor speed. This gives us an index that we can use to correct market prices so that they reflect quality improvements. In effect, we add to Figure 6 the value that has been created by technical change. Using this procedure we can then say how the price and cost of computing have changed.

Another possibility is to compare different existing computer configurations to deduce the contributions that different computer characteristics make to the total price. For example, the price difference between two similar computers, one with a 60 MHz and another with a 100 MHz Pentium processors, tells us something about the value of increase in processor speed. Using a large set of computers with different characteristics it is possible to develop so-called hedonic price models, which define the price as a function of characteristics. Comparing quality changes across years, such hedonic price models can give information on the quality-adjusted prices. One may then estimate the quality-adjusted price changes in computer products. In a simplified way, if a 100 MHz PC costs today 500 dollars more than a 60 MHz PC, we might assume that if a 100 MHz PC costs today as much as a 60 MHz PC a year ago, technical advance has been worth 500 dollars.

Quality-adjusted prices and hedonic price indexes have been studied extensively during the last two decades. Indeed, they have been used to correct national accounts since 1987 in the US.[18] The proper measurement and calculation of computer price changes is quite difficult and much has been learned about this during the last decade. One could therefore ask two independent but interesting questions about the validity of the economic versions of Moore's Law. The first is quite simple: Given available evidence around the beginning of the 1990s, could one have reasonably argued that the price of computing was halving every 18 months, taking quality improvements into account. Another question is: Could we today, with currently available knowledge, make the same argument.

The first question is relatively easy to answer, as it does not require any discussions of the particular models and methods used in analyzing quality corrected prices. The classic study of quality corrected prices in computing is by Chow (1967), who analyzed mainframe rental prices in the 1960s. According to Chow, quality-adjusted prices fell at an average annual growth rate (AAGR) of about –21 percent during the 1960-1965 period. Cole et al. (1986) studied the price declines of different computer components and found that over the 1972-1984 period, the AAGR for computer processors was –19.2 percent using the hedonic prices and –8.5 percent using the matched model procedure. Cartwright (1986), in turn, reported an AAGR of –13.8 percent from 1972 to 1984. According to Gordon (1989), quality adjusted mainframe prices fell 22 percent annually from 1951 to 1984. Triplett (1989) summarized earlier hedonic studies on mainframe computer prices and reported a "best-practice" quality-adjusted price decline of –27 percent over the 1953-1972 time period. Gordon (1990) then extended his earlier analysis to personal computers and reported 30 percent annual declines from 1981 to 1987. Berndt and Griliches (1990) collected a large sample of data on personal computers and reported 28 percent annual decreases from 1982 to 1988.

At the beginning of the 1990s, it would then have been possible to argue that the price of computing had been dropping 20 to 30 percent yearly. This translates to halving in every 45 or 32 months. Apparently, then, the claims of halving of computer costs per every 18 months were not based on computer prices. Similarly, the claim that computing power doubles every 18 months at a fixed price is not supported by available data.

In theory, these claims could have been based simply on semiconductor prices. In practice, however, semiconductor prices have not been following any simple trend. Using the price index for memory chips constructed by Grimm (1998), we can see that price fluctuations have been great. The dots in Figure 7 indicate averages over several years. During the 1974-1979 period, the price index dropped about 41 per cent per year. In the 1980-1985 period, the decline slowed to 37 per cent. In the 1986-1991 and 1991-1996 periods the decline was about 19 per cent.

[18] Quality-adjusted computer price indexes were included in the US producer price index on an operational basis from December 1990 (cf. Holdway, 2001). Other countries do not make similar corrections. This is one reason why the US economy has been growing faster in recent years than, for example, the European countries and Japan.

