

An Ultra-Compact Empirical Model for Throughput Projection for Gigascale Integration

J.C. Eble, S. P. Nugent, J. D. Meindl, D. S. Wills

Delivered instruction throughput, measured in completed instructions per second (IPS), is the most general metric for comparing processor performance. Instruction throughput is the product of cycle time (T_c) and CPI, both of which are processor implementation dependent. Since benchmarks and the instruction set are rarely modified, instruction throughput combines the processor implementation dependent elements of execution time while eliminating application dependency. Thus,

$$IPS = \frac{1}{T_c} CPI \quad (1)$$

Projections of future microprocessor performance have historically centered on clock frequency as the primary performance metric [1,2]. However, clock frequency alone does not fully determine the computational performance of future processor designs. CPI represents the architectural contribution to processor performance. The estimation of clock frequency though challenging is straightforward [2,3]. But CPI is more difficult to determine because of the many architectural parameters (e.g., exploited concurrency, memory latency, and branch prediction accuracy) that determine it. In order to predict the performance of future gigascale processors, a CPI model that forecasts architectural improvements, is needed.

Current efforts to determine the average CPI of microprocessor architectures primarily rely on simulators that run instructions or instruction traces to determine the number of cycles required for a task of known length. The trace size required for a complete benchmark is too large for efficient simulation, so trace sampling techniques are used to compress the trace size without sacrificing accuracy and reduce the simulation time considerably. Even with the speed advantage provided by trace sampling techniques [4], there are serious drawbacks that prevent this method of CPI extraction from becoming a viable alternative for projecting system performance across the span of the ITRS roadmap [1]. The greatest drawback is the complexity of an architectural simulator that demands a high degree of architectural specification. In order to project the throughput of future designs, a CPI model must easily generalize to a large cross section of processor architectures. A different simulator must be implemented for each proposed design before its CPI can be determined. As a result, the architectural performance required to meet ITRS projections has remained largely unexplored.

The logic-memory model is an empirical relationship based on the observation that CPI improving architectural concepts such as speculative execution and dynamic instruction scheduling [5] require hardware support resulting in increased transistor budgets. This approach considers the CPI metric from a global, abstract view. Based on empirical data, the most promising, globally observable factors that can be correlated to CPI increase are identified. The ultimate goal is to establish an empirical relationship between CPI and the characteristics of the chip technology and architecture.

The first correlation parameter is the number of logic or non-cache transistors on the chip. This factor heavily influences the logic component of the CPI. Increasing the transistor count allows more datapaths, functional units, and dynamic scheduling hardware all of which lead to reductions in the CPI of the chip. The second major design parameter is the memory hierarchy comprised of the on-chip cache sizes, organization, and interface to main memory and off-chip caches. These parameters form the memory portion of the CPI model.

The logic-memory model for CPI utilizes a power-law relationship for the logic portion of CPI and memory hierarchy parameters to project the throughput of microprocessors into the next decade [6]. The essence of the memory-logic model is presented below in (2).

$$CPI = E_c N_{logic}^{E_e} + M_{refs} M_{rate} M_{penalty} \quad (2)$$

The parameters E_c and E_e are the empirical throughput coefficient and exponent, respectively of the proposed power-law relationship [11]. These parameters capture the history of how effective logic transistors have been in reducing logic CPI. N_{logic} represents the number of logic transistors or gates in the chip.

The second term of (3) is the memory contribution to the CPI. An instruction cannot execute and complete until all required data is available, so a miss in the cache results in access to the slower main memory. The result is that the instruction stalls until the data access and retrieval is complete; therefore, an

instruction that may nominally take a few cycles to complete could take dozens of cycles. This problem is exacerbated as the gap between main memory access time and cycle time increases [7]. M_{refs} is a workload characteristic representing the number of memory references per instruction or reference frequency, M_{rate} is the percentage of memory references resulting in a cache miss, and M_{penalty} is the total time (in cycles) taken to access and transfer data from the main memory. The product of these three factors represents the increase in CPI due to cache misses.

