An Alternative Methodology: Valuing Quality Change for Microprocessors in the PPI

Mike Holdway
Bureau of Labor Statistics
Division of Industrial Prices and Price Indexes
Revised 1-01

An Alternative Methodology: Valuing Quality Change for Microprocessors in the PPI

Mike Holdway

Identifying the Target

Identifying and valuing quality change for the rapidly evolving microprocessor (mpu) requires several preparatory steps. Obviously we must obtain an understanding of the basic functional features that mpus are designed to provide and the manufacturing processes used . But we must first insure that our net is cast over the entire mpu market which is more broadly inclusive than the casual observer might suspect.

If you ask most people to define or describe a mpu they are likely to employ restrictive computer centric terms such as CPU, computer chip or even computer "brain". Many computer manufacturers imply an even more restrictive definition when they use the phrase "Intel Inside"[1] as an integrated and highly visible part of their marketing strategies. The popular association of mpus and Intel Corporation is not surprising because Intel enjoys a dominant position as a producer. Various trade publications, such as *Microprocessor Report* and *Electronic News*, estimate Intel's 1999 share of the computer mpu (cmpu) market at around 80 percent in terms of units shipped and 90 percent in terms of revenue. Visibility and marketing aside, cmpus are only a subset of the total mpu market.

Another type of mpu designed for non-computer applications is called the embedded[2] processor (empu). Empus are found in cellular and digital phones, motor vehicles, air transportation, communication networks, electronic organizers, pagers, printers, game consoles (i.e. Sega and Nintendo) and thousands of other consumer and industrial products. Empus account for approximately 98 percent of worldwide processor shipments[3], but less than 20 percent of revenue because of much lower average selling prices relative to cmpus. Perhaps because of their lack of identification with a highly visible company, empus not only have little recognition in the general public but have been ignored by much of the economic literature that has attempted to provide quantitative or technical insight to the general mpu market. In terms of economic measures such as price indexes, GDP and productivity, it is important that neither mpu subset is used as a proxy for the overall market due to significant differences in technology and price trend. Because embedded designs are often application or customer specific, they tend to have longer life cycles relative to cmpus which reduce product substitutions and the corresponding need to value quality change. In contrast, cmpus have exceptionally short life cycles which require the PPI to make almost

---

[1] Intel Corporation subsidizes the cost of computer (PC) manufacturers advertisements if they include this slogan and Intel's logo in their marketing materials. Intel has final approval of proposed OEM advertising for which they fund several hundred million dollars annually (approximately 3% of revenues). The subsidies began in 1991 and by 1994 "almost the entire PC industry had joined in". See *Intel Inside*, Jackson, T, Penguin Putnam, 1997, pg. 315.

[2] Trade journals such as *Electronic Buyers News* (www.ebn.com) often describe empus as any non-computer processor. This description is essentially correct though there are a few mpus that are sold into both embedded and computer markets.

[3] Approximately 140 million cmpus shipped last year vs almost 5 billion empus. "Guest Viewpoint: Embedded Systems and the Microprocessor", Microprocessor Report, 4-24-2000, http://www.mdronline.com/mpr/h/2000/0424/141702.html

continual estimations of quality change. Thus, the description of a targeted quality adjustment and product substitution strategy will be limited to cmpus.

A Brief Retrospect of PPI CMPU Coverage

Prior to 1997 the PPI's Microprocessor index (code 36741A201) was moved primarily by empus and small niche players in the cmpu market. The disparity between the PPI sample and mpu universe was due in part to significant non-response from a major segment of the cmpu market. As previously mentioned, the technology and price trends of empus and cmpus are dissimilar. To the extent that the PPI sample was not representative of the market, significant bias may have been introduced. The bias became evident over time as the PPI's Microprocessor index was correctly (in hind-sight) criticized by the Bureau of Economic Analysis[4] (BEA), the Federal Reserve, and economic consultants from the private sector. After careful review the PPI introduced secondary source[5] pricing data for cmpus on January 1997. The introduction of secondary source prices can be viewed as a limited supplemental sample designed to represent 85 percent of the cmpu market that was not available for direct repricing in the PPI. Chart 1 shows, in log form, the change in PPI's Microprocessor index starting from the last sample (1993) through January 2000 in order to present a kind of before and after.

Chart 1.
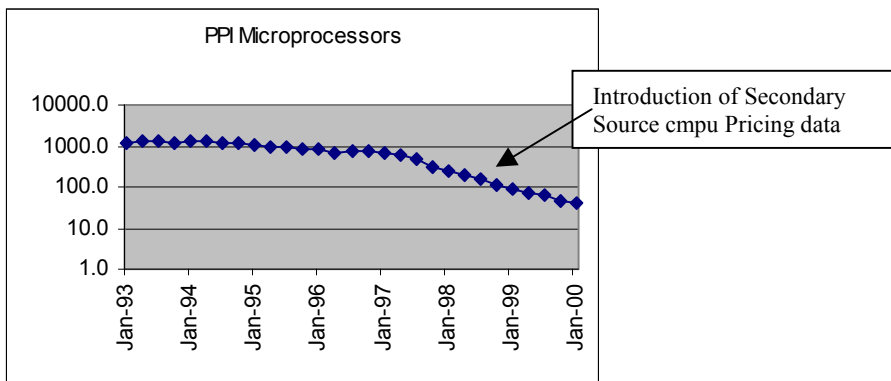
The significance of augmenting the PPI with secondary data to enhance its coverage of cmpu pricing trends is apparent.

As mentioned in the opening section, the first step in our inquiry was to identify the target. This step is largely complete, but at this point we are no closer to a technical definition of cmpus than when we started. A basic understanding of cmpu architecture is required if the PPI is to effectively address the issue of calculating

---

[4] Bruce Grimm, a research economist with the BEA, sent a paper to the PPI for comment in 1996 titled "Quality Adjusted Price Indexes for Metal Oxide Semiconductor Microprocessor Integrated Circuits". He uses secondary source data to construct a summary chained Fisher index for cmpus from 1985 to 1994. The Fisher cmpu index had an AGR of -35.3 percent for the 84-96 time frame (95-96 were extrapolated). The AGR for the PPI Microprocessor index in the same period was -2.9 percent. The difference in index formula (Fisher vs Laspeyres) can not explain the significant difference in AGRs. Ignoring index formulas and the empu segment (omitted from the BEA index), the most obvious difference is that BEA was able to purchase cmpu pricing data that was missing in the PPI.
[5] Pricing data for the major X86 cmpus are obtained from *Microprocessor Report* and *Electronic News* in 1,000 lot order sizes. Both publications are used as a cross-check for accuracy.

valuations for rapid technological change. For now, we can think of cmpus as semiconductor devices that are designed to process or execute instructions. They require power, memory and communication channels (input-output) to operate.

Then and Now

Intel invented the first mpu, the 4004, in 1971 which they initially marketed as a *microprogrammable computer on a chip* to computer hobbyists for around $200.   Within two years, competitive pressure and faster processors forced the price of the 4004 below $100. Compared to modern cmpus, the 4004 was a crude device with 4 bit internal data paths, 2,300 transistors and an operating speed of 108Khz[6]. Intel introduced its latest generation cmpu, the Pentium III, in 1999. This mainstream device includes 64 bit internal data paths, 28 million transistors, and a current maximum speed of 1,000Mhz. The Pentium III has additional architectural features that are far beyond the capabilities of 1971 technology. These features include but are not limited to L1/L2 caches, register renaming, out of order execution, multiple instruction units, pipelining, and single instruction-multiple execution.

The massive technological and performance differences between current generation cmpus and Intel's original device occurred as a progression of advances over time. Our initial view of this progression is best served with a simplified approach that limits technological comparisons to changes in the number of cmpu transistors used to manipulate data.

The internal structure of the cmpu is composed of multiple functional units that are made up of logic gates (if 1 do this; if 0 do that) which are in turn built with transistors. Transistors function as electronic switches in the sense they either allow the passage of electrons through a circuit, thereby signaling an ON condition, or block the flow of electrons signaling an OFF condition. This switching capability is key to a cmpu's ability to function because it is a binary device that can only accumulate and string together a series of 1s and 0s to represent words and numbers which in turn can be used as data or instructions. The ability of a cmpu to perform useful work by executing instructions efficiently has a great deal to do with the design and layout, also referred to as the architecture of the device. In other words, if both manufacturer A and B produce competing cmpu products with a similar transistor count[7], manufacturer A's device may outperform manufacturer B's due to a more efficient architecture. For this reason, and others, transistor counts should not be viewed as an absolute technology metric, but are useful as a general indicator of relative improvements in design as well as production processes. Table 1 shows the amazing increase (+12,172%) in the number of transistors that Intel has been able to design into their major cmpu products since 1971.

---

[6] *Microprocessor Timeline 1971-1976;* http://www.islandnet.com/~kpolsson/comphist/comp1971.htm
[7] And also use similar production technologies that yield comparable feature size.

Table 1*

MPU Product	Year of Introduction	# of Transistors	% Change
Intel 4004	1971	2,300	
Intel 8008	1972	3,500	52%
Intel 8080	1974	4,000	13%
Intel 8086	1978	29,000	625%
Intel 80286	1982	134,000	362%
Intel 80386	1985	275,000	105%
Intel 80486	1989	1,200,000	336%
Intel Pentium	1993	3,100,000	158%
Intel Pentium Pro	1995	5,500,000	77%
Intel Pentium II	1997	7,500,000	36%
Intel Pentium III	1999	28,000,000	273%

*From www.intel.com/intel/museum/25anniv/hof/tpecs.html.

The rapid growth in transistors that are packed within a tiny slice of silicon is one of many possible measures of technological advances that have occurred over time. It is the magnitude of technical change that is of interest because if unaccounted for in a price index, the index becomes unavoidably biased.

The PPI did not publish a Microprocessor index in 1971, but we can hypothesize an extreme example and ask how the PPI would go about constructing a index (T=2000; T-1=1971) while adjusting price change for the massive technological differences between the 4004 and a current generation cmpu. The Pentium III has a transistor count more than 12,000 times that of the 4004 and a Mhz rating that is more than 10,000 times faster. In terms of transistor density, the 4004 had 2,300 transistor in a 12mm^2 chip or 191 transistors per mm^2 of silicon. The Pentium III packs 28 million transistors into a 106mm^2 chip which is equivalent to 264,151 transistors per mm^2 of silicon. These are a few of the characteristics (out of hundreds) that partially define cmpu technical advances. Conventional PPI quality valuation methodology would use a comparison of production cost differences between the 4004 and Pentium III. This procedure (if relevant data could be obtained from manufacturers) implicitly assumes that a significant quality increase is accompanied by higher resource/input cost. To correctly apply changes in resource cost to value changes in cmpu quality requires that the PPI answer the question; What is the input cost difference between a obsolete cmpu and its replacement using the production function of the obsolete (Laspeyres) product? The relevant production function is reversed in a Paasche. However, one of the consequences of Moore's Law[8], is that cmpus not only get faster and better over time, but also cheaper. This is a powerful insight that is made possible by improvements in chip fabrication technologies that allow more transistors to be packed on smaller less costly chips. It is this latter point, lower unit cost that is related to technological improvements, that make a direct comparison of resource costs problematic. New input technologies that

---

[8] Gordon Moore, cofounder of Intel Corp., postulated in 1965 that that the logic density of silicon transistors doubled every year; he later changed this to 18 months, and the figure still applies today. Moore's Law can be graphed as an exponential curve; although it starts slowly, the pace of growth accelerates as time passes. Advances in cmpu input technologies have enabled producers to transform roomfuls of vacuum tubes into chips the size of a fingernail.

significantly change cmpu input requirements reflect a shift in the production function, not a movement along the function available in the reference period. Despite the difficult measurement problem caused by non-comparable production functions, it is important to keep the resource cost issue in perspective. Rapid technological change that simultaneously reduce unit input cost while substantially increasing output quality does NOT invalidate the resource cost approach in a output price index. While the PPI continues to emphasize the resource cost criteria for valuing quality change, cmpus and other high-tech products often require information that is not available in a real-time environment.

Triplett (1983) presents a more lucid description of the resource cost measurement problem using the example of rapid advances in computers that also involve a shift in the production function. He supports the correctness of the resource cost approach, but states that the *requirements function* is unknown. *That is, there may be data on the cost of the old machine under the old technology and the cost of the new machine under the new technology, but no data at all on the cost of both under comparable technologies*. In fact, the PPI usually has no timely information on cmpu input costs for either the old or new technology. Triplett proposes taking the analysis of quality change into characteristics space (hedonics) as one way of addressing the practical limitations of the resource cost criteria. However, the PPI's experience with hedonics[9] has shown that this approach may not provide a viable alternative for some high tech goods. Hedonics, as applied to cmpus, will be discussed in a later section of this paper.

CMPU Quality Improvement and Reduced Unit Input Cost

A quick review of fundamental cmpu production processes will help explain how cmpu unit production cost can fall as technical features improve. The following description is oversimplified, so I highly recommend the book "Microchip Fabrication", 3rd Edition, by Peter Van Zant for a more complete description.

The production process begins with the basic material, refined silicon, which is implanted with precise amounts of impurities[10]. The introduction of impurities, also called doping, give the silicon desirable properties such an excess or deficiency of electrons. It is the flow of electrons that are controlled by transistors that enable the ON/OFF states recognized and manipulated by the cmpu. Silicon wafers enter a complex production process enabled by some of the most advanced equipment in the world, such as ion implanters and photolithography projectors that are so precise that the wavelength of ordinary light is relatively crude. Photolithography equipment uses an intense light source to project multiple images of integrated circuits onto the silicon wafer. The image patterns are captured by a special film called a resist that has been applied to the wafer. The pattern is transferred from the resist to the wafer by an acid etching process that physically imprints the precise outline of the circuit design into the wafer surface. Current technology enables production quantities (tens of millions) of cmpus with circuit features of 0.18 micron or 1/600th the thickness of a human hair. These features can only be viewed through a powerful microscope and their ability to function correctly are due as much to physicists and materials scientists as

---

[9] The PPI has developed and applies quality valuations from hedonic models for several types of computers, printers, and storage devices. See Holdway (1999b).

to engineers and technicians. It is the rapid advances in production technology that enable discrete feature sizes that are beginning to approach the atomic level. In the 1980s many industry observers felt that the laws of physics would not allow semiconductor technology to break the 1.0 micron barrier. Since then the industry has introduced 0.80, 0.50, 0.35, 0.25 and 0.18 micron devices and is currently shifting to 0.13 micron designs. Each shift to a new micron generation allows more features, such as transistors, to be packed into a smaller section of a silicon wafer. As features are packed more tightly together the distance that electrons must travel is reduced, which improves performance.

The production process can also be viewed through analogy. Stencils are widely used to transfer a painted pattern or design onto a surface. In a similar fashion, the electronic circuit pattern is printed on a quartz/chrome mask that functions like a stencil but allows light rather than paint to transfer the circuit design onto the silicon wafer.

As the industry pursues lower cost and better performance, circuit designs have advanced to the point that the wave-height (amplitude) of most common light sources are too large to pass cleanly through the mask. This limitation was supposed to have been one of the laws of physics that would keep producers from breaking the 1.0 micron barrier. With billions of dollars at stake, manufacturers of semiconductor production equipment introduced photolithography designs that utilize a 248 nanometer krypton-fluoride (KrF) excimer light source. Krf's amplitude is short enough to project circuit designs through masks that are designed for 0.18 micron feature sizes. Equipment is currently being installed in domestic semiconductor fabs that use 193 nm argon fluoride (ArF)[11] that will enable a further shrink to 0.13 micron. When demand for circuit designs drop below 0.13 micron, prototypes of 157 nm molecular fluoride are already in the development stage that will eventually enable the production of cmpus several times faster than current designs, while simultaneously reducing unit cost.