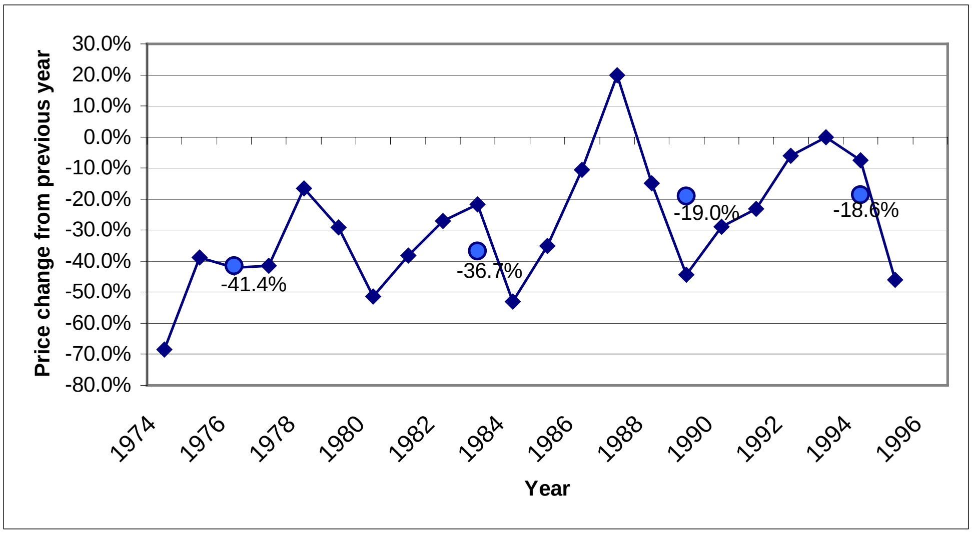

Figure 7. Percent change of price index for memory chips from previous year. (source: Grimm, 1998, Table 4)

Grimm has also calculated price indexes for microprocessors using the same methodology. For microprocessors the decline in price indexes has been considerably faster than for memory chips. During the 1985-1996 period, quality adjusted microprocessor prices dropped at an average annual rate of 35 percent. The changes, however, fluctuated strongly from year to year. In 1988, the decline was 12 percent, whereas in 1995 it was 66 percent. As there were important architectural changes in the microprocessors during these years, it is not easy to find hedonic models that incorporate the relevant characteristics.

A complete personal computer contains several different types of chips and other components, such as hard disk drives, CR-ROM drives, a keyboard, and a displays. Price changes in PC, therefore, reflect quality changes in several different types of technologies. The hedonic estimation models, however, tend to break down when new components and functionality are added to computers. When notebook computers started to become important towards the end of the 1980s, the different existing technical characteristics became revalued. For example, more memory perhaps meant progress for desktop users but for notebook users it implied shorter battery lifetime or the need to stay close to the power plug when using the computer. Indeed, it seems that from about 1987, the prices for technical characteristics started to differ between desktop and notebook computers. Moreover, the characteristics used to estimate the value of computers vary greatly from one year to another, indicating that the estimated price indexes do not capture well the user benefits (Berndt, Dulberger, & Rappaport, 2000).

Assuming that the value of different characteristics would have remained same over the 1979-1999 period, Berndt et al. estimate that the annual aggregate growth ratio has been about -25 percent. The rate of price change however has considerably varied over the years and has been speeding up. For desktop computers, from 1976 through 1989 the annual decrease was about 18 percent. It then accelerated to 34 percent in the 1989-1994 period, and further to 41 percent in the 1994-1999 boom years.

This recent growth may reflect increase in cost-adjusted performance. For example, architectural changes in microprocessors during the second half of the 1990s moved much of the external and expensive cache memory onto the processor chip. On the other hand, the rapid improvements in average PCs sold might also reflect, for example, availability of consumer credit and funding for information technology intensive firms.[19]

One should, however, note that different data collection methods and different estimation models lead to very different results. Whereas the early studies on hedonic prices gave relatively similar results when averaged over years, typically in the –20 to –30 percent AAGR range, recent research has revealed many fundamental problems in hedonic price models. This can be illustrated simply by comparing the results produced by different PC price index methods. These are depicted in Figure 8. According to some estimation methods, prices increased whereas some recorded over 30 percent decreases. Pakes (2002) discusses the reasons for these differences in great detail and proposes that from 1995 to 2000, the decline in PC prices was about 17 percent annually, varying, however, greatly from year to year.