The extraction of E_c and E_e for use in predicting throughput requires historical data on CPI for a microprocessor family and determination of the memory portion of the CPI. CPI is not usually a published figure-of-merit, but it can be estimated by the quotient of performance in *instructions* per second and clock frequency (Eq. 1). The standard measure of performance is the Standard Performance Evaluation Corporation (SPEC) suite of benchmarks in which the execution times on a number of programs are normalized to a baseline machine [8]. If this is taken as a measure of throughput or performance the IPC (1/CPI) is obtained.

$$SPEC \approx \frac{Instr}{s} \Rightarrow \frac{SPEC}{f_c} \approx \frac{Instr}{s} \frac{s}{cycle} = \frac{Instr}{cycle} = IPC \quad (3)$$

The CPI is calculated from the following formula:

$$CPI = \frac{f_c}{k \cdot SPEC} \quad (4)$$

Where k is a scaling factor used to calibrate the above expression to known CPI values that have been published in various studies across different processor families.

Once the relative CPI is known for different generations of a processor family, the memory portion of the CPI must be calculated. This requires information concerning the main memory subsystem including the external bus width (b_{bus}), bus frequency (f_{bus}), and main memory access time (t_{mm}). These parameters capture the effects of off-chip references to the main memory. The frequency of these off-chip references is determined by the reference frequency, M_{refs} . Values for the instruction/data references reported by Flynn [9] are used for this analysis. The method used to calculate the miss rate, M_{rate} , of the cache is the Design Target Miss Rate (DTMR) tables first introduced by Smith [10,11] and updated by Flynn [9]. This model of miss rate is used and is based on actual simulation and provides a simple way of determining the miss rate for a representative workload without introducing additional parameters.

Once the historical CPI data has been collected and the cache CPI portion for each point is calculated, the logic CPI is determined by subtracting the memory contribution. The remaining quantity is the “base CPI”:

$$(CPI)_{\text{base}} = E_c(N_{\text{logic}})^{E_e} = CPI - M_{\text{ref}}M_{\text{rate}}M_{\text{penalty}} \quad (5)$$

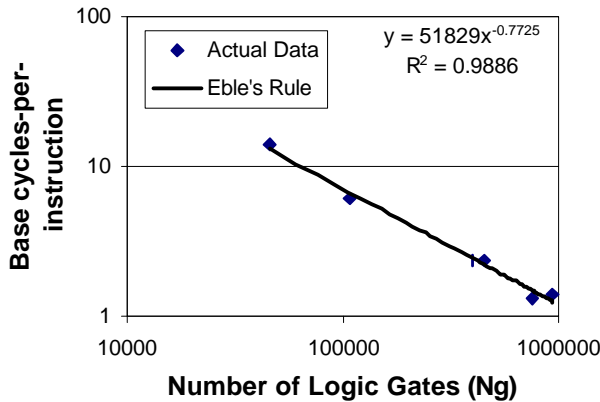


Figure 3: Validation of logic CPI model
 $(CPI)_{\text{base}} = E_c(N_{\text{logic}})^{E_e} = 51829(N_{\text{logic}})^{-0.7725}$

The throughput coefficient E_c and exponent E_e are determined by plotting the base CPI versus the number of logic gates.

A sample study of the Intel x86 processor family over a 12 year period is presented in Figure 1. This figure plots the power-law relationship for base CPI or $(CPI)_{\text{base}}$. This complete logic-memory model does an excellent job of predicting logic CPI and explains over 98% of the variation. The model also adequately predicts the CPI contribution from the memory hierarchy. Actual data indicates that the percentage of CPI due to the memory subsystem ranges from just a couple of percent to 40% across an application suite. The logic-memory CPI model predicts the memory portion to account

for 34% of the Pentium's total CPI, which falls within the range of the reported data.