Rather than get bogged down in the unique complexities of semiconductor production processes, I have attempted a couple of crude drawings (figures 1 and 2) that illustrate the general effects of industry shifts to more advanced micron technologies.

---

[10] Primarily phosphorous or boron.
[11] *ASML, SVG Shoot-out for 157nm IP*, Electronic News, 7-12-99, pg. 76.
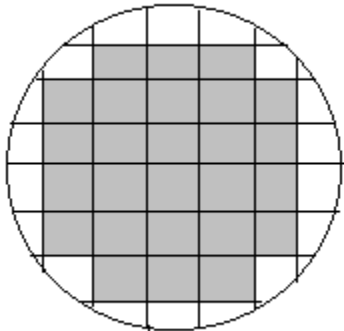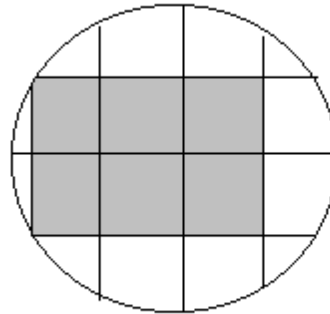
Figure 1 (0.25 micron design)　　　　　　**Figure 2 (0.35 micron design)**

Figures 1 and 2 represent 0.25 and 0.35 micron technology respectively. Wafer diameter is 8" in both processes, but it is clear that the 0.25 micron technology yields more usable cmpu dies (chips) than the 0.35 micron technology. Usable dies[12] are outlined in gray. Notice that the rectangular dies that are on the outer perimeter of the wafer are not fully formed and represent waste. The number of usable die shown in Figures 1 and 2 are for illustrative purposes and do not represent the actual yield improvement made possible by a shift from 0.35 to 0.25 micron technology.

Further details will be provided shortly, but for now let's assume that a producer is manufacturing 100Mhz cmpus using a 0.35 micron technology which yields 6 usable chips as shown in Figure 2. We also assume that each 8" wafer in figure 2 costs the producer $1,000 including an amortization factor that accounts for plant facilities and all capital equipment used in the production process. Under this scenario each 100Mhz cmpu costs $166.66 ($1,000/6). Now suppose that the producer can modify or introduce new production equipment which enables a shift to a 0.25 micron technology as shown in Figure 1. The new technology yields 26 usable cmpus but the cost of each 8" silicon wafer increases from $1,000 to $1,500 because of the investment in new capital equipment required to implement the 0.25 micron process. Each cmpu produced at 0.25 microns cost the producer $57.69 ($1,500/26), a substantial reduction in unit cost despite the increased cost per wafer. In addition, the 0.25 micron cmpus operate at 133Mhz due to the shorter distance that electrons must travel in the smaller device. The producer has not only lowered unit cost, but now has a faster (better) product to offer buyers. A few liberties were taken with this example to help illustrate one of the fundamental enablers of Moore's Law.

---

[12] I am ignoring defects such as those caused by dust or other impurities that can destroy the tiny circuit connections within the chip.

MicroDesign Resources presents a more elegant and precise description of the impact of shrinking feature size on production cost in their semiannual review of Intel's manufacturing process which is called *Intel Microprocessor Forecast, Product Roadmap, Volumes, Costs, & Prices* (IMF). This resource will be cited frequently because it not only focuses on the most important cmpu producer in the world, but also because the authors (Linley Gwennap and Mel Thompson[13]) have worked for many years in the semiconductor industry as microprocessor designers and product managers. Each edition of IMF devotes a chapter to an analysis of Intel's production costs that help to relate the concepts described above to one of the most important economic forces in the semiconductor industry. For example, table 2 describes the evolution of the Pentium based on data provided by the 3rd Edition (1998) of IMF, tables 5-3 and 5-4 pgs 55 and 57.

Table 2

MPU/Micron	Wafer Cost	Gross Die per Wafer	Net Die Per Wafer	Die Cost	Package Cost*	Total Chip Cost
Pentium/0.80	**$1,900**	80	**24**	**$80**	$31	**$111**
Pentium/0.50	**$2,400**	175	**89**	**$27**	$25	**$52**
Pentium/0.35	**$3,000**	298	**194**	**$15**	$25	**$40**

*Includes assembly and test [The production process through Die Cost is normally handled at one of Intel's domestic Fabs (in 1997), but packaging and test is handled at offshore Intel owned facilities primarily located in Malaysia and the Philippines. Offshore packaging and test is not an Intel specific practice but is used by many semiconductor producers to reduce cost.]

Gwennap and Thompson estimate wafer costs[14] by assuming full utilization (a sound assumption for Intel) and depreciate the cost of the fab and production equipment using the 4-year straight line method. The first iteration of the Pentium was introduced in 1993 using a 0.80 micron technology yielding 24 usable die per 8" wafer. Intel shifted Pentium production to 0.50 micron in 1994 which more than tripled the number of good die per wafer to 89 and then to 0.35 micron in 1995/96 which more than doubled good die to 194.

Note that the ratio of net (good) die to gross die per wafer increases at each process shrink. At 0.80 micron, only 30 percent (24/80) of the gross chips were usable, but at 0.35 micron the percent of usable chips jumped to 65 percent (194/298). There are several technical reasons why process shrinks improve the net to gross ratio but one of the most often discussed is defect density. Manufacturers spend millions of dollars to construct clean rooms to help minimize defects in their expensive wafers. These defects are often caused by tiny particles (even dust) that contaminate circuit pathways or features. A dust particle that may seem otherwise insignificant takes on the destructive proportions of a large boulder in the tiny sub-micron world of cmpus and other semiconductor

---

[13] Both authors are regular speakers at the annual Microprocessor Forum attended by most of the mpu producers including Intel, AMD, Texas Instruments, Cyrix, Hitachi, NEC and MIPS.

[14] Clean or unprocessed wafers (8") that exclude amortization of fab and production equipment, cost less than $200. It is the investment in the fab and production equipment that now routinely exceed $1 billion that drive wafer costs to the levels shown in Table 2. Note that the estimated wafer costs do not include the upfront cost of designing a new microprocessor or developing a new IC process. *IMF* estimates these upfront costs at *$200 million or more*, but because of *Intel's huge volumes* they would add *only a few dollars to the unit cost.*

products. Suppose a manufacturer is able to control the number of fatal occurrences of contamination so that they average no more than 10 per wafer. If the manufacturer can shift production to a new generation technology that enables the number of gross die to increase from 100 to 200, then the defect ratio will on average drop from 1 in 10 to 1 in 20. The higher gross number of die made possible by shifting to a more advanced technology, is further enhanced by lower defect densities. In other words, the ratio of net (marketable) Pentiums to gross Pentiums increase as microns decrease.

Drawing from the far right column in table 2, the shift from 0.80 to 0.50 micron reduced unit cost from $111 to $52 or 53 percent. Unit costs were reduced again with the shift to 0.35 micron, dropping from $52 to $40 or 23 percent. These unit cost reductions were accompanied by equally impressive quality improvements in cmpu performance. The 0.80 micron process yielded 60 and 66Mhz Pentiums, which were accelerated to 75, 90 and 100Mhz with the shift to 0.50 micron. Introduction of 0.35 micron technology enabled a jump to 120 and ultimately to 200Mhz[15]. The rapid transitions from 0.80 to 0.50 to 0.35 micron production technologies occurred over a brief 3 year period (1993-1995) and the pace continues to quicken.

Sematech, an industry research organization has recently changed it's highly regarded roadmap that presents a timetable for the introduction of new process technologies used for planning purposes by semiconductor equipment OEMs. The revised timetable was characterized in Semiconductor Business News, 6-9-99, with the observation that; *In the late 1990s, the pace of process shrinks has accelerated from three-year cycles to two years or less…*[16]. Electronic Buyers News (www.ebonline.com) reported on Intel's comments concerning their next process shrink with the statement, *Intel expects to achieve another 30% reduction in costs from the increased yields in the die shrink by moving to the 0.13- micron generation.*

CMPU Quality Change: Three Paths

At this point, the reader may understandably, but incorrectly, conclude that unit costs for cmpus are inevitably reduced over time. As long as we refer to the same cmpu generation and investments in more advanced production equipment continue to provide positive returns, speeds will increase and unit costs should decline. However, unit costs are likely to initially increase when new generation cmpus are introduced, and then

---

[15] Multiple speed ratings produced within a single micron process are a technical issue based partly on the amount of heat generated by the cmpu at a specified speed. Thermal tolerances are established by engineers to ensure product reliability. Tolerances must take into account several factors including voltage, resistance, and projected speed ratings because, in a given process, an increase in any of the three will increase heat. The photolithography process, while extremely accurate by most measures, does not transfer circuit designs to silicon with zero variance. To the extent that a slight imperfection/irregularity is created in a circuit path, additional resistance to the flow of electrons may be introduced. This additional resistance creates more heat at a given operating speed. Therefore in the 0.80 micron process, tiny imperfections may cause one cmpu to reach its thermal limit at 60Mhz, while an otherwise identical cmpu is able to remain within tolerance at 66Mhz.
[16] Robert Lineback, Semiconductor Business News, as reported by Electronic Buyers News Online- www.ebonline.com/story/OEG19990609S0007.

resume a downward trend. Additional cost may also be incurred when an existing generation is modified as was the case when the Pentium was redesigned in 1996 to include new instructions that improved integer performance. The redesigned Pentium was called the Pentium MMX and because of its larger L1 cache (L1 will be described later), was physically larger and more costly than the Pentium. Intel attempted to compensate for the size differential by shifting from a 0.35 to a 0.28 micron process which was only partially successful.

The MMX represented more of a design tweak, while the Pentium II (introduced in 1997) is an example of a new generation product. The Pentium II was initially produced in the same 0.28 micron process used for the MMX, but the Pentium II had so many additional features that it required 7.5 million transistors compared to the 5.5 million in the MMX. Without the benefit of a more advanced input technology, the Pentium II was physically larger than the Pentium/MMX which resulted in less good die per wafer and higher unit production cost. Table 3 extends the data provided in table 2 to include the MMX and Pentium II.

Table 3

MPU/Micron	Wafer Cost	Gross Die per Wafer	Net Die per Wafer	Die Cost	Package Cost*	Total Chip Cost
Pentium/0.80	**$1,900**	80	**24**	**$80**	$31	**$111**
Pentium/0.50	**$2,400**	175	**89**	**$27**	$25	**$52**
Pentium/0.35	**$3,000**	298	**194**	**$15**	$25	**$40**
Pentium MMX/0.28	**$2,800**	206	**125**	**$22**	$21	**$43**
Pentium II/0.28*	**$2,800**	123	**58**	**$49**	$31	**$80**

The Pentium II was introduced with a new package called slot 1 which included a 512K L2 Cache. This additional feature (not available with the Pentium/MMX) has been excluded to keep the cost estimates reasonably comparable. Pentiums were also designed to work with L2 caches, but they were normally provided as a separate device (and cost) by the motherboard manufacturer.

Note that Gwennap and Thompson estimate total amortized wafer cost for the 0.28 micron MMX and Pentium II dropped relative to the predecessor 0.35 micron wafer cost. The reduced wafer cost estimate is due to Intel's decision to refit existing 0.35 micron production facilities to make them 0.28 micron capable. By modifying existing facilities and equipment that were already partially amortized Intel was able to reduce wafer cost relative to the construction of a new production facility. Making broad generalizations for an industry as complex as semiconductors is probably ill-advised and can easily lead to embarrassing contradictions. Nevertheless, wafer costs should increase as multigenerational production shifts occur, but may actually decrease when existing facilities are modified to implement a single generation shift.

Returning to the example of the MMX and Pentium II, while the shift from 0.35 to 0.28 micron reduced wafer cost, unit cost is actually higher for MMX and Pentium II because of their relative bulk. The reduced yield that is directly caused by the physically larger cmpus is pinpointed by Gwennap and Thompson in table 4 through destructive analysis (opening the package that seals and protects the cmpu).

Table 4

MPU/Production Process	Die Size
Pentium @ 0.80 micron	294mm^2
Pentium @ 0.50 micron	148mm^2
Pentium @ 0.35 micron	91mm^2
Pentium MMX @0.28 micron	128mm^2
Pentium II @ 0.28 micron	203mm^2

The data presented in table 4 can be viewed as a kind of forensic trail that enables us to better understand how input quantities and costs are affected by the three primary types of cmpu quality change. These changes are identified as Types 1, 2 and 3.

Type 1: The first three Pentium entries in table 4 are examples of type 1 quality changes. As the input technology advances from 0.80 to 0.50 and finally to 0.35 micron, the physical size of the chip is progressively reduced which directly lowers cost and improves quality. Type 1 changes refer to input quantity (technology) changes that generally result in lower unit cost and better quality, in this case speed. Type 1 changes DO NOT include changes to an MPU's architecture or design[17].

Type 2: The fourth entry is the Pentium MMX which represents a type 2 quality change. The Pentium's architecture was altered with MMX in two important ways. First, 57 new instructions were added that were specific to multimedia integer performance. The second change doubled the L1 cache size. This latter change was the primary reason that the number of transistors jumped 36 percent relative to the Pentium (3.3 vs 5.5 million). A type 2 quality change includes architectural modifications that do not involve a redesign of core logic functions. In other words, type 2 changes DO NOT include modifications that define a new generation cmpu. Marginal changes to input quantities under the type 2 scenario do not have a fixed impact on unit cost. Unit costs, depending on the nature of the design change and the production process used, can decline, increase or, though unlikely, remain constant.

Type 3: The Pentium II entry is an example of a type 3 quality change. Architectural differences between the Pentium II and the Pentium/MMX are so significant that they represent a new class of cmpu. Some of the most important distinguishing characteristics of the Pentium II architecture are described below.

- Dynamic Execution Technology: Enables out of order and speculative execution. Dynamic execution is designed to minimize linear constraints that are inevitable in software programs. This new technology enables up to four instructions per clock cycle compared to the Pentium's two instructions per cycle.

---

[17] The Pentium underwent some minor peripheral changes, the most notable a reduction in voltage from its initial 5 volts to 3.3 volts. The voltage reduction was made possible by smaller feature sizes and should be considered a quality improvement due to improved energy efficiency and lower heat dissipation. This type of quality improvement is similar to the improvement in speed that is, in effect, a byproduct of the input quantity change that reduced the cmpus physical dimensions. Several other improvements were made that had the effect of removing certain latencies in the Pentium's circuit design. These improvements were of a highly technical nature and beyond the scope of this paper. (See Intel Technology Journal 3Q97 for details).

- Superpipelining: The Pentium II's pipeline (think of pipelines as an instruction queue) extends to 12 stages compared to the Pentium's relatively small 5 stage pipeline. The practical consequence of deeper pipelines is the ability to achieve higher clock speeds using the same manufacturing process. Intel claims that the Pentium II's pipeline enables up to a 50 percent Mhz increase relative to the Pentium, even if the same micron production process is used for both cmpus.
- Dual Independent Bus Architecture: Essentially provides a dedicated bus to the L2 cache. This enables a much greater cache throughput. For example the Pentium/MMX was limited to a 66Mhz shared L2/system bus, while the Pentium II's dedicated L2 bus could be clocked at half the MPU speed (i.e. a 300Mhz Pentium II's L2 cache runs at 150Mhz).

These architectural improvements are only a part of a more comprehensive list described by Intel in their *Pentium II Processor Performance Brief*, January 1998 (order # 243336-004). A type 3 quality change always includes fundamental changes in core logic design. The new mix of input quantities usually result in higher unit cost relative to the previous generation cmpu. Note that the higher unit cost of type 3 change tends to be quickly offset with more advanced input technologies as a series of type 1 changes progressively shrink the physical dimensions of the new generation cmpu. For example, Intel introduced the Pentium II at 0.28 microns in May 1997, but then shifted to a 0.25 micron production process eight months later. The new process decreased unit cost[18] 30 percent from $80 to $56. Another benefit of this type 1 quality change was that the Pentium II's maximum speed increased from 300Mhz to 450Mhz.