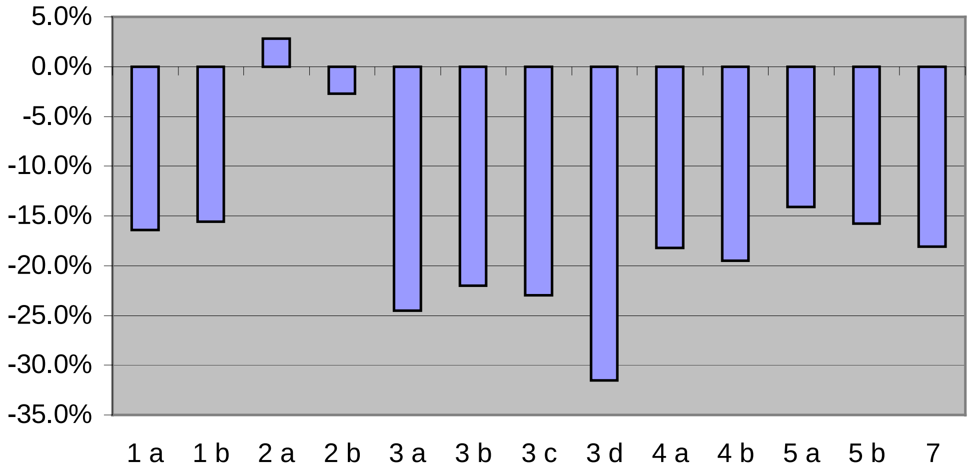

Figure 8. Average change in PC price indexes in 1995-1999 using different estimation models. (source: Pakes, 2002, Table 6)

Price changes, of course, reflect market competition, imbalances in supply and demand, technical change, and new markets that open as new uses are found for technology. These have very little to do with the original formulations of Moore's Laws. Yet, one may ask whether the current economic evidence supports the claim the cost of computing halves every 18 months. As can be seen from above, the answer is no. According to the estimates produced by Berndt et al. (2000) for desktop computers, in the 1976-1989 the average price decrease corresponded to halving of price about every 50 months. At the beginning of the 1990s it corresponded to about 28 months. During the last years of the 1990s, the prices halved in 24 months.[20] This

[19] I will discuss the alternative explanations in a short paper under preparation, "Notes on Nordhaus and the new economy."

[20] It should be noted that the main focus of Berndt et al. is to study whether the basic technical parameters used in hedonic models, i.e., hard disk memory size, processor speed, the amount of RAM memory, included CD-ROM, and manufacturer brand, are stable across time. Their result is that they

last result can be compared with the Pakes's result that quality-adjusted PC prices dropped from 1995 to 2000 annually about 17 percent, corresponding to halving in 53 months. The change in the average values as well as the yearly fluctuations show that the price decreases have not followed an exponential trend. The great differences between the results of the different modeling exercises show that we do not know how quality-adjusted computer prices should be measured of how they have changed during the last decades.

6. Computers and development

In the previous sections we have reviewed the original formulations of Moore's Law and its revisions. We found that Moore changed his interpretations of Moore's Law during the 1960s and 1970s, and that its subsequent extensions have added qualitatively new and important aspects to it. Whereas the original formulations of Moore's Law focused on counting components on integrated circuits, its extensions made claims of exponential increase in processing power and exponentially dropping quality-adjusted prices of computing. We reviewed the available empirical evidence for these different versions of Moore's Law and found that they have little empirical support. Semiconductor technology has not developed according to Moore's Law. The claims that future developments in semiconductors, computer technology, or information processing would be determined by the continuation of Moore's Law are, therefore, obviously invalid.

This result is either trivial or quite illuminating. It is trivial in the sense that no-one has seriously been arguing that Moore's Law would be a deterministic natural law. It is supposed to be a rule-of-thumb that tries to give an overall picture of the dynamics of development in semiconductor technology and, more generally, in computer technology and information society.

The result, however, is also far from trivial. It implies that references to Moore's Law qualitatively miss the character of development in semiconductor technology and the information society.

Gordon (2000) has argued that the rapid decrease in semiconductor and computer prices may reflect a rapidly decreasing marginal utility of computing technology. This is an important and interesting argument. In essence it says that very rapid advances in semiconductor technology have been necessary because without the quality improvements customers would not have been interested in investing in computing technology. Manufacturers, in other words, have had to ship continuously more complex chips with better processing capabilities at the same time continuously decreasing prices. Without huge advances in processing capability, the market for computing would have become saturated.

