

# PREDICTING PROCESSOR PERFORMANCE

G. Kent Webb, San Jose State University, webb\_k@cob.sjsu.edu

## ABSTRACT

*Systems analysts and managers involved in planning for computer system deployments and upgrades often require an estimate of future component performance. With a focus on processors used in desktop systems, this paper examines how long run technical trends, such as expressed in Moore's Law, can be used to predict future computing performance. Historical data on the number of transistors and clock speeds were collected for Intel microprocessors beginning with their introduction in 1971. The data show that Moore's Law significantly overestimates long-term development which has been increasing but at a decreasing rate. Results of experiments with other prediction equations are also presented.*

**Keywords:** Technology Management, Systems Analysis, Moore's Law, Forecasting

## INTRODUCTION

According to an ACM article (4), Moore's Law predicts that microprocessor performance will double every 18 months and that this forecast has been remarkably accurate after 35 years. Similar comments can be found in other articles (5, 6). A review of this literature shows there is considerable confusion about what Moore's Law actually proposes.

A discussion of Moore's Law, with text from the original article, is presented in the next section. This is followed by a description of data spanning more than 40 years for Intel microprocessors that describe two performance measures, number of transistors and clock speed. The data are used to check the accuracy of Moore's Law and then to calibrate an alternative regression model. The form of the model and resulting forecast is then discussed.

### Moore's Law

In predicting the future of the integrated circuit, Gordon Moore, a founder of Intel, (1) published a 1965 article in *Electronics* describing the "costs and curves" that would drive the industry forward. The article establishes the basis for a relatively famous technological relationship, Moore's Law, which has been widely quoted in a variety of forms. A review of the literature reveals many different formulations of Moore's Law (12, 13, 14). For clarification, two key paragraphs from the original article appear below (*italics added*):

Reduced cost is one of the big attractions of integrated electronics, and the cost advantage continues to increase as the technology evolves toward the production of larger and larger circuit functions on a single semiconductor substrate. For simple circuits, the cost per component is nearly inversely proportional to the number of components, the result of the equivalent piece of semiconductor in the equivalent package containing more components. But as components are added,

decreased yields more than compensate for the increased complexity, tending to raise the cost per component. Thus there is a minimum cost at any given time in the evolution of the technology. At present, it is reached when 50 components are used per circuit. But the minimum is rising rapidly while the entire cost curve is falling. If we look ahead five years [to 1970], a plot of costs suggests that the minimum cost per component might be expected in circuits with about 1,000 components per circuit (providing such circuit functions can be produced in moderate quantities.) In 1970, the manufacturing cost per component can be expected to be only a tenth of the present cost.

*The complexity for minimum component costs has increased at a rate of roughly a factor of two per year. Certainly over the short term this rate can be expected to continue, if not to increase. Over the longer term, the rate of increase is a bit more uncertain, although there is no reason to believe it will not remain nearly constant for at least 10 years. That means by 1970, the number of components per integrated circuit for minimum cost will be 65, 000.*

Moore makes a number of prescient technological predictions in this article. Much of the article is devoted to an explanation of why the long run average total cost curve for the integrated circuit industry is steeply downward sloping and that this situation can be expected to persist for many years to come.

His famous law (in italics above), formulated as an equation below, summarized recent experience for a projection up of 10 years beyond 1965. The law proposes that the number of components, specifically transistors, on a circuit may continue to double every year. One of the reasons the law seems so accurate is that for some reason it has been reformulated over the years (2). Instead of doubling every year as first suggested, the doubling went to 18 months and is now at a rate of about every 2 years.

Moore's Law             $C_t = C_0 2^t$   
where,            C are the number of components (transistors) per integrated circuit  
                         t is the time period measured in years for  $t = 0, 1, 2, \dots, n$

Moore has himself acknowledged that a prediction of transistors per circuit on the basis of the 1965 article overestimates the observed values 20 to 40 years later. However, the decline in cost per transistor and the increase in MIPS (million instructions per second) are closer to the exponential pattern described in the original article (10).

## Processor Performance

Moore's Law referred to the number of transistors on a circuit, not the performance of the circuit. Although the two are closely related, other issues such as addressable memory, bus width and speed, and processor architecture are also important.

Clock speed is the most often used general performance measure for desktop computer systems. Systems are typically listed for sale with the processor speed prominently displayed. Obviously this is not the whole story. AMD has taken to listing the equivalent processor speed (the speed that an Intel chip would need to run at in order to be of equal performance). A number of other benchmarks have been developed which attempt to give a more accurate picture of processor and computer system performance. A thorough discussion of the problems of measuring performance and the use of benchmarking is available in an article by Lewis, et al (9). Also, Emma (11) discusses the use of instructions per second or instructions per cycle as a performance measure.

The performance measures maintained by Intel on their company website cataloging the history of microprocessor performance include clock speed, number of transistors per circuit, bus speed, and addressable memory. Some of the landmark events from this history are presented in Table 1 (3).