The three types of quality change described are the most common examples that are likely to be encountered in a price index covering cmpus. However, before proceeding to the next section, additional context is needed for type 2 changes. Due to industry hype and America's love affair with things that are new and improved it is easy to fall into the trap of assuming that cmpus only improve over time. This assumption, if blindly adhered to, can distort perceptions about how technology is wielded by producers to protect or expand market share. For example, speed or Mhz ratings continue to rapidly increase and if used as a quality benchmark would imply an unbroken upward quality trend. One could ask the question, if other metrics besides the superficial speed ratings were used, would the quality trend maintain this unbroken upward path?

To illustrate, we need only review Intel's Pentium II product lineup. The Pentium II was introduced at 233 and 266Mhz in May 1997 at $636 and $775 respectively[19]. By May 1998, the prices for these cmpus had fallen to $198 and $246, but Intel's Pentium II offerings had expanded to include additional speed ratings shown in table 5.

Table 5

---

[18] The 0.25 micron process increased the yield or net die per wafer from 58 to 120. Net die increased more than 100 percent, but total amortized wafer cost for the new process increased only 21 percent (from $2,800 to $3,400). It is no wonder that Intel plowed back over $4 billion in capital expenditures in 1997 and $5 billion in 1998 to convert to the 0.25 micron process. *IMF*, tbl. 5-4, pg. 57.
[19] *IMF*, tbl. A-3, pg. 114.

0.28 Micron PII-233	0.28 Micron PII-266	**0.28 Micron CPII-266**	0.28 Micron PII-300	0.28 Micron PII-333
$198	$246	**$155**	$375	$492

Prices are based on 1,000 lot quantities effective 5-98.

The most interesting entry in table 5 is the CPII-266Mhz that Intel introduced on 5-98 at $155. All cmpus shown were produced in the same 0.28 micron process and therefore have approximately the same die cost. The CPII is actually a new version of the Pentium II that has been stripped of all L2 cache to reduce package cost (a type 2 quality change). Intel called this modified Pentium II the "Celeron" and introduced it to maintain market share in the low-margin but high volume segment[20] of the PC market. To the unwary buyer, the Celeron ran at a respectable 266Mhz. However, with no L2 cache, the Celeron was widely reported in various industry journals such as PC Magazine to have lower performance than the PREVIOUS generation 233 Mhz Pentium MMX. By Intel's own admission, the elimination of the L2 cache in the 266Mhz Celeron reduced performance by 30 percent[21] relative to the 266Mhz Pentium II with an L2 cache.

A more complex example of a quality decline can be observed with Intel's 486 cmpu family. One of the big improvements in the 486 relative to its predecessor product, the 386, was the addition of a floating point unit (FPU). FPUs enable much faster calculations of formulas that involve fractional numbers which are commonplace in engineering, drawing, statistical and multimedia applications. Prior to the 486, if a consumer needed floating point performance they had to buy a separate specialized chip in addition to the cmpu. Intel produced two versions of the 486, the DX and the SX. The DX was introduced in 1989 at more than $900 which limited sales to the relatively low volume high performance segment of the PC market. In order to penetrate the high volume low end of the PC market Intel needed to reduce prices while simultaneously displacing its previous generation 386[22]. So, in 1991 Intel introduced the 486 SX. The 486 DX and SX were identical, except that Intel disabled[23] the FPU in the SX to rationalize a lower price. The SX enabled Intel to offer a product to the low end market without cannibalizing high margin DX sales. Since the SX was really a DX with the FPU disabled at the factory, accounting for quality change based on a change in input quantities is problematic. *IMF* estimates that $1.5 billion of SX chips were sold between 1Q93 and 2Q95. Unfortunately their revenue data does not capture pre-1993 sales so much of the total SX revenue is missing. Even though revenues are incomplete, it is clear that despite reduced capability the SX was a successful product.

One last example to show that cmpu quality declines are no fluke can be found in Intel's first 32 bit cmpu, the 386.

---

[20] It is also pertinent that Intel's competition (i.e. AMD and Cyrix) were gaining market share in this segment.

[21] Based on Intel's composite performance benchmark which they call I-Comp. The 266Mhz Pentium II was rated by Intel at 3.03 I-Comps, but Intel could only coax 2.13 I-Comps from the 266Mhz Celeron.

[22] One of the reasons Intel wanted to steer the PC market away from the 386 to the 486 was because AMD offered a competing 386. In fact AMD offered a less expensive and faster 386 (40Mhz vs 33Mhz).

[23] The emasculation of the 486 was well documented in technical journals such as the *PC Processors Guide @* http://www.x86.org/articles/computalk/help.htm which stated that, *Unbeknownst to the consumer, the 80486 SX was an 80486 DX with a non-functional math unit (though later versions of the chip actually removed the math unit).*

The 386, like the 486, were offered in a DX and SX version. The DX was introduced in 1985, followed by the SX in 1988. As in the previous example, Intel felt the need to move the 386 from the relatively low volume high performance segment of the market to the high volume low end of the market without sacrificing the DX's high profit margins[24]. Intel wanted to protect high margin DX sales by introducing a 386 with lower performance characteristics which they called the SX. Unlike the 486, the 386 had no FPU to disable. Instead, Intel decided to go back to the old 16 bit system bus used in the predecessor 286. Of course mating the 32 bit 386 with a 16 bit system bus seems a bit odd, but was successful in the marketplace. Essentially Intel was offering a cmpu that could internally process data 32 bits at a time, but could only communicate with other system components, such as DRAM, 16 bits at a time. Intel's I-Comp performance benchmark[25] showed that their re-introduction of old 16 bit technology for the SX reduced performance 26 percent relative to the DX.

These examples have been presented for two reasons. The first is to show that cmpu performance has not followed an unbroken upward path over time despite the general accuracy and popular acceptance of Moore's Law. The second is to point out that if the PPI adopts new procedures for valuing cmpu quality improvements without also capturing quality declines, then price measurement will have an unavoidable downward bias of unknown magnitude. This caveat is also relevant when applied to alternative cmpu indexes based on average price data that are produced outside of BLS from time to time.

It would be disingenuous to imply that the PPI has been able to properly value and account for technological change in its cmpu price measurements. The standard PPI methodologies for valuing quality change is rather limited when faced with quality improvements that are accompanied by reduced input costs due to shifts in the production function.

The remainder of this paper will describe possible solutions to the problem of valuing cmpu technological change. One of the criteria for an acceptable alternative is that it must enable the PPI to calculate reasonable and consistent estimates for the value of technical change so that the residual price relative yields a measure of pure price change.

Standard PPI Quality Adjustment Procedures

Choosing among standard PPI methodologies to adjust cmpu price relatives for violations of the targeted Fixed-Input Output Price Index (FIOPI) model is a conceptually difficult obstacle. To cut to the quick, I will use simple price relative calculations to clearly show the consequences of choosing among standard PPI procedures to value quality change. The PPI's Laspeyres formula is ignored because it is needlessly cumbersome for this purpose and more useful for comparing index formula aggregation effects which are beyond the scope of this paper. The

---

[24] The 386DX cost Intel $141 and was sold for $900 according to *Inside Intel*, pg. 282.
[25] The University of Berkeley maintains a web site *@www.infopad.eecs.berkeley.edu/CIC/suimmary* that provides a history of Intel performance benchmarks that are no longer directly available from Intel. According to this site, Intel rated the 25Mhz 386 DX at 49 I-Comps and the 25Mhz SX at 39 I-Comps. Note that I-Comps are recalibrated over time so that they are often not comparable from one cmpu generation to the next.

examples are limited to the most common type 1 quality changes because if an acceptable solution[26] for type 1 is adopted, then it should also be applicable to type 2 and 3 changes.

The pricing data shown in table 6 overlap different input technologies and will be used to contrast the differences among standard PPI quality adjustment procedures on price relative calculations at the item level.

*Table 6

	60Mhz (0.80 micron)	**66Mhz (0.80 micron)**	90Mhz (0.50 micron)	120Mhz (0.35 micron)
2Q93	$878	**$965**	N/A	N/A
2Q94	$675	**$750**	$849	N/A
May 95	$245	**$260**	$377	$935
Nov 95	$230 discontinued	**$230 discontinued**	$247	$357

*Prices are based on 1,000 unit order size. From *IMF*, 3rd Ed., Tbl. A2, pg. 113.

We can further simplify with the assumption that the domestic cmpu universe was limited to 60 and 66Mhz Pentiums in 1993. The PPI samples this universe and disaggregates to the 66Mhz version. Tracking price change for this cmpu to its end-of-life on Nov 95 is straightforward. Based on the prices in table 6, example A shows that the PPI's Microprocessor index would have dropped 76 percent over this 2 ½ year period.

Example A: $P66/$P66= [1- $230/$965] = -76%

However, since both 60 and 66Mhz are discontinued in Nov. 1995, what replacement cmpu should be selected to maintain index continuity?   The closest match is 90Mhz produced with the more advanced 0.50 micron input technology. One of the problems presented by the proposed substitute is that it represents a shift in the production possibilities curve. In other words, the assumption of fixed input quantities, including technology, that is basic to the PPI's target FIOPI model has been violated. The correct adjustment for this violation is one of the most difficult challenges faced by PPI analysts. Standard PPI methodology offers several possible techniques that will theoretically address this situation depending on the amount of information that is available[27].

Direct Compare

For example, if the discontinued 66Mhz and its 90Mhz replacement are deemed to be very similar and no cost information is available, perhaps a direct comparison is appropriate. Using direct comparison, the $17 premium for the replacement results in a 7.3 percent increase in example B.

Example B: $P90/$P66= [1-$247/$230]= +7.3%

On the other hand, the replacement has a speed rating that is 36 percent faster than its predecessor. This is hardly an insignificant difference and is completely ignored in a direct comparison. If cmpu output were defined in terms of Mhz, then the PPI's use of direct comparison will grossly understate output.

---

[26] Any proposed methodology designed to value cmpu quality change must take into account a resource constrained operational environment that may affect the choice of the "best" solution.

[27] Gousen, Monk, and Gerduk provide an overview of standard QA procedures in *Producer Price Measurement, Concepts and*

Direct Link

Standard PPI methodology also includes a direct link procedure to compare prices of a discontinued product and its replacement. With direct link, an implicit assumption is made that the entire price difference between the obsolete product and its replacement is due to quality change. In other words if we view quality change strictly in terms of Mhz, then direct link treats the $17 price premium for the replacement as a valuation for the increased quantity of Mhz. The direct link adjusted price relative is shown in example C.

Example C: $P90\text{-}VQA/\$P66 = [1\text{-}(\$247\text{-}\$17)/\$230] = 1\text{-}\$230/\$230 = 0.0\%$

Direct link is often used when the PPI analyst does not have the information required to estimate an explicit quality valuation and the difference between the old and new products are significant enough that a direct comparison appears to be inappropriate. The direct link, if used frequently for products like cmpus that undergo significant and rapid change, will almost certainly introduce bias of unknown magnitude and direction into the PPI. The problem is an old one, but takes on added importance as "high-tech" products continue to grow relative to GDP.

Explicit Quality Adjustment

The preferred methodology in the PPI's quality adjustment arsenal is an explicit valuation of technical change. Using the previous example of the 66Mhz to 90Mhz transition, the PPI analyst would ask the reporter to provide an estimate for the change in marginal cost that is directly tied to the 24Mhz difference between old and new products. In a output index, significant improvements in quality are assumed to require additional inputs and it is the cost of additional inputs that are used to value related quality improvements. If the opposite occurs and quality declines significantly, say from 90Mhz to 66Mhz, then it is assumed that input quantities have been reduced and the cost reduction associated with new input requirements are used to value quality decline. In both cases a violation of the FIOPI model has occurred.

The methodology of using changes in marginal input cost to value quality change, in theory, returns the PPI to the original base period production possibilities curve. Some may say that we have "fixed" the violation of the FIOPI model when quality adjustment procedures are applied correctly. To illustrate, let's pretend that we have not yet reached a transition point in which the industrial revolution is giving way to the information technology revolution. The traditional industrial sector is generally more amenable to the PPI's explicit quality adjustment assumptions. More specifically, in the short term, measurements of increased quality are positively correlated with more costly inputs (higher marginal cost). Under this scenario, if the PPI needs to replace 66Mhz with a 90Mhz and we discover that the 90Mhz product has an additional marginal cost of $50 then the PPI analyst is on solid footing. Applying standard explicit quality adjustment procedures, the price relative is simply recalculated in example D to account for the increased marginal cost that is tied to the 24Mhz improvement.

Example D: $P90\text{-}VQA/\$P66 = [1\text{-}(\$247\text{-}\$50)/\$230] = 1\text{-}\$197/\$230 = \text{-}14.3\%$

Methods, pgs 95-99.

In example D the effect of the $50 dollar valuation for quality improvement is to transform a 7 percent increase in the directly compared nominal relative (example B) into a 14 percent decline in the explicit quality adjusted price relative. As long as the change in marginal cost exceeds the difference in nominal prices, then a pure price decline has occurred. If the increase in marginal cost is less than the increase in nominal prices then a pure price increase has occurred and if the change in marginal cost is equal to the difference in nominal prices then no change will be recorded (equivalent in effect to a direct link).

In the context of a deflator, because the PPI's explicit quality procedure for the new product "adjusts" the price relative from a nominal increase to a decline, the "real" measure of cmpu output increases. On the other hand, direct comparison makes no adjustment for technological change and simply measures the nominal price increase which decreases the real measure of cmpu output. In this case we know that the valuation of technology change embedded in the 90Mhz cmpu exceeds its price premium relative to 66 Mhz and therefore represents more output. Thus, the explicit quality adjustment procedure provides a closer approximation for measures of real cmpu output.

But if we try to use the explicit quality adjustment procedure to "fix" violations of the FIOPI model when quality improvements are accompanied by declining resource costs then we run into trouble. Using the rationale in the previous example, if technological improvements result in lower unit input costs, then the amount of this reduction should be added to rather than subtracted from the numerator of the price comparison. Or, if a technological decline results in higher input costs, then the amount of the increase should be subtracted from the price comparison. Both instances, using circular logic, should return the PPI to the original production possibilities curve.

The real effect would be as follows: The 66Mhz cmpu has reached end-of-life and is replaced by 90Mhz to maintain index continuity. The reporter informs the PPI analyst that the 90Mhz cost $50 less than the 66Mhz version. If we blindly follow the standard explicit quality adjustment formula then the adjusted price relative comparison will take the form in example E resulting in a 29 percent price increase.

Example E: $P90+VQA/$P66= [1-($247+$50)/$230]=1-$297/230= +29.1%

Remember that the nominal price increase associated with new cmpu is only 7 percent. We can make the example more extreme if the reporter decides to pass on half of the cost savings to the consumer by offering the 90Mhz for $222 instead of $247. The adjusted price relative is restated to show this in example. F.

Example F: $P90+VQA/$P66= [1-($222+$50)/$230]=1-$272/$230= +18.2%

In example F the nominal price change is a negative 3.5 percent (1-$222/$230 = -3.5), but the PPI "quality-adjustment" transforms the nominal decline into an 18 percent increase for the faster, cheaper replacement.

Explicit quality adjustment will continue to be used extensively in the PPI, but one can question if this procedure "fixes" violations of the FIOPI model for the rapidly growing information technology sector. As high technology

industries continue to grow in importance, many of the tools and concepts used by PPI analysts may need to be refined if we are to properly account for rapid quality change. I do not mean to imply that all information technology related industries constantly improve quality while simultaneously reducing unit cost. However, in addition to semiconductors, other high profile industries such as computers and telecommunications continue to offer greater functionality and performance while lowering unit production costs relative to predecessor products.