---

are not. In other words, a 10 MHz processor is a qualitatively different thing than a 1000 MHz processor. They also focus on the question whether the estimated parameters are different for mobile and desktop computers and find out that they are not. In other words, memory size and processor speed mean different things to laptop users and desktop users. Their analysis, however, has clear limitations. For example, parameters such as laptop weights and battery lifetimes are not included.

Here, of course, the industry dynamics play an important role. For example, computers require software. One of the important drivers for buying increasingly powerful computing equipment has been that new versions of operating system and application software have typically demanded more processing power. Although it seems clear that today personal computers are much more functional than twenty years ago, it is not clear how much more functional they are.

The regular doubling and exponential growth that underlies the different versions of Moore's Law implies a very unique claim. It fundamentally says that the described phenomenon grows purely based on its internal characteristics. Exponential growth means that after the growth speed is set, the future unfolds based on nothing but the current state of the system. Contrary to what some commentators of Moore's Law have claimed, exponential growth is not uncommon. When we put money on a fixed interest rate account, and reinvest the returns, it will grow exponentially. In this sense, a bank account is a prototype of self-determined endogenous growth. Exponential growth, however, is very uncommon in real world. It usually ends when it starts to matter.

During its history, the semiconductor industry has several times hit the speed limit. First it was bailed out by the digital clock and calculator industry, then by mini and mainframe computer industry. In the mid-1980s, just when no one seemed to be able to make a profit, the IBM PC and Microsoft saved the day. In the mid-1990s, the Internet and World Wide Web exploded the hard disk and memory market and created the need for new processor architectures that were able to handle images, sound, and video. During this time, the investment required to set up a new semiconductor factory grew roughly exponentially, reaching the US$ 2 billion limit in April 2002.[21]

It is therefore no surprise that semiconductor industry has not actually followed an exponential growth trend. Yet, the very rapid technical advance does indicate that the semiconductor industry has been astonishingly little constrained by the outside reality. One explanation for this is that it has created its own reality where old limits have not much mattered. In particular, the rapid technical advance in semiconductors has become essentially dependent on the evolution of software industry. Moore's Law therefore perhaps has created both Intel and Microsoft, which, in turn, have constantly reproduced each other.

As the size and importance of computer and information processing technologies now is becoming more than a couple of percents of national economies, it can be predicted that the endogenous growth in this industry cluster is about to end. The imbalance between supply and demand shifts and the social basis of demand makes itself increasingly visible. The open source movement, for example, effectively disconnects the economics of operating systems from the economics of semiconductor manufacturing, thus splitting the industry cluster in half.

Economists have tried to use hedonic models to link technical change and consumer's utility. Hedonic pricing models, however, have important theoretical limitations. The value of a computer changes radically, for example, when it can be connected to the

---

[21] This is commonly known as Moore's Second Law, c.f. Meieran (1998). Intel's new $2 billion chip manufacturing facility in Ireland was announced in April 2002. It is expected to start operations in 2004, cf. http://www.intel.com/pressroom/archive/releases/20020425corp_b.htm.

Internet, when relevant content becomes available on the net, and when the user gains competence in using the technology. The value of computer technology also depends on available software applications and the ways of using computers in social practices. Many of the technical advances that are modeled in hedonic price models have been consumed by increasingly complex software that is needed to operate the system, thus creating producer economic welfare instead of consumer welfare. One theoretical economic approach to address some of these issues is to extend hedonic models with "utility distributions" that reflect the different contexts of use. This, however, is theoretically problematic as the "utilities" themselves depend on technological context and advances.

As Pakes (2002:25) has noted, hedonic models do not register any gains that result from introducing goods that truly expand the range of use. In other words, although hedonic models try to connect technological advance and economics, they are blind to true innovation. They register only improvement. In the history of computing, many of the most important innovations have been based unintended uses, and made by the users. This empirical unpredictability of beneficial uses of technology means that hedonic models miss discontinuous technical change. Furthermore, they easily miss the various alternative ways technologies are used in different social contexts.