Table 1: A Brief History of the Intel Microprocessor

Date	Processor Introduced	Clock Speed	Transistors	Significance, Application
Nov, 1971	4004	400 KHz	2,300	Intel's first processor. Calculator, arithmetic manipulation
June, 1978	8086	5 MHz	29,000	First processor for personal computing, established X86 standard architecture
Feb, 1982	80286	6 MHz	134,000	Second generation of X86 standard
Oct, 1985	80386	16 MHz	275,000	Third generation X86
Sept, 1991	80486	25 MHz	900,000	Fourth generation X86
Mar, 1993	Pentium	66 MHz	3.1 Million	Fifth generation X86
May, 1997	Pentium II	300 MHz	7.5 Million	Sixth generation X86
Feb, 1999	Pentium III	500 MHz	9.5 Million	Seventh generation X86
Jan, 2001	Pentium 4	1.5 GHz	42 Million	Eight generation X86
May, 2001	Itanium	800 MHz	25 Million (now 410)	Intel departs from X86 standard. Work Station and server.
Feb, 2004	Latest Pentium 4	3.4 GHz	178 Million	Current fastest desktop processor

## DATA ANALYSIS

### The Data

A time series of monthly data was created from the information maintained on Intel's site. For each month, the data are organized to identify the fastest clock speed available for use in a desktop or equivalent system and the number of transistors per processor. In the early years, the

data includes some processors that were actually used for calculators but that are included by Intel in the processor family.

A cursory examination of the data reveals that the time it takes to double the number of components (transistors) on the circuit has been increasing. Taken from 1971 to 2004, the number of transistors per circuit has doubled about every two years, on average. By comparison, Moore's original extrapolation was for a doubling every year. The rate of change has been increasing at a decreasing rate..

In terms of processor speed, starting from the 1971 processor running at 4 KHz, a doubling every two years would yield a clock speed of 28.6 GHz, substantially above the actual 3.4 GHz. Processor clock speeds have doubled on average every 31 months since 1971; but as with transistors per circuit, clock speeds accelerated faster during the early years, suggesting a functional form for a prediction equation that is consistent with change increasing at a decreasing rate.

### **The Model**

Modeling efforts focused on predicting clock speeds, the most common measure of processor performance. A number of functional forms for a prediction equation were explored using regression analysis. To start, Moore's Law was reformulated in the lagged format described in the equation below. This lagged model is a common first approximation of capturing trend data and is discussed by Vilata (8) as related to predicting computer performance.

$$S_{t+1} = a + bS_t$$

where,

$S_t$  is processor speed at time  $t$  for  $t = 0, 1, 2, 3, 4 \dots n$

$a$  and  $b$  are estimated coefficients

For Moore's Law to be strictly applied to processor speed in this equation, the value for coefficient  $a$ , the intercept, would be 0 and the value for  $b$  would be 2. Based on annual data from 1971 as reported in Table 2, the estimated coefficient for  $a$  in the above equation was about 29 (although it was not statistically significant) and the coefficient for  $b$  was about 1.26. Although the general statistics for this equation, such as  $t$  and R-square, are good; this formulation significantly overstates the processor speeds in the early years of development. This formulation does approximately capture some of the basic features of development: that speeds are increasing at a decreasing rate. This equation forecasts a processor speed of 58.7 GHz in 2014.

Monthly, rather than annual data were used to explore other function forms for the prediction equation since the timing of new product introductions in any year varies quite a bit and often more than one new processor will be introduced in any one year. The other functional forms involved using natural logs and combinations of a time trend variable  $t$ .

The functional form that provided the best overall fit of the data involved transforming the clock speed into the square root of the clock speed. The resulting Equation 2 in Table 2 captures the

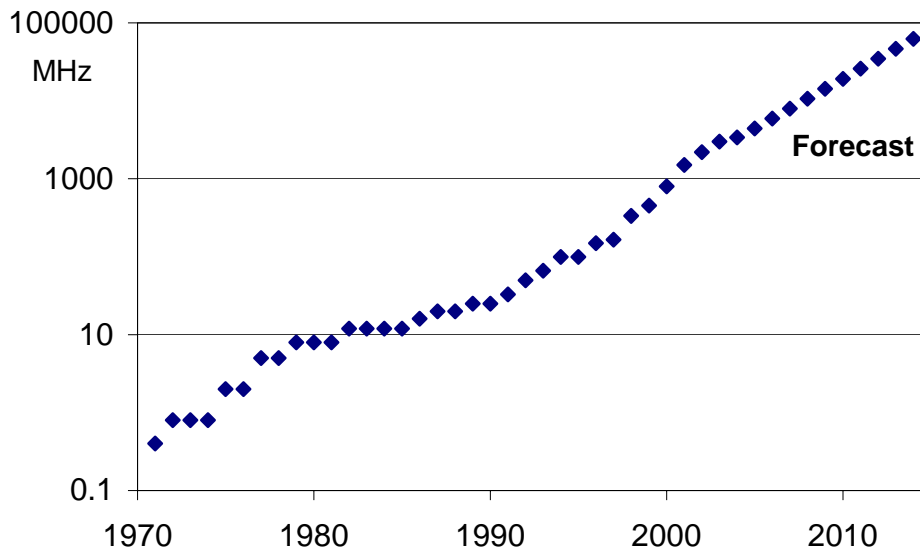
logic that clock speed rates of increases are slowing. Extrapolating out to 2014 with this equation yields a processor speed of 62.3 GHz. The historical data and the forecast from Equation 2 are illustrated in Figure 2.