A case can be made that semiconductors take on the role of mother and computer and telecommunication products the role of children. The Semiconductor Industry Association's (SIA) 1999 directory estimates that computers contain more than half the semiconductors sold worldwide. Advances in semiconductors that can be characterized as "better and cheaper" are quickly adopted by computer and telecommunication equipment manufacturers to produce better products with lower input costs. No one questions that technical capabilities of computers have advanced tremendously over the last decade while prices have plunged[28]. To the extent that advanced procedures such as hedonic models are used to value quality change in computers, but not semiconductors, measures of output and productivity at the industry level are distorted. One of the consequences of this distortion is that real computer net output has been overstated due to an understatement of their most important input, semiconductors.

Alternative Quality Adjustment Methodologies

The challenge of accounting for new input technologies that reduce unit cost while improving quality was originally addressed by the PPI in the late 1980s when hedonic regression techniques were applied to an experimental computer index[29]. After careful review of the performance and production worthiness of the experimental indexes, the PPI officially began publication of Computer industry price indexes on December 1990. These were the first indexes to be integrated into the PPI structure that employed hedonic models to value quality change. In the field of government produced economic data, the PPI was not entering uncharted territory. The BEA introduced hedonic models[30] into the NIPAs to produce constant quality price indexes for computers in 1985. BEA's chief economist at the time, Jack Triplett, presents a clear description of the conceptual framework for applying hedonic techniques in *The Economic Interpretation of Hedonic Models*, Survey of Current Business, Jan. 1986.

The introduction of hedonic models into the NIPAs did not go unchallenged. Jorgenson and Stiroh (1994) present an interesting description of the controversy in *Computers and Growth*. They describe a "heated exchange" between BEA and Edward Dennison, *one of the founders of NIPA methodology in the 1950s and head of national*

---

[28] PC Magazine (8-88) reviewed a Compaq Deskpro 386S PC equipped with a 16Mhz 386, 1MB of DRAM, 40MB hard drive and DOS 3.1 that sold for the astronomical (by today's standards) **$5,199**. Compaq currently (1999) offers desktop PCs with a Pentium III 500Mhz, 64MB of SDRAM, DVD drive, 17" Monitor, 3D Video capability, sound card w/speakers, MS Office and one year on-site warranty for around $2,000.

[29] Sinclair, Catron, *An Experimental Price Index for the Computer Industry*, Monthly Labor Review, Bureau of Labor Statistics, Oct. 1990, pgs. 16-24.

[30] BEA's model was based on a IBM study, *Quality-Adjusted Price Indexes for Computer Processors and Selected Peripheral Equipment*, that was presented in the Jan. 1986 issue of Survey of Current Business, pgs. 41-50.

accounts at BEA from 1979 to 1982. Dennison *attacked* the use of hedonics and *argued vigorously against the introduction of constant quality indices into NIPA.* Triplett provided some of the key counterpoints to Dennison's position, particularly that correctly specified[31] models not only measured changes in consumer utility but also changes in marginal cost. Triplett went on to say that *...implicit prices measure value on both sides of the market, as do any prices*.

If Dennison had prevailed, then one can assume (in a worst case scenario) that the BEA would have continued to estimate computer industry output by measuring changes in nominal revenues. Measuring changes in computer output by changes in nominal revenue implicitly assumes a price index of 1 for the three decades of tremendous advances in computer technology prior to 1985. This of course is a difficult position to defend, but since the PPI did not introduce its own computer price index until 1990, the BEA had little choice but to develop an alternative.

Applying Hedonic Techniques in the PPI

The PPI's implementation of hedonics differs from the BEA in that price indexes are not directly calculated from a hedonic model based on pooled data and time dummy variables. Instead the PPI builds cross-sectional models to calculate values or implicit prices of computer characteristics. These values are then used to adjust prices reported to the PPI by producers when the characteristics of their sampled products change. Hedonic models and their supporting databases are updated on a regular basis (quarterly for desktop computers), to account for the rapid introduction of new characteristics.

Unfortunately correctly specified hedonic models for dynamic products can be elusive as technology induced disequilibriums may cause independent variables to have different interpretations for different observations. Comprehensive industry knowledge is one of the most important prerequisites in developing correctly specified models for technologically complex products. Otherwise, how do you judge the suitability of a supporting database (they are often expensive)? Even with a detailed understanding of technological features and how they interact, the availability of an appropriate supporting database is a major problem. This last point is not particularly newsworthy to anyone that has developed a hedonic model for high-tech products, but I mention the issue because it has a special relevance for cmpus.

I have no doubt that BEA's hedonic model for cmpus (see footnote 4) was supported by the required detailed product knowledge. But I do question the use of models based on long-term pooled databases within the PPI to adjust directly reported producer prices for current period quality change. The specifics of the problem are best presented through a brief review of commonly available cmpu characteristics. Rather than analyze each of the characteristics in BEA's model, the review is limited to the Mhz and transistor characteristics. Mhz and transistors are two of the most common and accurately identified features likely to be available in a supporting database.

---

[31] A correctly specified hedonic model is a reference to the selection of independent variables (product characteristics) that represent both outputs and inputs. Outputs absorb resources and inputs provide user value.

Interpreting the Mhz Characteristic

Mhz is often accepted as a measure of relative performance and was used as such in the BEA model. Mhz is also referred to as clock frequency which in turn is controlled by a small crystal that pulses at a steady frequency (or clock ticks). For each clock tick some action(s) can take place inside the cmpu, so as the frequency increases, more clock ticks are available per second to process data. For example, a 500Mhz cmpu has 500 million clock ticks available per second. In the world of cmpus, speed is the coin of the realm. A cmpu that can execute more instructions in a set period of time is able to run software more efficiently and therefore command a price premium relative to slower cmpus. However, MHz can be a misleading metric if more than one cmpu type, or generation is included in the same model. A quick look at Mhz ratings using Intel's 386-33Mhz as a baseline should help illustrate the problem.

Table 7

MPU Generations	Mhz	% change in Mhz
386	33	
486	66	100
Pentium	133	300
Pentium II	450	1,263

While it is clear that Mhz has increased rapidly from the 386 to the Pentium II, it still is a poor measure for the real increase in the speed of instruction execution as we progress through succeeding generations. The following is somewhat technical but necessary to understand the weakness of Mhz as a consistent measure of relative performance.

The 386 did not have a pipeline (a method of queuing instructions) which forced delays in the flow of instructions to its execution unit. In other words the 386 was sub-scalar and limited to about 0.8 executions per Mhz or clock tick. The 486 introduced a small pipeline which theoretically enabled one execution per cycle, the first scalar Intel cmpu. The Pentium introduced dual pipelines and execution units as well as branch prediction which enabled a maximum of two instruction executions per cycle, the first superscalar Intel cmpu. And finally, the Pentium II added[32] superpipelining, out of order and speculative execution which enabled a maximum of four instructions per cycle.

Using executions per cycle information, Table 7 data is adjusted in Table 8 to reflect the impact of these architectural changes[33].

Table 8

MPU	Rated Mhz	% change in Rated Mhz	*Rated Mhz Adjusted to Executions/cycle	% change in adjusted *Mhz

---

[32] The Pentium Pro was actually the first cmpu (1995) to introduce out of order and speculative execution, but is not included here due to technological tradeoffs in its design that would add unnecessary complexity to the examples.

[33] The architectural improvements described are key to understanding and quantifying their effect on Mhz. Unfortunately, most databases, even expensive ones, do not provide the level of technical detail required to properly specify a cmpu model. I may be proven wrong on this at some point. However, if this happy event should occur, there is the still the matter of limited observations that present another hurdle that would be difficult to overcome in a properly specified model.

386	33		26	**-21**
486	66	**100**	66	**153**
Pentium	133	**300**	266	**923**
Pentium II	450	**1,263**	1,800	**6,823**

*Executions per cycle multiplied by the rated Mhz, for example the Pentium II's adjusted Mhz is (450 * 4 executions per Mhz=1,800)

The data in table 8 clearly shows that if unadjusted Mhz is used in a model that includes more than one cmpu generation, then it is an unstable explanatory variable. The degree of instability (if history is any guide) will increase at an accelerated rate as the time span covered by a pooled database increases. The problem with specifying Mhz in a hedonic model is that it is really a proxy variable for other characteristics that are generally not included in supporting databases. For example, changes in Mhz for early generation cmpus such as the 8086, 286 and 386 (all referenced in the BEA model), are primarily a function of declining feature size enabled by new input technologies. This is a bit oversimplified, but the performance slope should be reasonably stable and linear as long as Mhz proxies similar, but unobserved characteristics. On the other hand, the 486, Pentium and Pentium II introduced new unobserved characteristics such as pipelines, register renaming, out of order execution and many other performance enhancing technologies. For these latter generations, Mhz takes on a new role as a proxy variable, because in addition to smaller feature size, Mhz is now a function of new technologies which change its performance slope dramatically. Proxy variables are particularly hazardous in pooled models that are used to investigate price/characteristic relationships for products that undergo rapid technological change. Even products that exhibit a slower rate of change require vigilance when proxy variables are employed. Triplett describes the use of weight as a proxy variable in early hedonic studies on automobiles. Weight was used as a proxy for the "true" characteristics that may have been unavailable or difficult to individually measure. Triplett describes the problem as follows…*Use of a proxy variable, however, introduces the possibility of error whenever the relation between the proxy and the true variables change, and one can never be entirely sure whether such shifts have occurred.*[34]

---

[34] *Economic Interpretation of Hedonic Models*, pg. 38, Current Survey of Business, January 1986.

Interpreting the Transistors Characteristic

Transistors are another popular variable used to quantify technological change and were included in the BEA model. This variable is also unstable over extended periods as a measure of relative quality change. If transistor counts are precisely defined, it is true that as the number of transistors increase and as their physical dimensions decrease, performance is enhanced. However, total transistor counts quoted by Intel and others do not always distinguish between transistors dedicated to logic and those dedicated to on-board memory (L1 cache). The distinction between transistors used for logic and those used for cache is important as will be shown.

Early generation cmpus (prior to the 486) used transistors primarily to form logic arrays that enabled them to process instructions and perform useful work. These early devices processed data/instructions (D/I) more slowly than system memory could deliver D/I. In other words, DRAM speed was sufficient to keep early cmpus "fed" with enough D/I that most performance bottlenecks were internal[35] to the cmpu. Starting with the 486 generation, cmpus could process D/I at a rate that substantially exceeded the speed of DRAM. The 486 and ensuing generations have a voracious appetite for data and DRAM was simply too slow to keep the new cmpus "fed". This performance bottleneck, absent a technological solution, would cause cmpus to stall as they had to wait on D/I before completing work in progress. Producers of cmpus were concerned because their new generation products were running into a performance brick wall caused by other system components. As long as DRAM set the performance ceiling, why would anyone pay a premium for a new relatively expensive 486 if they could get the same performance from a 386?

Producers devised a clever way to get around this performance barrier by adding small chunks of very fast memory directly into the integrated circuit design of their chips. This type of memory is generally known as L1 cache and requires vast amounts of transistors to implement. L1 cache operates at the same speed as the cmpu, but it can only hold a tiny amount (8 Kilobytes for the 486) of D/I relative to system memory. Cmpu producers placed stringent limits on the size of L1 cache for two reasons. The first was to minimize the unit cost of their new products. The second reason is not as transparent, but far more interesting. Savvy consumers could ask the logical question, if L1 cache has such a low capacity for holding D/I , how can it possibly make that much difference in removing the performance bottleneck presented by system memory? The answer is that L1 does not remove the entire bottleneck, but through the use of algorithms can eliminate 80-85 percent[36] of cmpu stalls that previously occurred as the result of accesses to system memory. The ability of producers to do more with less is based on the knowledge that most software requests for D/I are predictable. A software program may require several megabytes of DRAM to store all of its components and functionality, but most D/I requests are contiguous. In other words, if a function is provided by first executing instruction A, followed by B and then C, L1 cache need only grab B and

---

[35] I am ignoring other types of system interactions such as cmpus and disk drives.
[36] The percent estimate is based on synthesized tests performed by Intel to determine the percent of cache hits compared to cache misses for instructions that need to be loaded into the cmpu pipeline. The percent can vary according to the algorithm

C, while A is executing. When the cmpu is finished with instruction A, B and C are conveniently waiting in fast L1 cache for immediate access. However, if execution patterns are not correctly predicted by algorithms, then L1 cache provides little if any benefit. Let's say instruction A is followed by S and then by G. Since S and G are not contiguous with A, the algorithm used by the cmpu, may incorrectly load instructions B and C. This is called a cache miss and requires the cmpu to look for the S and G instructions in relatively slow system memory causing a delay.

The main point is that the algorithms used in L1 cache are so efficient at predicting software behavior that cache hits are far more likely than cache misses which eliminates most of the performance bottleneck caused by system memory accesses. With the introduction of L1 cache, producers dodged a technology bullet that could have greatly reduced the desirability and sales of new generation cmpus.

The reason for this level of detail is to establish the rationale for the adoption of L1 cache and it's inclusion in all subsequent generations[37] that followed the 386. So, what are the implications of this somewhat arcane description of evolving cmpu architecture? Logic transistors and memory transistors could hardly be more different. First, it is important to understand that logic transistors are smart and memory transistors are dumb. Logic is the core or active part of the cmpu that executes software instructions. Memory is passive and acts as a buffer between cmpu logic and slower system components. Prior to the 486, all transistor counts represented logic, simply because L1 caches had not been introduced. If transistors are used in a pooled model that spans a significant time period due to limited observations, then interpretation of the transistor characteristic takes on a different meaning for the 386 relative to the 486[38].

One expectation for a properly specified cmpu model is that price is positively related to increased quantities of technological characteristics. However, if technology redefines characteristics over time, then implicit prices for these characteristics may be difficult to interpret or even the "wrong" sign. Returning to the BEA model, it is reasonable to assume that as the number of transistors increased in early generation cmpus, this characteristic provided a sensible metric for quality change due to more efficient logic arrays and circuit design. However, once the 486 and its L1 cache were introduced, changes in transistors were due to changes in logic AND the addition of L1 cache. The BEA attempted to address this issue by defining a separate variable for cache, but Mr. Grimm mentioned that the cache coefficients varied from insignificant to the wrong sign and were therefore removed from the model. My hunch is that the problem with cache coefficients was due to a highly collinear relationship with transistors. Mr. Grimm also acknowledged *..the explanatory variables were highly correlated. All of the non-*

---

used and the type of software that is simulated.

[37] Memory speeds have improved in recent years through innovations such as EDO and SDRAM, but cmpu speeds have improved even more. Today's cmpus top out at around 1,000Mhz, but they are connected to memory that transfers D/I at 100 to 133Mhz. With this imbalance, Intel and other producers face the same performance issues first encountered in 1988 when L1 cache was introduced.

[38] Using a more recent example, the Celeron 300A had a transistor count of 19 million, of which only 7.5 million are logic with the rest dedicated to cache.

dummy variables for Intel chips had correlations with one another of .8 or higher. The more general problem with correlation is not surprising. After all a cmpu is an integrated circuit with most of its features/characteristics designed to work together with the ultimate goal of processing D/I as quickly as possible.

Additional Perspective on the CMPU market

If my references to the BEA model are taken simply as criticisms of their efforts, then the fault is mine. Mr. Grimm's paper has provided a valuable counterpoint to PPI coverage and played a role in our decision to investigate alternative means for improving cmpu price measurement. As previously stated, (and illustrated in chart 1) the most significant source of upward bias in PPI's Microprocessor index was due to non-response from a significant market participant on the producer side. There are few major products in the world that can be consistently produced for less than $100 and introduced to the market at $900 or more. The large difference between cost and initial selling price, enables large price reductions for existing products as new products are introduced to the market. Intellectual property, market inertia and large capital requirements are some of the reasons that more semiconductor companies have not been able to penetrate or reduce the concentration[39] observed in the cmpu market. The PPI has partially addressed this issue with the introduction of reliable secondary source prices for specified cmpus sold at a specified quantity. However, the PPI has never considered the addition of secondary prices as a resolution to the difficult problem of providing constant quality measures of cmpu price change. Separating pure price change from quality change requires a more rigorous framework than direct-link, average prices, or hedonics.