In the semiconductor industry, discontinuity and continuity have co-existed in an interesting way. At the beginning of the 1960s, Robert Noyce made at Fairchild a bold strategic move when he promised to sell integrated chips much below their current manufacturing costs, relying on the expectation of rapidly dropping costs. After Noyce and Moore left Fairchild to found Intel, this business model has became a commonly accepted axiom in the semiconductor industry. The industry has been continuously falling forward, hoping that Moore's Law will hold, economically save the day, and justify the belief in technical progress.

In reality, the belief in rapid development has often paid off. Discontinuous innovations have created new uses and markets for semiconductors and have produced an expanding market. Instead of filling a market need, the semiconductor industry has actively and aggressively created markets. At times the aggregate market has grown at a speed that has appeared to be almost infinite in relation to existing manufacturing capability.

The rapid growth of semiconductor industry, therefore, has not been driven simply by technical advance in semiconductor industry. Although the aggressive pricing policy has facilitated the wide use of semiconductors, the high demand for semiconductor technology has fundamentally reflected a continuous stream of innovations that have occurred outside the semiconductor industry. In other words, the big bang in semiconductor technology is also an illusion. It has been produced in a rapidly expanding space where the boundaries of reality have escaped fast enough to create an effective vacuum that has suck up all emerging transistors.

The need to radically drop the component and computer prices, however, also reflects the heavy competition that has resulted partly as new entrants have swarmed to the rapidly growing and exciting semiconductor and computer industries. It also indicates that the suppliers have tried to push an increasing number of chips to a market that has not always been interested in using them. The world of computing has been expanding but it has not been infinite.

As semiconductor and computer engineers have historically focused on technological features and technology has been viewed as a bottleneck, large investments have been made on research and development and supply of technology. This has accelerated the development of technology beyond the speed of development that it would have had if supply and demand had been in a better balance. Demand, however, is fundamentally about the uses and usefulness of computing. This was the fundamental problem which the integrated circuit industry faced in the 1965-1968 period, and which it solved by focusing on memory chips and microprocessors. Retrospectively, this was a good choice. It enabled the users to figure out what to do with the chips. Semiconductor industry, therefore, became a platform for innovation. The balancing of supply and demand, however, eventually requires a perspective from which we can say something about the actual benefits of computing.

Over-investment in technology development is not necessarily a bad thing. Speculative bubbles of various sorts have played an important role in restructuring economy and by creating new unforeseen and unintended opportunities. Indeed, today we have much computing and information processing capability. The question is what are we going to do with it.

Moore's Law gave us a compact and a deceptively exact way to express beliefs in technological determinism. Later it became transformed to economic determinism, which argued that people would buy computers because they will be ridiculously cheap. Moore's Law also provided a convincing basis for arguing that the development of economies and societies is at the hands of technical experts. The fact that Moore's Law has so often been misrepresented and used together with contradictory evidence indicates that it has expressed strong and fundamental convictions about the nature of progress. Contrary to what its users have often claimed to say—that the history of semiconductors and computing has followed a well-defined exponential path—the rhetoric point of Moore's Law has been directed towards the future, determined by technological development and understood by the speaker.

Gordon Moore obviously was right in predicting that the complexity of semiconductors would grow rapidly and that silicon chips would become economically and socially extremely important. Indeed, his 1965 analysis of the dynamics of integrated circuit industry contained key insights that made semiconductor industry what it is today. In a way his prediction, however, was too successful. It allowed technologists, economists, and politicians to neglect important factors that have been driving social, technical, and economic development during the last decades. Although the increasing use of computing technology has made people more aware of, for example, social, cultural, organizational, political, ethical, and cognitive issues related to information processing, physics is still commonly seen as the hard core of the future developments. As a result, many discussions on the future of Moore's Law have focused on physical limits. In recent years also economic considerations have gained legitimacy in this context, partly because Moore himself has frequently predicted that the increases in chip complexity will not be limited by physics but by the exponentially increasing costs of manufacturing plants.

As computing technology becomes increasingly pervasive, we eventually have to ask what benefits it actually brings. Fundamentally, this question can only be answered in a theoretical framework that is able to define development. In theory, there are many different ways to approach this question, both old and new. It should, however, be clear that development cannot be reduced to shrinking line-widths, maximum number of components on a chip, or minimal manufacturing costs. Indeed, one of the paradoxes of information society is that we still have very limited understanding on how to link technical advance and development. Partly this is because technical advances have simply been defined as development. This exaggerated and somewhat limited focus on technical advances has produced a wide variety of extensions to Moore's Law, eventually pushing it far beyond its original scope and available evidence. In the process, some of the kinks in the historical evolution have been eliminated and scientific facts have been created when needed. True innovation, however, is not predictable. It requires that we remember our history so that we are not doomed to repeat it.