Table 2: Regression Analysis of Historic Data

Equation 1: Dependent Variable: Next Period Processor Speed

Independent Variables (Annual Data)	Coefficient	t-statistic	P-value
Intercept	29.65493	1.1542	0.22572
Processor Speed for Previous Year	1.2647	35.463	0
R-Square = 0.976			
Equation 2: Dependent Variable: Square Root of Next Period Processor Speed			
Independent Variables (Monthly Data)	Coefficient	t-statistic	P-value
Intercept	0.0132511	0.354	0.723343
Square Root of Processor Speed for Previous Month	1.0123417	490.24	0
R-Square = 0.998			

Figure 1: Processor Speed History and Forecast  
Log Scale



## CONCLUSION

Although clock speeds and transistors per circuit have not kept pace with the original exponential forecast known as Moore's Law, doubling every year, computing performance and cost efficiencies continue to advance at a remarkable pace. Competition among the major processor manufacturers, Intel, AMD, IBM, Sun, and Texas Instruments can be expected to push the industry down the long-run average total cost curves described by Gordon Moore in 1965. As he predicted, the result will be dramatic improvements and much lower prices for computing performance.

While clock speeds may continue to be a standard measure of performance, the development focus has turned to more general measures. In a recent interview, Intel's new Chief Operating Officer, Paul Otellini, said, "Our view has evolved from, 'How do you build a faster microprocessor to, increasingly, looking at everything that goes around microprocessors'" (7). As a result, both of the equations used to extrapolate historical data forward have probably resulted in forecasts for processor speed that are significantly too high.

With the Itanium generation of microprocessors, Intel has abandoned the X86 architecture standard that has driven most of the computer market for the past 20 years. The new architecture, developed with Hewlett-Packard, will substantially differentiate future products from those made by archrival AMD. If Intel succeeds in differentiating its processors from AMD, and given the small and declining share of the market using the Sun and IBM processors, there is a possibility that Intel will have less competitive pressure to advance technology in the future. Another trend that may change the focus from advancing the performance of individual processors is the increasing use of multiprocessor systems and processor arrays.

## REFERENCES

1. Moore, Gordon (1965). Cramming More Components Onto Integrated Circuits, Electronics, (38) 8. (Available at: <http://www.intel.com/research/silicon/moorespaper.pdf>)
2. Webopedia, [http://www.webopedia.com/TERM/M/Moores\\_Law.html](http://www.webopedia.com/TERM/M/Moores_Law.html)
3. Intel Microprocessor Quick Guide, <http://www.intel.com/pressroom/kits/quickrefyr.htm>
4. Anonymous (1998), Promise and Limits of Amdahl's Law and Moore's Law, Communications of the ACM, (Nov, Vol. 41, Issue 11), 38
5. Webb, G. Kent (2003). A Rule Based Forecast of Hard Disk Drive Costs, Issues in Information Systems, 4(1), 337.
6. Wolff, Michael F. (2004). Chase Moore's Law, Inventors Urged, Research Technology Management, (Jan/Feb, Vol. 47, Issue 1), 6
7. Greenemeier, Larry (2004). Intel's Crystal Ball, Information Week, (Issue 977, Feb. 23, 38).
8. Vilalta, Ricardo, Apte, Chidanand V., Hellerstein, Joseph L., Ma, Sheng, Weiss, Sholom M. (2002). Predictive Algorithms in the Management of Computer Systems, IBM Systems Journal, (Vol. 41, Issue 3), 461
9. Lewis, Byron C.; Crews, Albert E (1985). The Evolution of Benchmarking as a Computer Performance Evaluation Technique, MIS Quarterly, (Mar. 85, Vol. 9 Issue 1), 7

10. Moore, Gordon (2003). No Exponential Is Forever...But We Can Delay 'Forever'. International Solid State Circuits Conference. (The slideshows from this presentation are available on Intel's Website at: [ftp://download.intel.com/research/silicon/Gordon\\_Moore\\_ISSCC\\_021003.pdf](ftp://download.intel.com/research/silicon/Gordon_Moore_ISSCC_021003.pdf)).
11. Emma, Philip G. (1997) Understanding Some Simple Processor-Performance Limits, IBM Journal of Research & Development, (May, Vol. 41, Issue 3).
12. Coffee, Peter (2003), Moore's Law Lives On. eWeek, 10/13/2003, (Vol. 20 Issue 41), 42,
13. Fox, Robert (2002) Moore's Law and Cryonic Suspension, Communications of the ACM, (Oct, Vol. 45 Issue 10), 10.
14. Jovanovic, Boyan; Rousseau, Peter L. (2002) Moore's Law and Learning by Doing, Review of Economic Dynamics, (Vol. 5 Issue 2), 346.