As an alternative to standard PPI methodologies, hedonic models have provided a means to move the analysis of quality change from an often opaque product space to a more transparent characteristics space. However, if data is unavailable or inappropriate to support a robust hedonic model, then products such as cmpus may require a different approach.

---

[39] The fact that the rapidly changing cmpu market and it's technology remains highly concentrated in terms of products offered and pricing strategy makes for a fascinating story. The story becomes more compelling when you consider that this control has extended for more than a decade. Contributing factors are manifold and beyond the scope of this paper. However, it is the uniqueness of the cmpu market that should discourage any attempt to use measures of price change for cmpus as proxies for empus or microcontrollers, much less other semiconductor products. And it is also true that any price index designed to measure mpu price change cannot accurately do so without including mainstream cmpu products and prices.

If Not Hedonics, Then What?

The difficulties associated with resource cost and hedonics has left the PPI with essentially no operational tool to explicitly value cmpu quality change. Unfortunately, the fall-back direct link procedure may induce a downward bias to the PPI's Microprocessor index if substitute cmpus routinely command price premiums that exceed reasonable valuations of quality change[40]. The term reasonable is open to interpretation, but metrics are available that may enable estimated valuations of cmpu quality change.

Fortunately, even when a hedonic model proves inadequate, the process of identifying important product characteristics may be useful for evaluating alternative valuation methods. In the case of cmpus, most characteristics are designed to interact in ways that speed the execution of software instructions. A simple model could take the form of Eq.1 which identifies some of the interdependent performance enhancing characteristics on the right side of the formula; and price, which tends to increase as performance increases, on the left.

$$Eq1. \rightarrow P_i = \hat{\beta}_1 + \hat{\beta}_2 X_2 + \hat{\beta}_3 X_3 + \hat{\beta}_4 X_4 + \hat{\beta}_5 X_5 + \hat{\beta}_6 X_6 + \hat{\beta}_7 X_7 + \hat{\beta}_8 X_8 + \hat{u}_i$$

X_2=L1 KB (Data)	X_3=L1 KB (Instructions)	X_4=Transistor Logic	X_5=Pipeline Depth
X_6=Instructions per Cmpu cycle		X_7=Speculative Execution	X_8=Register Renaming

The characteristics used in Eq. 1 have individual and joint effects on performance which in turn is the most important determinant of cmpu prices. One of the assumptions for any reasonable model is that price is positively related to increased quantities of technological characteristics. If these characteristics are constantly redefined by technology, then the model must have sufficient flexibility to adapt in real time, particularly for a monthly price index such as the PPI. A key to developing a flexible model is to avoid a rigid specification regime that is based on the obvious but interdependent characteristics that define a cmpu's architecture. Technological advances such as deeper pipelines, more memory registers, faster and bigger caches, out of order execution or even speculative execution are all designed to work together[41] with one overriding objective. That is, they are introduced or improved to increase the speed at which cmpus execute integer and/or floating point instructions.

Since most of the important characteristics that define cmpus are related in some way to performance, perhaps a more direct approach would better serve PPI quality adjustment requirements. Mr. Gwennap, previously cited in this paper, has stated that *microarchitecture is merely a tool to deliver performance to the end user*. If we first accept that changes in cmpu characteristics are exposed by changes in relative performance then we have implicitly identified a possible alternative to the hedonic approach. Instead of trying to determine the effect of multiple interdependent characteristics on price, a more direct and potentially productive inquiry simply investigates the effect of cmpu performance on price. The reference to "simply investigates" is a bit misleading. I don't remember

---

[40] Or an upward bias if the reverse is true.

[41] A user friendly description of the dependencies among cmpu characteristics is presented in *PC Processor Microarchitecture, A Concise Review of the Techniques Used in Modern PC Processors*, Microprocessor Report, 7-12-99, pgs. 16-22.

the exact source, but I recall an old industry reference to "lies, damn lies and performance benchmarks" or something to that effect.

Performance Benchmarks: An Alternative to the Alternative

There are many ways to measure relative cmpu performance, none of them perfect. One of the most common, but misunderstood benchmarks is known as MIPS (millions of instructions per second). If MIPS is taken at face value, it can be easily misused as a measure of relative cmpu performance. As far back as 1988, technical reports were published that attempted to describe the shortcomings of MIPS because they were often ignored in an attempt to simplify a complex problem. *Understanding Benchmarks*, authored by Mr. Jim Geers of AIM Technology, presents the simple example of a Motorola 68020 16Mhz cmpu that tested at 70 MIPS in one study, 131 in another and 218 in yet another. Which study is right? Keep in mind that we are talking about identical cmpus operating at the same Mhz. The reason such extreme variance was encountered is due to lack of comparability in the testing environments. Cmpus are designed to interact with other computer components such as memory, diskdrives and chip sets. Chip sets control the flow of information among system components much as traffic lights and yield signs control the flow of motor vehicle traffic. Each of the three tests used differently configured computer systems which directly caused the variance in MIPS. Unfortunately, this was a common scenario in the 1980s which meant that MIPS ratings were best viewed as measures of relative performance among different computer systems, and NOT as a measure of relative cmpu performance.

Another equally serious MIPS issue is that it remains a poor measure of cmpu performance even with identically configured computer systems. One might reasonably think that if system components are held constant across test observations then the problem described above is eliminated or at least reduced. In such a tightly controlled testing environment, it is tempting to view changes in MIPS as an objective measure of relative performance. This would enable a conceptually simple metric that may help to isolate price change from quality change. Once again, it is not so simple. MIPS proponents can trap themselves with the seductively easy appeal of millions of instructions per second as a reasonable and objective performance measure. If true, a 10MIPS cmpu should be twice as fast as 5MIPS. Cmpus are complex and I have tried to keep the tech talk to a minimum, but we must ask the question, what is an instruction? Before tackling this question, it should be obvious that instructions, like system components, must be comparable across performance tests. Otherwise changes in a MIPS benchmark due to non-comparable or ill-suited instructions will not accurately isolate changes in relative cmpu performance.

Instructions come in various forms and levels[42]. At the lowest level we have instructions that are written in machine language which only use numbers and are therefore rarely used directly by programmers. Instead, programmers have access to what are known as high level languages (HLLs) to write their programs/instructions. HLLs employ a relatively user friendly syntax to encapsulate machine language in large scale commands. Once a

---

[42] Randall, Neal, "Dissecting the Brains of Your Compute" and "What Makes Your Processor Think", PC Magazine, 6-30-98,

program is written in a HLL such as C, it must be translated back to a low level machine language because cmpus only understand strings of numbers. The translation occurs in a compiler in a two step process. Translation first separates instructions that are architecturally neutral from those that are specific to the general design of a cmpu such as CISC or RISC. CISC is based on a different instruction set than RISC which means that a CISC device can not recognize and execute the exact same instruction set as a RISC. There is also the issue of compiler tweaking that enables instructions to be executed in such a way that one cmpu can be unfairly favored over another. Ultimately the mix of instructions (simple and complex) that are directed to the cmpu and how they are compiled can have a greater impact on MIPS than actual real-world differences in performance.

Returning to our Motorola example, Mr. Geers, using different versions of UNIX software, observed a 50 percent difference between the high and low MIPS ratings for identical 68020-16Mhz cmpus. Unlike the MIPS tests in the first example, Mr. Geers used a single computer system to generate the results. By eliminating variance among hardware components, Mr. Geers was able to illustrate that variance among software instructions could also significantly affect MIPS ratings. To make matters worse, anyone, no matter what their qualifications can release MIPS test results conducted under whatever test conditions they feel are appropriate. With such chaotic testing conditions it is curious that MIPS have received such extensive coverage. Fortunately most of this coverage has been limited to advertisements and marketing hype, but not as a serious measure of relative cmpu performance among cmpu designers and engineers.

Computer configurations and instruction sets are not the only factors that cloud the interpretation of benchmarks like MIPS. Rather than continue with descriptions of test-induced bias, I will simply point out the obvious. Performance tests that are conducted without rigorous, meaningful, and impartial environmental controls are not generally useful despite claims to the contrary. Fortunately significant progress was made in the 1990s to provide more objective measures of relative cmpu performance. Forearmed with the "there are lies, damn lies and performance benchmarks" caveat, we turn to one of most respected cmpu performance metrics in the industry.

SPEC

The Standard Performance Evaluation Corporation (SPEC) was formed in 1988 as a non-profit corporation devoted to *establishing, maintaining and endorsing a standardized set of relevant benchmarks that can be applied to the newest generation of high performance computers*. Their first product was called SPEC89 *which provided a standardized measure of compute-intensive microprocessor performance. This product replaced the vague and confusing MIPS and MFLOPS ratings then used in the computer industry* [43]. One of SPEC89's strong points was that it enforced rules that ensured that all test platforms performed exactly the same reproducible operations *providing comparability across different architectures*.

---

http://www.zdnet.com/pcmag/pctech/content/17/12/tu1712.001.html.

[43] Mr. Jeff Reilly, Intel Corporation in a September 1995 press release from SPEC. The SPEC organization made a stronger statement on their web site (@www.spec.org/spec/) observing that "an ounce of honest data was worth more than a pound of marketing hype."

SPEC membership includes industry heavyweights such as AMD, Apple, Compaq, Dell, Hewlett Packard, IBM, Intel, Microsoft, Motorola, and Sun as well as representatives of industry trade journals and universities. The power of SPEC lies in its control over the definition of all performance benchmarks that carry the SPEC name.

One of the most well-known SPEC tests in the Microprocessor industry is CPU95[44]. This benchmark is specifically designed to measure the relative performance of cmpus. It is relative, because CPU95 is a performance index that references the test results of a 40Mhz Sun SuperSparc which equals one by definition. If another cmpu, such as a 450Mhz , has a CPU95 rating of 17, then it has executed the same set of instructions as the reference system, but 17 times as fast.

SPEC CPU95: Caveats

We already know that cmpus are proficient number crunchers, but we need to drill a little deeper to better understand what the SPEC performance benchmarks are designed to measure. A general purpose benchmark such as SPEC must account for two kinds of numbers processed by modern cmpus, whole and fractional, also called integer and floating point (fp). Integer instructions are relatively easy to process and handled differently than more complex instructions involving fp. Until the advent of Intel's 486, all X86s were optimized for integer calculations. Instructions that used decimal points, caused a significant performance penalty. To compensate for the fp problem, early generation cmpus were often connected to expensive specialized chips that could more efficiently handle decimal points. These chips were called math co-processors or X87 co-processors. Producers of modern cmpus have eliminated the need for co-processors by designing fp execution units directly into their architectures. However, most fp calculations still require more cycles than integer calculations.

The reason that fp requires more clock cycles relative to integer is due to the binary nature of cmpus which limit instruction processing to strings of zeros and ones. The fact that cmpus only understand zeros and ones is not a serious limitation for processing whole numbers. However, instructions that include fp create additional complexity because special handling is required to accommodate values on the right side of a decimal point. Significant digits must be separately stored as a unit called mantissa, while the radix point (decimal point in base 10) must be simultaneously stored in a separate unit called the exponent-for additional information see www.techweb.com/encyclopedia/defineterm?term=FP. Engineers have responded by building separate dedicated execution units into the processor core that are optimized for processing fp calculations. One might reasonably ask why bother with this level of detail? The short answer is that fp operations are extensively used in scientific, graphics and game/entertainment applications, while integer operations are used mainly in word processing and general business applications. Benchmarks that only measure integer or only fp will present an incomplete picture of cmpu performance. SPEC directly addresses this issue by defining CPU95 as a suite composed of *SPECint95,* and *SPECfp95;* the meaning behind these titles should be transparent.

---

[44] SPEC updates CPU95 approximately every 3 years to better reflect typical workloads placed on cmpus.

Because the SPEC organization controls and defines the procedures for running integer and fp calculations, relative measures of performance can be obtained across disparate cmpu architectures. Manufacturers must limit any optimizations to processes that mimic features that are available to their customers in standard products. No optimizations are allowed that are designed specifically to take shortcuts around a SPEC benchmark. For example, a cmpu manufacturer may want to take advantage of a relatively large transistor budget allocated to memory registers that minimize access to slower system memory. The result of such an optimization is reasonable and should yield improved real world performance relative to a non-optimized rating. This last point, illustrates one of the weaknesses of all so-called cmpu benchmarks.

There is no pure benchmark in the sense that all instructions designed to measure performance are handled entirely within the cmpu. SPEC acknowledges as much when they describe their benchmarks as *comparative measure<s> of performance of the processor, memory, and compiler on a known workload*.

By virtue of SPEC's control over compiler optimizations, memory is the primary wild card in their benchmark suite that could potentially distort relative measures of performance. Fortunately, any influence traceable to system memory differentiation are likely minor on most integer and fp tests conducted on mainstream cmpus[45].

Despite the caveats described (there are others of less immediate consequence), SPEC accurately characterize their CPUinteger95 benchmark as *representing the CPU-intensive part of system or commercial application programs* and their CPUfloating-point95 benchmark as *representing the CPU-intensive part of numeric-scientific application programs*. The integer and fp benchmarks do not measure performance related to computer I/O, networking, videocards, harddrives, CD-ROMs, etc. which make them superior measures of cmpu performance compared to other well-known alternatives such as MIPS, MFLOPS, WINBENCH or BAPCO.

---

[45] When SPEC tests are run on systems outside the commodity box environment, such as multiprocessor (SMP) servers, the effect of system memory on performance can be significant due to proprietary hardware design. Performance ratings generated from this type of environment should be avoided by the PPI for the following reasons. SMP systems are often designed to maximize data throughput in a client-server topology and therefore are not appropriate candidates for integer and fp ratings used to proxy cmpu quality change. SMP systems have their own unique benchmark requirements which SPEC provides under a different set of tests called *SPECrate*. Producers that attempt to use *SPECint* or *SPECfp* for SMP systems (and they do), are generally looking at how effectively multiple cmpus scale. In other words, how close can they come to doubling the performance of a two-processor system with a four-processor system. The PPI should ignore cmpu benchmarks obtained with SMP systems because we are attempting to measure quality change for specific cmpus, how OEMs connect them together in computer servers is irrelevant in a microprocessor index.

Using SPEC to Measure Cmpu Price/Performance

In the interest of maximizing useful information without disclosing confidential PPI data, Intel cmpus will be used in the following examples that apply SPEC benchmarks to establish price-performance ratios. Intel was chosen because most of their cmpu price data are publicly available and have been confirmed as representative of price change at the transaction level by computer OEMs that report to the PPI. Table 9 presents the raw SPEC integer and fp ratings for a range of mainstream cmpus beginning with the 133Mhz Pentium introduced on 5-95 and ending with the 450Mhz Pentium II introduced on 9-98[46]. All SPEC ratings shown are based on systems configured with 64MB of system memory.

Table 9

MPU	*SPEC int95	% Change	*SPEC fp95	% Change
PII 450	17.20	12.42	12.90	7.50
PII 400	15.30	14.18	12.00	11.11
PII 350	13.40	5.51	10.80	16.76
PII 333	12.70	9.48	9.25	8.95
PII 300	11.60	7.41	8.49	6.39
PII 266	10.80	14.41	7.98	9.17
PII 233	9.44	34.47	7.31	41.12
P5 233MMX	7.02	10.20	5.18	7.25
P5 200MMX	6.37	23.69	4.83	11.81
P5 200	5.15	13.44	4.32	8.82
P5 166	4.54	11.00	3.97	10.89
P5 150	4.09	3.54	3.58	1.42
P5 133	3.95		3.53	

*From www.spec.org

Ideally the integer and fp benchmarks should be combined into a composite measure of cmpu performance. However, a determination must first be made of the relative importance assigned to each component of the benchmark. The easy solution is to equally weight integer and fp, but such a scheme would grossly overstate the importance of fp in a general cmpu performance composite. Despite the rapid growth of multimedia and other specialized software that utilize fp, the majority of instructions processed by cmpus continue to emphasize integer calculations. A possible solution to the weighting issue is suggested by Intel's own proprietary benchmarks.