References

AMD. (2001). Understanding Processor Performance. Advanced Micro Devices, Inc., August 24, 2001. http://athlonxp.amd.com/includes/content/whitePapers/understanding.pdf.

Berndt, E.R., Dulberger, E.R., & Rappaport, N.J. (2000). Price and quality of desktop and mobile personal computers: a quarter century of history. 17 July 2000, http://www.nber.org/~confer/2000/si2000/berndt.pdf.

Berndt, E.R., & Griliches, Z. (1990). Price Indexes for Microcomputers: An Exploratory Study. National Bureau of Economic Research. Working Paper, 3378.

Birnbaum, J., & Williams, R.S. (2000). Physics and the information revolution. *Physics Today, (January 2000),* pp.38-42.

Brenner, A.E. (2001). More on Moore's Law. *Physics Today,* **54** *(7),* pp.84-84.

Cartwright, D.W. (1986). Improved deflation of purchases of computers. *Survey of Current Business,* **66** *(3),* pp.7-9.

Chow, G.C. (1967). Technological change and the demand for computers. *American Economic Review,* **57** *(5),* pp.1117-30.

Cole, R., Chen, Y.C., Barquin-Stolleman, J.A., Dulberger, E.R., Helvacian, N., & Hodge, J.H. (1986). Quality-adjusted price indexes for computer processors and selected peripheral equipment. *Survey of Current Business,* **66** *(1),* pp.41-50.

Cringely, R.X. (1992). *Accidental Empires: How the Boys of Silicon Valley Make Their Millions, Battle Foreign Competition, and Still Can't Get a Date.* Reading, MA: Addison-Wesley.

DeLong, J.B. (2001). Do we have a "new" macroeconomy. http://www.j-bradford-delong.net/.

EC. (2000). European Competitiveness Report 2000. European Commission. SEC(2000) 1823. http://europa.eu.int/comm/enterprise/enterprise_policy/competitiveness/doc/compet_rep_2000/cr-2000_en.pdf.

Gates, B. (1997). Keynote address, May 29, 1996. In O'Reilly & Associates (Ed.), *The Internet and Society.* (pp. 27-35). Sebastopol, CA: O'Reilly & Associates, Inc.

Gelsinger, P., Gargini, P., Parker, G., & Yu, A. (1989). Microprocessors circa 2000. *IEEE Spectrum, (October 1989)* http://www.intel.com/research/silicon/ieee/ieee5.htm

Gibbs, W.W. (1997a). Gordon E. Moore: Part 1: The birth of the microprocessor. *Scientific American, (September 22, 1997)*

Gibbs, W.W. (1997b). Gordon E. Moore: Part 2: Moore's Law. *Scientific American, (September 22, 1997)*

Gibbs, W.W. (1997c). Gordon E. Moore: Part 3: Advancing the technology. *Scientific American, (September 22, 1997)*

Gordon, R.J. (1989). The postwar evolution of computer prices. In D.W. Jorgenson & R. Landau (Eds.), *Technology and Capital Formation.* (pp. 77-125). Cambridge, MA: MIT Press.

Gordon, R.J. (1990). *The Measurement of Durable Goods Prices.* Chicago, Il: University of Chicago Press.

Gordon, R.J. (2000). Does the "New Economy" measure up to the great inventions of the past? *Journal of Economic Perspectives,* **14** *(4),* pp.49-74.

Gore, A. (1999). Remarks as prepared for delivery by Vice President Al Gore, Lifelong Learning Summit, January 12, 1999. http://clinton3.nara.gov/WH/EOP/OVP/speeches/lifelong.html.

Grimm, B.T. (1998). Price indexes for selected semiconductors, 1974-96. *Survey of Current Business, (February 1998),* pp.8-24.