Intel publishes a separate set of cmpu benchmarks called iCOMP®. Interestingly, the Intel benchmarks also include SPECint and SPECfp, but incorporate additional application specific tests that are based on popular commercially available software. If we strip out the application tests which tend to be influenced by non-cmpu components such as disk drives or system I/O, then the relative weight of SPECint to SPECfp is 4 to 1. Adopting the 4:1 ratio appears to be a reasonable estimate for constructing a representative composite benchmark. This ratio may require periodic adjustment as the flow of instructions change over time in response to new technologies.

The recommended PPI version of a cmpu benchmark combines the SPECint and SPECfp results from table 9 into a

composite metric based on an 80-20 relative importance. This benchmark is simply called CMPUint/fp (SPECint *
0.80 + SPECfp * 0.20= CMPUint/fp) and is presented in table 10.

Table 10 MPU	CMPUint/fp	% Change CMPUint/fp	% Change Mhz
PII450	16.34	11.61	12.5
PII400	14.64	13.66	14.3
PII350	12.88	7.24	5.1
PII333	12.01	9.38	11.0
PII300	10.98	7.23	12.8
PII266	10.24	13.65	14.2
PII233	9.01	35.29	0
P5233MMX	6.66	9.72	16.5
P5200MMX	6.07	21.89	0
P5200	4.98	12.67	20.5
P5166	4.42	10.78	10.6
P5150	3.99	3.10	12.7
P5133	3.87		

Note that percent changes in the CMPUint/fp ratings differ substantially from changes in the Mhz ratings. This is
most noticeable when technology enables transitions from the P5200 to the P5200MMX and from the P5233MMX
to the PII233. In both transitions Mhz is unchanged despite technological advances that enabled major gains in
performance. This illustrates the sensitivity of CMPUint/fp to changes in relative performance that are invisible to
the inadequate Mhz rating.

Armed with a reasonable performance metric, price/performance ratios can be easily calculated by dividing the
price of a cmpu by its CMPUint/fp rating. Table 11 shows prices for a range of Intel cmpus from *IMF* and for each
of these cmpus a price/performance ratio is displayed in the $/Perf column.

*Table 11

Month/ Yr	$ PII450	$/Perf PII450	$ PII400	$/Perf PII400	$ PII350	$/Perf PII350	$ PII333	$/Perf PII333	$ PII300	$/Perf PII300
12-99	$230	14.06	$173	11.82	EOL	N/A	EOL	N/A	EOL	N/A
8-99	$230	14.06	$173	11.82	EOL	N/A	EOL	N/A	EOL	N/A
5-99	$268	16.40	$193	13.18	$163	12.65	EOL	N/A	EOL	N/A
4-99	$396	24.24	$234	15.98	$163	12.65	EOL	N/A	EOL	N/A
3-99	$476	29.13	$264	18.03	$192	14.91	EOL	N/A	EOL	N/A
1-99	$562	34.39	$353	24.11	$202	15.68	$181	15.07	EOL	N/A
10-98	$562	34.39	$375	25.61	$213	16.54	$181	15.07	EOL	N/A
9-98	$669	40.94	$482	32.92	$299	23.21	**$234**	**19.48**	**$192**	**17.49**
7-98	N/A	N/A	$589	40.23	$423	32.84	**$316**	**26.31**	**$209**	**19.03**
6-98	N/A	N/A	$722	49.32	$519	40.30	**$412**	**34.30**	**$305**	**27.78**
5-98	N/A	N/A	$824	56.28	$621	48.21	**$492**	**40.97**	**$375**	**34.15**
4-98	N/A	N/A	N/A	N/A	N/A	N/A	**$583**	**48.54**	**$530**	**48.27**
2-98	N/A	N/A	N/A	N/A	N/A	N/A	$722	60.11	$530	48.27
1-98	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	$738	67.21
8-97	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	$851	77.50

* All prices shown are based on 1,000 lot order size. Months for which no price changes occurred are not shown.

---

[46] SPEC ratings are generally available within a few weeks of the introduction date of a cmpu.

In a matched model, relative changes in price/performance ratios are exactly equal to changes in nominal prices. To illustrate how this data might be used, suppose a simple price index and its price/performance counterpart are composed of unweighted[47] 333 and 300Mhz cmpus with a reference or base date of 4-98. Using nominal price and price/performance ratios from table 11, a matched model can be maintained until 9-98, at which time the 300Mhz reaches end of life (EOL). Within this time frame it is of no consequence whether the index measures changes in nominal prices or price/performance. Formulas for both indexes are shown below, followed by equation 1 which calculates the respective index levels through 9-98.

$$\mathbf{Ip}_{t_b,t_c} = \Sigma \mathbf{P}_{t_c} \mathbf{Q}_{t_b} / \Sigma \mathbf{P}_{t_b} \mathbf{Q}_{t_b} \qquad\qquad \mathbf{Ipp}_{t_b,t_c} = \Sigma \mathbf{Pperf}_{t_c} \mathbf{Q}_{t_b} / \Sigma \mathbf{Pperf}_{t_b} \mathbf{Q}_{t_b}$$

Ip =Price Index $\qquad\qquad\qquad$ Ipp =Price/Performance Index

P =Price $\qquad\qquad\qquad\qquad\quad$ $Pperf$ =Price/Performance ratio.

$\qquad\qquad$ $\mathbf{Q}_t = \mathbf{1}$ in the unweighted indexes

Eq. 1 (Matched Model for 300 and 333Mhz cmpus; $t_b = 4\text{-}98$, $t_c = 9\text{-}98$)

$$\mathbf{Ip}_{t_b,t_c} = \Sigma \mathbf{P}_{t_c} / \Sigma \mathbf{P}_{t_b} = \$426/\$1113 = \mathbf{0.383}$$

$$\Sigma \mathbf{P}_{t_c} = \mathbf{300Mhz + 333Mhz} = \Sigma \$192 + \$234 = \$426$$

$$\Sigma \mathbf{P}_{t_b} = \mathbf{300Mhz + 333Mhz} = \Sigma \$530 + \$583 = \$1113$$

OR

$$\mathbf{Ipp}_{t_b,t_c} = \Sigma \mathbf{Pperf}_{t_c} / \Sigma \mathbf{Pperf}_{t_b} = \$36.97/\$96.81 = \mathbf{0.383}$$

$$\Sigma \mathbf{Pperf}_{t_c} = \mathbf{300Mhz + 333Mhz} = \$17.49 + \$19.48 = \$36.97$$

$$\Sigma \mathbf{Pperf}_{t_b} = \mathbf{300Mhz + 333Mhz} = \$48.27 + \$48.54 = \$96.81$$

Both of the indexes in Eq. 1 decline 62 percent (1.00 to 0.38) in the 4-98 to 9-98 time period, but index continuity is compromised on 10-98 when the 300Mhz cmpu reaches EOL. Unless a substitute is obtained for the obsolete cmpu, the indexes can only measure price change for the 333 which may be an inadequate measure of aggregate cmpu price change. To simplify the following description of product substitution, only the **Ip** formula is used. If we assume that a producer's entire output on 10-98 is defined by table 11 then either the 450, 400 or 350Mhz cmpus can be selected as a substitute. If the 400 is selected to replace the 300, then assume, since it is almost always true, that the PPI cannot explicitly value the quality difference. Under current methodology, the entire difference in nominal prices between the EOL 300Mhz and its 400Mhz replacement will be attributed to quality change, which is equivalent to what the PPI calls direct link. This approach reduces to:

PPI Implicit Valuation of Quality Change (IVQC) = $400Mhz-$300Mhz=$375-$192= $183

Using the direct link procedure with its implicit valuation of quality change, the index for 10-98 is shown in

---

[47]I am not discounting the importance of weight, much less choice of index formula, but this review is of necessity narrow in

equation 2.

Eq. 2 [(Direct Link of 300Mhz to 400Mhz) + 333Mhz $t_b = 4\text{-}98, \ t_c = 10\text{-}98$]

$$\mathbf{Ip}_{t_b,t_c} = \Sigma(\mathbf{P_{x1_c}} - \mathbf{IVQC_{x1_c}}) + \mathbf{P_{x2_c}} \Big/ \Sigma \mathbf{P_{x1_b}} + \mathbf{P_{x2_b}} = \$373/\$1113 = 0.335$$

$$\text{Where } \mathbf{P_{x1_c}} = \mathbf{400Mhz} \ ; \ \mathbf{IVQC_{x1_c}} = \$183 \ ; \mathbf{P_{x2_c}} = \mathbf{333Mhz} \text{ and } \mathbf{P_{x1_b}} = \mathbf{300Mhz} \ ; \ \mathbf{P_{x2_b}} = \mathbf{333Mhz}$$

$$\Sigma \mathbf{P_{t_c}} = (\mathbf{400Mhz} - \mathbf{IVQC}) + \mathbf{333Mhz} = \Sigma(\$375 - \$183) + \$181 = \$373$$

$$\Sigma \mathbf{P_{t_b}} = \mathbf{300Mhz} + \mathbf{333Mhz} = \Sigma\$530 + \$583 = \$1113$$

Relative to 9-98, the index drops 12.5 percent (from .383 to .335). The accuracy of the decline is an open question, because with direct link only the price relative of the 333Mhz cmpu can change since the entire price difference between the 300Mhz and its 400Mhz replacement is attributed to quality change. Only in the improbable event that the unobserved explicit (real) quality change valuation is equal to $\mathbf{IVQC_{x1_c}}$ does the direct link provide an unbiased estimate.

An important premise of this paper is that most of the advances in cmpu architecture are transparent to and can be represented by objective performance measures. One might say that change in cmpu output can be viewed as change in the amount of processing power that is produced. The CMPUint/fp ratings allow the PPI analyst to estimate this change. If changes in processing power provide reasonable measures for changes in output quality, then a clear benefit from this type of analysis is that explicit estimates of this change that are removed from the numerator of cmpu price relatives show up in quantity relatives through deflation.

Applying CMPUint/fp to Value Changes in Processing Power

Several steps are required to explicitly value quality change with a performance metric. The CMPU/int/fp rating must first be converted to dollar values so that price relative comparisons between an obsolete product and its replacement can be adjusted for differences in processing power. The obsolete 300Mhz had a CMPUint/fp rating of 10.98 (table 10) which the market valued at $192 in the 9-98 reference period (table 11). A 300Mhz price performance ratio calculated from these two values is shown below:

300Mhz price/performance ratio = Nominal Price/CMPUint/fp=$192/10.98 = $17.49

This ratio can be interpreted as the market value per unit of performance in the reference period using reference period technology. If the price per unit of performance for an incumbent cmpu, such as the 300, is multiplied by the CMPUint/fp rating of a substitute cmpu, such as the 400, then we can obtain the equivalent of a constant quality price based on the relative change in processing power. The 400Mhz substitute has a CMPUint/fp rating of 14.64 (table 10) that can be used to calculate its constant performance price as shown below.

focus.

Constant Performance Price For the 400 Relative to the 300 Cmpu					
400 CMPUint/fp	X	300 P/Perf	=	Constant Performance Price	
14.64	X	$17.49	=	$256	

An explicit valuation of quality change (EVQC) is obtained from the difference between the constant performance price of the substitute and the nominal price of the discontinued cmpu. The quality change valuation for the 400Mhz relative to the obsolete 300Mhz is shown below.

EVQC = 400Mhz Constant Performance Price - 300Mhz Nominal Price
=$256 - $192
=$64

Interestingly, when the 400 is introduced as a substitute, its nominal price is $375. The difference between the 400's nominal price and its estimated constant performance price can be viewed as a premium or additional margin available to the producer for this more advanced early-life cmpu relative to the EOL cmpu. This premium shows up in a constant quality index as a price increase. If the nominal price of the 400 on 10-98 had been less than its constant performance price, then a constant quality measure would capture a real price decline. The mechanics of calculating a constant quality cmpu price relative is summarized in table 12.

Table 12

Step 1.	Calculate Price Per Unit of Performance for obsolete CMPU
Step 2.	Multiply value from step 1 by the CMPUint/fp of the substitute to obtain Constant Performance Price Measure
Step 3.	Obtain explicit quality change valuation from the difference between the step 2 value and the nominal price of the obsolete CMPU
Step 4.	Evaluate explicit quality change valuation from step 3 with the difference between the adjusted price of a substitute and the nominal price of the obsolete CMPU

The cmpu index on 10-98 can now be recalculated to reflect this quality adjustment procedure as shown in Eq. 3.

Eq. 3 [(EVQC of 300Mhz to 400Mhz) plus 333Mhz; $t_b = 4\text{-}98,\ t_c = 10\text{-}98$]

$$\mathbf{Ip_{t_b, t_c} = \Sigma(P_{x1_c} - EVQC_{x1_c}) + P_{x2_c} / \Sigma P_{x1_b} + P_{x2_b} = \$492/\$1113 = 0.442}$$

Where $\mathbf{P_{x1_c} = 400Mhz}$; $\mathbf{EVQC_{x1_c} = \$64}$; $\mathbf{P_{x2_c} = 333Mhz}$ and $\mathbf{P_{x1_b} = 300Mhz}$; $\mathbf{P_{x2_b} = 333Mhz}$

$$\mathbf{\Sigma P_{t_c} = (400Mhz - EVQC) + 333Mhz = \Sigma(\$375 - \$64) + \$181 = \$492}$$

$$\mathbf{\Sigma P_{t_b} = 300Mhz + 333Mhz = \Sigma\$530 + \$583 = \$1113}$$

Applying the EVQC to the substitute cmpu causes the price index to increase 15.4 percent relative to 9-98 (0.383 to 0.442), while in the same period the direct link (IVQC) index (Eq. 2) dropped 13 percent. The increase caused by EVQC may seem unusual because of the commonly held perception that Moore's Law (faster, better, cheaper)

is not only at work for producers, but also consumers. Note that the unit cost of a 400Mhz cmpu is actually less than the cost of the obsolete 300 due to a more advanced input technology (0.25 vs 0.35 microns). The example constant quality index does not repeal the part of Moore's law that implies "cheaper", but does point out the importance of choosing a replacement that is the closest match to the obsolete cmpu. For instance, when the 300 met its demise, the replacement candidates were the 450, 400 or 350. The 350 was a closer "match" both in terms of performance and market position. If the 350 had been chosen instead of the 400, then the cmpu index would have behaved much differently. Applying the same procedures used in Eq. 3, the index is restated in Eq. 4 with the 350 designated as the substitute for the 300.

Eq. 4 [(EVQC of 300Mhz to 350Mhz) plus 333Mhz; $t_b = 4\text{-}98$, $t_c = 10\text{-}98$]

$$\mathbf{Ip_{t_b,t_c}} = \Sigma(\mathbf{P_{x1_c}} - \mathbf{EVQC_{x1_c}}) + \mathbf{P_{x2_c}} / \Sigma \mathbf{P_{x1_b}} + \mathbf{P_{x2_b}} = \$361/\$1113 = 0.324$$

Where $\mathbf{P_{x1_c}} = \mathbf{350Mhz}$; $\mathbf{EVQC_{x1_c}} = \$33$; $\mathbf{P_{x2_c}} = \mathbf{333Mhz}$ and $\mathbf{P_{x1_b}} = \mathbf{300Mhz}$; $\mathbf{P_{x2_b}} = \mathbf{333Mhz}$

$$\Sigma \mathbf{P_{t_c}} = (\mathbf{350Mhz} - \mathbf{EVQC}) + \mathbf{333Mhz} = \Sigma(\$213 - \$33) + \$181 = \$361$$

$$\Sigma \mathbf{P_{t_b}} = \mathbf{300Mhz} + \mathbf{333Mhz} = \Sigma\$530 + \$583 = \$1113$$

The derivation of the EVQC and constant performance price used to calculate eq. 4 is shown below.

Constant Price Performance For the 350 Relative to the 300 Cmpu

350 CMPUint/fp	X	300 P/Perf	=	Constant Performance Price
12.88	X	$17.49	=	$225.27

EVQC = 350Mhz Constant Performance price - 300Mhz nominal price
        =$225-$192
        = $33

Note that the nominal price of the 350 on 10-98 is $213 $(\mathbf{P_{t_c}})$, but its constant performance price is $225.27. In other words, the 350 sells at a discount to its constant performance price and the amount of this discount relative to the nominal price of the 300 will show up as a measure of a constant quality price decline.

As in eq. 3, the EVQC is subtracted from the 350's nominal price in eq. 4 to obtain a quality adjusted price relative. Using the 350 as a substitute instead of the 400, the cmpu index reverses directions and drops 15.4 percent relative to 9-98 (0.324/0.383).

The Margin Effect

It is important to understand why the choice of cmpu replacement for an EOL product can cause significant variation in measures of quality adjusted price change. Cmpu producer margins vary across their product lines, with relatively powerful cmpus commanding much higher margins than cmpus in the low-end segment or those approaching obsolescence[48]. To help illustrate, table 11 is repeated below.

---

[48] Moore's Law is also at work here.

Table 11

Month/Yr	$ PII450	$/Perf PII450	$ PII400	$/Perf PII400	$ PII350	$/Perf PII350	$ PII333	$/Perf PII333	$ PII300	$/Perf PII300
12-99	$230	14.06	$173	11.82	EOL	N/A	EOL	N/A	EOL	N/A
8-99	$230	14.06	$173	11.82	EOL	N/A	EOL	N/A	EOL	N/A
5-99	$268	16.40	$193	13.18	$163	12.65	EOL	N/A	EOL	N/A
4-99	$396	24.24	$234	15.98	$163	12.65	EOL	N/A	EOL	N/A
3-99	$476	29.13	$264	18.03	$192	14.91	EOL	N/A	EOL	N/A
1-99	$562	34.39	$353	24.11	$202	15.68	$181	15.07	EOL	N/A
10-98	$562	34.39	$375	25.61	$213	16.54	$181	15.07	EOL	N/A
9-98	$669	40.94	$482	32.92	$299	23.21	$234	19.48	$192	17.49
7-98	N/A	N/A	$589	40.23	$423	32.84	$316	26.31	$209	19.03
6-98	N/A	N/A	$722	49.32	$519	40.30	$412	34.30	$305	27.78
5-98	N/A	N/A	$824	56.28	$621	48.21	$492	40.97	$375	34.15
4-98	N/A	N/A	N/A	N/A	N/A	N/A	$583	48.54	$530	48.27
2-98	N/A	N/A	N/A	N/A	N/A	N/A	$722	60.11	$530	48.27
1-98	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	$738	67.21
8-97	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	$851	77.50

Note the relatively high P/Perf ratios in the early stages of a cmpu's life cycle. Producers have observed and responded to the fact that as they introduce new faster cmpus, the most performance demanding segment[49] of the market is willing to pay a significant premium relative to explicit measures of performance-based change. The premium can be seen as additional margin on the producer side and additional utility for a specific market segment. This price-performance premium or technological reward, is available to the producer even when unit manufacturing costs drop relative to lower quality, late life cycle cmpus[50]. Using table 11 data, we can see that the 300 was introduced on 8-97 as the fastest and highest margin X86 cmpu on the planet. By 9-98, the 300 had fallen so far behind the performance curve that it became obsolete. Within this 13 month life cycle, the 300 fell from its initial price of $851 to $192. The 333 suffered a similar fate, introduced on 2-98, but on the scrap heap by 1-99. The 400 was introduced on 5-98 at $824 and by 8-99 had declined to $173 as new faster cmpus were introduced. The large difference between cost and initial selling price, enables a company like Intel to rapidly reduce prices for existing products as they bring new products to market. Quoting from IMF[51], *A consistent theme in Intel's product strategy over the past few years has been breathtaking price cuts. An average of 25-30% per quarter, a rate of 65-75% per year! In other words, many Intel processors today cost one-third to one-quarter of their price one year ago. Over the long term, this rapid pace is fueled by the company's continual movement from one IC process generation* <new inputs> *to the next*. The life cycle descriptions and quote from IMF lend support to the notion that relative performance and market segmentation plays a more significant role than unit cost in determining

---

[49] Examples are ubiquitous. For example Intel's 1998 Annual Report states that *Intel's strategy is to introduce ever higher performance microprocessors tailored for the different segments of the world computing market, using a tiered branding approach…the Company's gross margin varies depending on the mix of types and speeds of microprocessors*.
[50] When unit costs drop additional large premiums would be unlikely in a more competitive market but in a less competitive market additional premiums combined with declining unit cost enables the dominant producer to acquire and then allocate significantly larger resources to capital expenditures and R&D.
[51] Intel Microprocessor Forecast, Product Roadmap, Volumes, Costs, & Prices, 4th Ed., pg. 37, Publisher: MicroDesign Resources, 1998.

pricing strategy for cmpus.

It should be clear that if an obsolete cmpu is replaced with one that is at or near the beginning of its life cycle and therefore commands a relatively high margin, constant quality comparisons within the PPI will usually result in a significant pure price increase. On the other hand, if we rigorously adhere to the goal of selecting cmpu substitutes that are the closest match in terms of features, performance and market segment, most of the margin variance disappears and the difference in nominal prices is generally less than the difference in relative performance. To illustrate the importance of identifying appropriate cmpu replacements, Appendix A presents three price indexes constructed over a two year period, each based on different substitution strategies. Significant margin differentials between early and late cycle cmpus are likely to continue as long as the market lacks vigorous competition in the high performance segment or if the laws of physics and their application to input technologies do not eventually repeal Moore's law.

The proposed methodology for estimating cmpu quality change can be complimented with a more aggressive substitution strategy. This strategy should have the goal of ensuring that closest match cmpu substitutes are selected while also satisfying the seemingly contradictory goal of maintaining a representative mix of early, mid and late life cycle products.

### Minimizing Potential Biases Induced By Rapid Product Displacement

Many of the high-tech producers in the PPI are in a state of perpetual revolution, where rapid innovation is a survival requirement. The business models for these producers depend on technical innovations that cannibalize existing products in order to maintain or increase market share. Technological improvements can quickly force predecessor products, including those of the competition, into a low-margin commodity status and then obsolescence. As products are displaced by technology, the relative importance[52] of the PPI's sampled output may diverge significantly from industry output. This displacement has potentially negative consequences for measures of real domestic output, value added by industry and productivity.

A substitution strategy that correctly adapts to rapid product displacement is complicated by several factors. We have already established that cmpu life cycles are exceptionally short, often less than a year. During this time, technological innovation can force a cmpu to move quickly through different market segments that have different demand functions. For example, when the PPI samples an early life cycle cmpu it may be a cutting-edge, relatively high-priced, but low volume product. Within a few months, steep price declines enable a transition to a mainstream, high volume product. In a few more months, continued price declines enable a transition to a relatively low-priced, low volume product and then obsolescence. The breathtaking speed of passage through

---

[52]Displaced products in this context usually have a different rate of price change and therefore are not accurate proxies for the newly dominant but unobserved product(s).

these roughly defined market segments[53] (high performance, mainstream, low performance) is illustrated in Chart 2.

Chart 2.
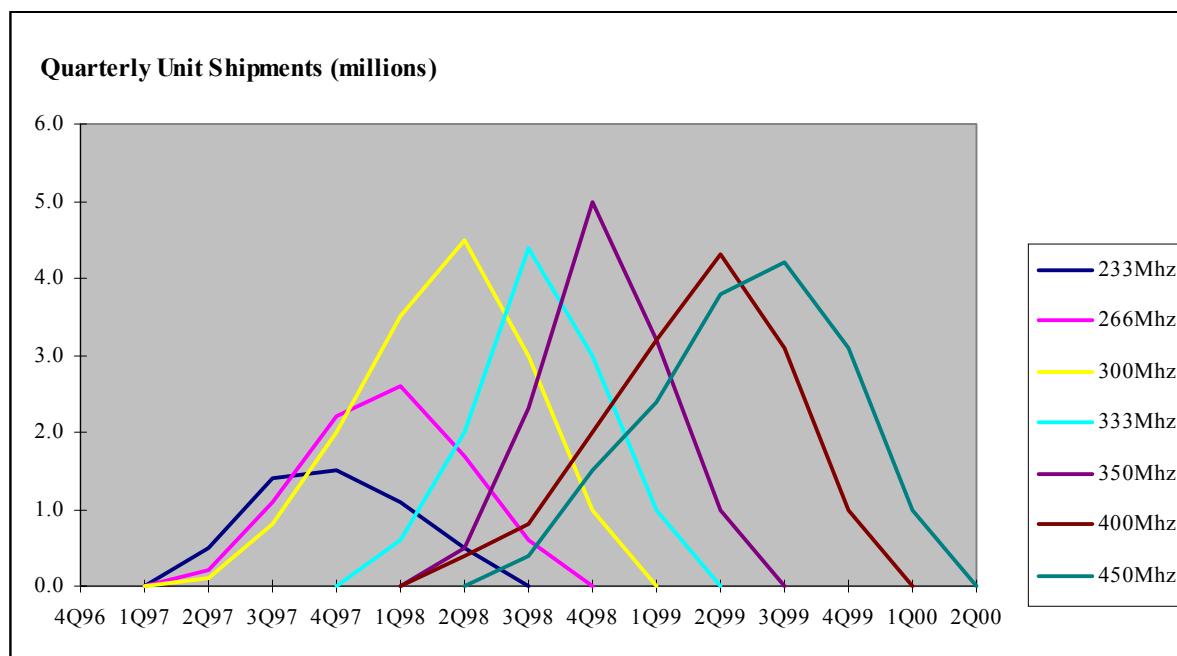
Constructed from estimated historical shipment data in "Intel Microprocessor Forecast",1H00Edition, MicroDesign Resources, pg. 132. The end-of-life points in the estimated shipments data have been adjusted to reflect publicly available pricing data from the manufacturer. The pricing data spanned the production life of each cmpu and had the effect of moving the end-of-life point for several cmpus back one quarter. The small difference is likely due to the shipments data including sales made from mfg. or distributor inventories after production had ceased.

One of the effects of the cmpus short and volatile life cycle can be described as follows. Assume that a PPI sample includes four cmpus that represent a cross section of the performance range shown in chart 2. As we progress through time, the price index is composed of cmpus whose relative importance in the marketplace are shifting so quickly that an overall measure of price change may be distorted[54]. For instance, if 4Q98 is arbitrarily chosen as the base date for a index comprised[55] of the 300, 333, 350 and 400Mhz cmpus, the rate of price change for each varies as it progresses through different life cycle stages (see table 13). The relative importance of these different rates of price change is exceptionally volatile due to large and rapid changes in unit volumes. The PPI

---

[53] Intel's 1998 annual report provides an interesting and relevant producer perspective. *The industry in which Intel operates is characterized by very short product life cycles, and the Company's continued success is dependent on technological advances…and implementations of new processes and new strategic products for specific market segments.*

[54] The most rapid price declines often occur in the early stage of a cmpu's life cycle but may slow as obsolescence approaches. If the rate of price change is stable throughout the cmpus life cycle, then relative importance at the item level is of less interest.

[55] The contiguous ordering within the performance universe of cmpus is used to simplify substitution choices. A sample based on probability proportionate to shipments, as in the PPI, may be tightly grouped for mainstream cmpus and less so for early and old life cycle cmpus.

cannot directly respond to monthly real-world changes in relative importance. Instead the analyst is faced with a continuous parade of late life cycle cmpus that are quickly forced into obsolescence, but is generally aware of the following market conditions that were illustrated in Chart 2.

- The 300Mhz has essentially reached end-of-life in 4Q98, with an extremely sharp drop in unit shipments from the peak reached in the previous two quarters.

- The closest replacement for the 300 in terms of technology and performance is the 333 which will match the rapid decline in unit shipments of the 300 and reach end of life in 2Q99.

- As the 333 reaches end of life the closest match both in terms of performance and lagged unit shipment trend is the 350 and the pattern continues.

*Table 13

Month/Yr	Non-Index 450Mhz	Index Item 1 400Mhz	Index Item 2 350Mhz	Index Item 3 333Mhz	Index Item 4 300Mhz
12-99	$230	$173	EOL	EOL	EOL
8-99	$230	$173	EOL	EOL	EOL
5-99	$268	$193	$163	EOL	EOL
4-99	$396	$234	$163	EOL	EOL
3-99	$476	$264	$192	EOL	EOL
1-99	$562	$353	$202	$181	EOL
10-98	$562	$375	$213	$181	EOL
9-98	$669	$482	$299	$234	$192
7-98	n/a	$589	$423	$316	$209
6-98	n/a	$722	$519	$412	$305
5-98	n/a	$824	$621	$492	$375
4-98	n/a	n/a	n/a	$583	$530
2-98	n/a	n/a	n/a	$722	$530
1-98	n/a	n/a	n/a	n/a	$738
8-97	n/a	n/a	n/a	n/a	$851

*Pricing data for the hypothetical sample is a subset of prices and cmpus shown in table 11.

With the market conditions described, the PPI analyst must obtain a replacement product each time a cmpu reaches end-of-life to maintain index continuity. As a general rule continuity is best obtained with the introduction of a substitute that is the closest match in terms of technology and market position. However, if this strategy is followed for an index that is composed of short life cycle products, then the index can be quickly biased towards late life cycle products. In other words, if the original sample included an early life cycle cmpu which subsequently becomes obsolete, then the closest match at the time of substitution is another late life cycle product that is approaching, but not yet reached EOL. For instance, when the 300Mhz reaches EOL, the 333Mhz cmpu is the closest match in terms of technology, market position, and price trend but is unavailable because it is already included in the price index. The closest AVAILABE substitute is the 450Mhz cmpu which is, at this point, an early life product that is produced with different input technologies and priced for a significantly different market. If the 450Mhz is introduced as a substitute for the 300Mhz, the PPI analyst is confronted with a nominal price relative of $562/$192. The complexity and uncertainty of quality adjusting nominal prices, particularly for high-

tech products, is magnified by the degree of dissimilarity between an old product and its substitute. A modification of the PPI's normal substitution strategy may help to improve index continuity by more rigorously adhering to the principle of closest match.

The proposed modification is straightforward, but requires the PPI analyst to view the sample holistically when replacements are necessary. For example, cmpus in our hypothetical price index are presented in table 14 that span a range of technologies and life cycles represented by 400Mhz (item 1), 350Mhz (item 2), 333Mhz (item 3) and 300Mhz (item 4). When the 300 becomes obsolete, rather than arbitrarily replace it with the next available cmpu that is not in the index (the 450), it is replaced with it's closest "match", the 333. Since the 333 is already in the index the mechanics of such a replacement strategy can be accomplished with the following actions. Item 4 is replaced with item 3, item 3 is replaced with item 2 and item 2 is replaced with item 1. The original item 1, which was occupied by the 400 and represented the early life, high performance, but low volume segment is now vacant but can be filled with its closest available match, the 450, which becomes the new early life cycle representative.