Holdway, M. (2001). Quality-adjusting computer prices in the producer price index: an overview. http://www.bls.gov/ppi/ppicomqa.htm.

IBM. (2001). Strained Silicon' Speeds Electrons and Rewrites Moore's Law. http://researchweb.watson.ibm.com/resources/news/20010608_strained_silicon.shtml.

ITRS. (2001). International Technology Roadmap for Semiconductors, 2001 Edition, Executive Summary. http://public.itrs.net.

Jovanovic, B., & Rousseau, P.L. (2002). Moore's Law and Learning-By-Doing. National Bureau of Economic Research. Working Paper 8762, February 2002. http://www.nber.org/papers/w8762.

Mann, C.C. (2000). The end of Moore's Law? *Technology Review Newsletter,* **1** *(2, May 23, 2000)*

Markoff, J., & Lohr, S. 2002. Intel's huge bet turns iffy. *The New York Times, September 29, 2002.*

Meieran, E.S. (1998). 21st century semiconductor manufacturing capabilities. *Intel Technology Journal, (4th quarter, 1998)* http://www.intel.com/technology/itj/q41998/articles/art_1.htm

Merton, R.K. (1973). *The Sociology of Science.* Chicago, Il: University of Chicago Press.

Merton, R.K. (1988). The Matthew Effect in science, II: Cumulative advantage and the symbolism of intellectual property. *Isis,* **79** *(299),* pp.606-23.

Merton, R.K. (1995). The Thomas Theorem and the Matthew Effect. *Social Forces,* **74** *(2),* pp.379-401.

Metcalfe, B. (2000). What will come next if silicon fails to sustain the 35-year-old Moore's Law? *InfoWorld, (June 16, 2000)*

Moore, G.E. (1965). Cramming more components onto integrated circuits. *Electronics,* **38** *(8)*

Moore, G.E. (1975). Progress in digital integrated electronics. *IEEE, International Electron Devices Meeting, IEDM Tech.Digest* **-** *1975,* pp.11-13.

Moore, G.E. (1979). VLSI: some fundamental challenges. *IEEE Spectrum, (April 1979),* pp.30-37.

Moore, G.E. (1997). An update on Moore's Law. Intel Developer Forum Keynote, September 30, 1997. http://www.intel.com.

Moore, G.E. (1999). A pioneer looks back at semiconductors. *IEEE Design & Test of Computers,* **16** *(2),* pp.8-14.

Moore, G.E., & Davis, K. (2000). Learning the Silicon Valley way. Prepared for the GREEG Conference "Silicon Valley and Its Imitators" July 28, 2000.

Mowery, D.C., & N. Rosenberg. (1998). *Paths of Innovation: Technological Change in 20th-Century America.* Cambridge: Cambridge University Press.

Pakes, A. (2002). A reconsideration of hedonic price indices with an application to PC's. National Bureau of Economic Research. Working Paper 8715, January 2002. http://www.nber.org/papers/w8715.

Raymond, E.R. (1999). Jargon File. http://www.tuxedo.org/~esr/jargon/html/entry/Moore's-Law.html.

Schaller, B. (1996). The origin, nature, and implications of "Moore's Law". http://mason.gmu.edu/~rschalle/moorelaw.html.

Schmitt, L. (2000). An interview with Gordon Moore. *Ingenuity: University of Illinois at Urbana-Champaign Electrical & Computer Engineering Department*, **5** *(2, May 2000)* http://www.ece.uiuc.edu/ingenuity/500/mooreint.html

Stigler, S.M. (1999). *Statistics on the Table: The History of Statistical Concepts and Methods.* Cambridge, MA: Harvard University Press.

Talero, E., & Gaudette, P. (1996). Harnessing Information for Development - A Proposal for a World Bank Group Strategy. The World Bank. http://www.worldbank.org/html/fpd/harnessing/.

Triplett, J.E. (1989). Price and technological change in a capital good: a survey on research on computers. In D.W. Jorgenson & R. Landau (Eds.), *Technology and Capital Formation.* (pp. 127-213). Cambridge, MA: MIT Press.

Yang, D.J. (2000). On Moore's Law and fishing: Gordon Moore speaks out. *U.S.News Online, (7/10/00)* http://www.usnews.com/usnews/transcripts/moore.htm