Table 14

Original Sample			
Item 1	Item 2	Item 3	Item 4
400Mhz	350Mhz	333Mhz	300Mhz
Original Sample After Obsolescence of 300Mhz (Item 4)			
Item 1	Item 2	Item 3	Item 4
400Mhz→	350Mhz→	333Mhz→	end-of-life
Updated Sample Via Modified Directed Substitution			
Item 1	Item 2	Item 3	Item 4
450Mhz	400Mhz	350Mhz	333Mhz

This replacement scenario may seem aggressive but has some desirable qualities. First, as cmpus reach end-of-life, the sample is adjusted so that it is essentially forced to maintain a representative product mix that spans the entire cmpu life cycle spectrum. Each item # in effect, becomes a placeholder for a predetermined life cycle stage that is roughly equivalent to the following: item 1 is reserved for low volume early life cycle cmpus, items 2 and 3 for high volume mature cmpus, and item 4 for low volume late life cmpus. This strategy also addresses the problem of tremendous disparity in unit shipments within various stages of a cmpus short life cycle. The 4Q98 time period (index base date) in chart 2 presents a snapshot of this weighting issue. Because item weights that are provided by the producer are held fixed at the base period, the placeholder system insures that a high relative importance mid-life cmpu, such as item 2, will not be introduced as a substitute for a low relative importance late-life cmpu, such as item 4. However, every two to three months (if history is any guide) an item 4 cmpu becomes obsolete, which under the proposed substitution procedure, triggers a simultaneous shift of the remaining cmpus among items 2 through 4 and introduces a new early-life cmpu for item 1. This procedure is likely to provide a better match for the abrupt changes in shipment patterns shown in chart 2. The effect on the price index from following this substitution pattern should be an improvement in the relative importance of cmpu price relatives in the Laspeyres-based PPI.

Unfortunately, a tailored substitution strategy for cmpus is not costless. The PPI analyst will be faced with making multiple[56] substitutions when a cmpu becomes obsolete compared to one substitute under the existing procedure. In addition, the analyst must now estimate valuations of quality change for each of the multiple substitutes. This strategy is more vulnerable to inadequate measures of quality change due to their greater frequency, but the previously described CMPUint/fp measure may address this issue.

Summary of CMPU Quality Adjustment Proposal

The use of open, technically agnostic and consistently defined cmpu performance benchmarks present the PPI with a means of valuing quality change that has important advantages over existing procedures. Several types of quality change are routinely exhibited by cmpus that marginalize the utility of conventional (including hedonic) techniques in a real time production environment. The availability of SPEC benchmarks across the range of currently produced domestic cmpus give the PPI an alternative procedure to isolate pure price change from quality change.

The perfect quality adjustment tool is elusive because valuations of quality change are routinely based on estimates that are in turn often based on incomplete data. Cmpus are an extreme example and the proposed CMPUint/fp benchmark is a compromise that in a best case scenario provides a rough approximation of values associated with observable changes in cmpu performance. Even in this best case scenario periodic adjustments may be required as the importance of integer and fp executions evolve along with cmpu architectures[57]. I should also note that CMPUint/fp is primarily targeted at mainstream cmpus designed for non-portable and non-server applications. Low-voltage versions of many X86 cmpus[58] are produced for notebook computers. Measuring quality change for low voltage cmpus is more complex because producers differentiate these products with technologies that are not as focused on performance. For example, input technologies for low-voltage cmpus place much greater emphasis on extending battery life and minimizing heat generation. Performance is still important, but it fails to capture significant technology features unique to the low-voltage products. It may be possible to develop a hybrid benchmark for low-voltage cmpus that measures relative changes in energy management features (such as watts/Performance) which could be added to the integer and floating point measure with an appropriate weighting estimate.

Another issue that has not been presented is the redefinition of the SPEC guidelines and test parameters that occur approximately every three years (1989, 1992, 1995 and 2000) . When SPEC updates their benchmarks they are

---

[56] The PPI microprocessor index currently has about 15 cmpu items. This does not mean an obsolete product will trigger 15 substitutes. Items are company specific and so are substitutes. Another factor that limits the range of substitutions, is that reallocation of cmpus among item placeholders only occurs within a cmpu family. For instance, the X86 market is composed of different technology families targeted at distinct markets such as low voltage cmpus for portable applications. A change in one family does not necessarily trigger a product rotation in another.

[57] As fp improves, cmpus may take on functions currently handled by peripheral equipment such as modems and sound cards. There is also the possibility that future cmpus may include entirely new executions units such as digital signal processors (DSPs) to handle these functions which may require the addition of a 3rd component to the composite benchmark, perhaps CMPUint/fp/dsp.

[58] Low-voltage cmpus represent about 15% of the cmpu market.

generally not backward compatible. However, these problems are relatively minor compared to the current lack of an effective cmpu quality adjustment methodology in the PPI. The ability to quickly calculate constant quality prices enables the PPI to substantially improve its Microprocessor index which also increase its accuracy as a deflator.

As long as the focus of cmpu technological change continues to broadly target the execution speed of integer and fp instructions, then CMPUint/fp may serve as a viable alternative quality adjustment tool. If the net effects of cmpu technological change begin to shift into features and capabilities that are opaque to CMPUint/fp, then enhancements to this measure of quality change can be explored.

Appendix A
Price Indexes for Intel MPUs (1-97 thru 11-99)

Three price indexes were constructed for cmpus based on the same secondary data sources currently used as a supplement to track prices in the PPI. Note that the selection of specific cmpus for these indexes differs from those used in the PPI. A start (base) date of 1-97 was chosen to limit, thereby simplifying, price comparisons to three distinct technological generations. The start date also corresponds to the effective date that PPI introduced secondary price data into its microprocessor index. A description of methodology and general conclusions follow.

Table 1*	Direct Link	Explicit QA	Explicit QA Best Match
Jan-97	100.0	100.0	100.0
Feb-97	79.9	79.9	79.9
Mar-97	72.6	72.6	72.6
Apr-97	72.6	72.6	72.6
May-97	65.3	65.3	65.3
Jun-97	65.3	65.3	65.3
Jul-97	65.3	65.3	65.3
Aug-97	41.7	61.8	43.5
Sep-97	41.7	61.8	43.5
Oct-97	41.7	61.8	43.5
Nov-97	35.6	63.8	39.5
Dec-97	35.6	63.8	39.5
Jan-98	33.4	57.4	37.5
Feb-98	27.3	77.4	46.2
Mar-98	27.3	77.4	46.2
Apr-98	26.8	74.3	44.3
May-98	20.3	55.1	33.8
Jun-98	17.5	45.8	29.5
Jul-98	16.1	46.6	28.2
Aug-98	16.1	46.6	28.2
Sep-98	14.6	79.8	32.6
Oct-98	12.2	65.6	27.9
Nov-98	12.2	65.6	27.9
Dec-98	12.2	65.6	27.9
Jan-99	11.1	60.4	25.5
Feb-99	10.4	68.7	27.7
Mar-99	9.4	59.8	25.3
Apr-99	8.0	53.2	22.4
May-99	7.3	45.9	20.4
Jun-99	7.0	44.4	20.0
Jul-99	7.0	44.4	20.0
Aug-99	6.6	42.6	19.3
Sep-99	6.3	47.5	22.0
Oct-99	6.3	47.5	22.0
Nov-99	5.9	45.3	20.8

* The 35 month period covered by the indexes include a range of technologies starting with the Pentium 120 and extending to the Pentium III 550.

Index Construction:

To simplify index construction, all of the items that make up the three indexes were unweighted and based primarily on mainstream cmpus. Each index month includes 7 items that represent a cross-section of the market. For instance, the January 97 indexes include prices for the Pentium 120, 133, 150, 166, 200, 166MMX and 200MMX. All of these products were sold into the mainstream market and represented approximately 75 percent of cmpus revenue in the base period. Three different strategies were employed to deal with rapid obsolescence that is the norm for the computer microprocessor industry. These strategies can be summed up by *direct link*, *explicit QA* and *explicit QA with best match*. In all cases prices were tracked until the cmpu reached obsolescence and then a replacement was selected to maintain index continuity.

The *direct link* index treats nominal price differences between an obsolete product and its replacement as an implicit measure of quality change. In other words, when a replacement is introduced, the price relative is adjusted to show no change. This procedure is the current default for the PPI due to a historic lack of appropriate data to calculate explicit quality valuations for technical change. The direct link index exhibits the largest decline over the almost 3-year period, dropping 94.1 percent.

The *explicit QA* index uses the recently developed CMPUint/fp benchmarks (see table 10) to calculate valuations for changes in the processing capabilities (quality) of replacement cmpus relative to their predecessor. As cmpus became obsolete and required replacement, the closest available technological match that was not currently in the index was introduced. One of the implications for this strategy is random volatility occurs when cmpus at the end of their life cycle are sometimes unavoidably replaced with cmpus that are relatively early in their life cycle. Note that the life of cmpus is about 12 months, sometimes less. See pgs 34-36 for a description of the disproportionate price/performance premium that early life cycle cmpus command relative to those that have reached end-of-life. For index users, the effect is to cause seemingly random upward spikes in the explicit QA index. This index showed an overall decline of 54.7 percent, but on several occasions jumped significantly. For instance, on Feb-98, the index jumped 34.8 percent and on Sep-98, the index skyrocketed 71.2 percent. Several smaller increases were observed on Nov-97, Feb-99 and Sep-99. In all cases, the increases were due to large differences in the nominal prices of obsolete and early life cycle cmpu replacements relative to significantly smaller differences in measures of performance or quality change.

The *explicit QA best match* index uses the same CMPUint/fp benchmarks as the *explicit QA* index, but takes a more aggressive approach for insuring the most appropriate technological match is selected as a replacement for obsolete cmpus. This index moderates much of the random volatility exhibited in the explicit QA index, but still exhibits increases in several months. For instance, on Feb-98 the index moves up 23.2 percent with smaller increases on three other occasions. The overall decline is 79.2 percent compared to 54.7 percent for the *explicit QA* and 94.1 percent for the *direct link* [The PPI's published Microprocessor index which also employs direct link dropped 93.5 percent in a roughly comparable period[59]]. The volatility could be further reduced in the best match index if comparisons are not made for cmpus that cross market boundaries, such as replacing a Pentium II with a Celeron. This latter circumstance was unavoidable and touched on briefly in footnote 56 pg. 42 with the observation that new classes of cmpus may need to be periodically introduced as supplements to the original sample rather than force an inappropriate comparison. Of course, another way to deal with this issue is through the default direct link procedure which will treat all price change as quality change but may also re-introduce another potential source of bias. However, for consistency, direct links were not used in either of the explicit QA indexes. Unless trends undergo a dramatic change, the issue of new market segment cmpus will be rare, even for this dynamic industry.

Conclusions

If the CMPUint/fp benchmarks, or an equivalent, are adopted as a means to isolate pure price change from quality change in the PPI's Microprocessor index, user reaction to this change should be anticipated. I refer to the generally accepted notion that cmpu prices have followed a steady and dramatic downward trend. While it is true that most cmpus are introduced at relatively high prices and then quickly decline, when explicit quality adjustments are available for replacements this notion may be turned upside down. The reality is that when a $800 cmpu replaces an end-of-life $100 cmpu, the $700 price difference cannot be entirely explained by quality change. To do so would deny one of the most fundamental truths of the semiconductor industry, namely Moore's Law. Moore's Law has consistently show an approximate doubling of cmpu performance every 18 months for the last 25 years. This is a remarkable achievement for any industry, however when a replacement cmpu sells at a multiple as high as 8 times the obsolete cmpu there appears to be an additional margin available to the producer for the technical improvement. If this real price increase relative to the predecessor is not shown, then output price measures are unavoidably

---

[59] The three example indexes cover Jan-97 through Nov-99. The PPI MPU index comparison is based on Jan-97 through Oct-99.

downward biased and output quantity measures are unavoidably upward biased.

No matter what methodology is used, price indexes for cmpus will always be subject to the question of; What is the appropriate replacement choice for an obsolete cmpu? This is true for all products, but has particularly strong index effects for the output of high-tech rapidly changing industries. A strong case can be made that a direct link index is downward biased because it can never account for price-performance premiums that have partly characterized the cmpu market for more than 20 years. On the other hand, a quality adjusted index that limits the cmpu replacement candidates to those outside the index will almost certainly encounter inappropriate replacement choices. For instance, the largest increases in the *explicit QA* index were due to having to choose among cmpu replacements that were in the early life, high margin and high performance market segment. These cmpus are not sold or positioned in the market to replace late life, relatively low performance cmpus.   Such comparisons introduce an upward bias because they almost guarantee a large pure price increase, but the comparison is one that should not be made in the first place. The *explicit QA best match* represents a middle ground between the respective upward and downward biases of *explicit QA* and *direct link*. It should be noted, that even with the *explicit QA best match*, counterintuitive increases in the index may occur from time to time, but the long-term movement (as shown in the example) should be a substantial negative AGR.

If the closest match strategy is followed, then occasional increases in the index may simply reflect a dynamic market that is never in equilibrium. Another factor, is that many of the classical assumptions of perfect competition and price behavior do not describe the microprocessor industry. If entry barriers (such as very large capital requirements) enable producers to adopt a pricing strategy that maximizes profits but not necessarily output, then additional complexity is introduced into constant quality measures in an output index. To the extent that users of the PPI's Microprocessor index view the market as a series of new technology introductions followed by rapid price declines they are likely to be caught off-guard by occasional increases in a index that are due to estimated values of a rapidly changing price-performance surface. It may be that any methodology designed to isolate pure cmpu price change that generates index movement that challenges long-held anecdotal assumptions is initially likely to be a ripe target for criticism.

As a final note, weighing issues have been ignored. To the extent that both early life and end of life cmpus carry little weight in the market, much of the price-performance disparity between these two classes of cmpus will be dissipated. However, the assignment of correct item weights is a difficult problem. A new, relatively high-priced cmpu may start off with monthly unit shipments of 50,000 but reach 8-10 million in six months and then zero after 11-14 months. Initial fixed weight assignments that are in turn moved by declining price relatives introduce a different set of issues that may affect index movement for an extremely dynamic market.

References

Berndt, E. R., Griliches, Z., and Rappaport, N. R., "Econometric Estimates of Price Indexes for Personal Computers in the 1990's", Journal of Econometrics, 68, 1995, pp. 243-268.

Cole, R., Chen, Y., Stolleman, J., Dullberger, E., Helvacian, N., and Hodge, J., "Quality-Adjusted Price Indexes for Computer Processors and Selected Peripheral Equipment", Survey of Current Business, January 1986, pp. 41-50.

Diefendorff, K. "PC Processor Microarchitecture, A Concise Review of the Techniques Used in Modern PC Processors", Microprocessor Report, July 12, 1999.

Holdway, M., 1999b, Quality-Adjusting Computer Prices in the Producer Price Index:
An Overview, PPI paper, January 2000.

Grimm, B., "Price Indexes for Selected Semiconductors, 1974-96", Survey of Current Business, February 1998.

Gwennap, L., and Thomsen, M., "Intel Microprocessor Forecast, Product Roadmap, Volumes, Costs, and Prices", 4th Edition, 1998, Publisher, MicroDesign Resources.

Jorgenson, D. and Stiroh, K., 1994, "Computers and Growth", Ecorz, Innovation and New Technologies, 1995, Vol. 3. pp. 295-316.

Sinclair, J., Catron, B., "An Experimental Price Index for the Computer Industry", Monthly Labor Review, Bureau of Labor Statistics, October 1990, pp. 16-24.

Triplett, J., 1983, "Concepts of Quality in Input and Output Price Measures: A Resolution of the User-Value Resource-Cost Debate", in Studies in Income and Wealth , Volume 47, University of Chicago Press for the National Bureau of Economic Research, pp. 296-311.

Triplett, J. "Economic Interpretation of Hedonic Models", Survey of Current Business, January 1986, pp. 36-40.

Triplett, J. "Economic Theory and BEA's Alternative Quantity and Price Indexes", Survey of Current Business, April 1992, pp. 49-52.