The projected throughput of future microprocessors was evaluated with the Generic System Simulator, GENESYS, a tool that engages a hierarchical set of analytical models based on physical principles and established empirical knowledge to calculate key performance metrics for microprocessors [12]. Utilizing the technology and integration levels indicated by the ITRS [1], the maximum throughputs for three architectures families, Intel/AMD x86 and DEC Alpha, are projected through the year 2011 by 1) extracting the empirical throughput parameters and *assuming a constant logic depth* characteristic of the family, 2) optimizing each level of the multi-level interconnect architecture, 3) inserting repeaters in long interconnects, 4) finding the cache size that maximizes throughput for a specified total transistor count and allowable die area, and 5) assuming a single zone of synchrony that indicates a single global clock frequency. Figure 1 plots the throughput versus calendar year for all three processor families. The logic depths are grossly estimated and the Rent's exponents affecting the interconnect distribution are constant between the two, so the specifications are merely a rough approximation of these three commercial

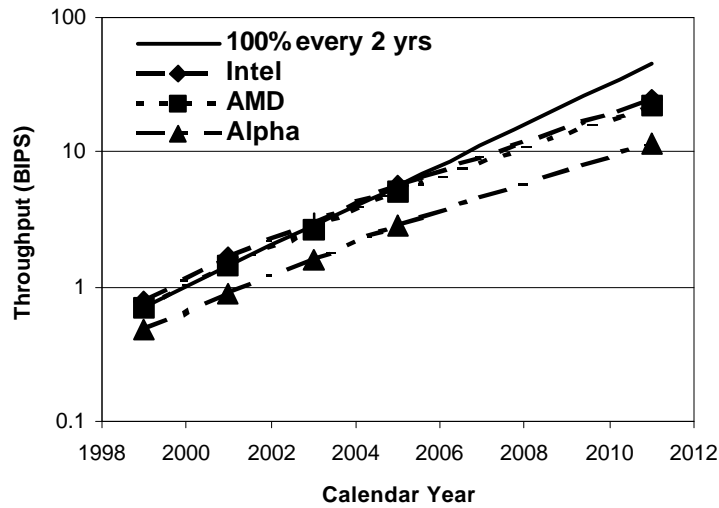


Figure 2: Throughput VS Calendar Year

processors. In Figure 2 the projected throughput for each is benchmarked against an exponential type performance growth that assumes a doubling of performance every 24 months. Beyond 2003 both architectures fail to maintain the historical rate of growth. At the end of the roadmap, the performance reaches only 20-40% of that projected by the current growth trends! The cause of this degradation in the performance trend can be determined by examining the three components of the throughput projections; logic, memory, and clock frequency. The clock frequency projection for this plot are taken directly from the ITRS and a log-linear plot of frequency VS calendar year demonstrates a constant slope, so the clock frequency is not the source of degradation. The form of the empirical logic CPI model (a power-law expression) forces a linear trend on a log plot, so it cannot be the cause. An inspection of the memory CPI however, reveals that the memory contribution to the total CPI does not exhibit a linear trend as is illustrated in Figure 3. The initial sharp decrease in the memory CPI is not sustained beyond 2000. This is because that large scale changes to the cache and main memory systems such as larger cache sizes, faster memory, larger/faster buses are required to directly impact the CPI and such changes occur at an historically slower pace. The result of this trend is that the memory interface will become an increasingly vital component in evaluating efforts to increasing microprocessor performance. In order to maintain the 2x/24mth performance improvements beyond 2003 the logic component of the CPI will have to scale at a rate faster than is projected from the historical trend. It must be noted that major improvements to memory technology which improve upon the projected memory CPI will combat this trend degradation at its source.

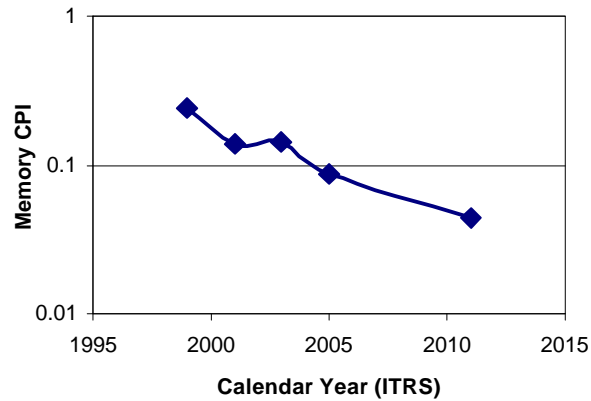


Figure 3: Memory CPI VS Calendar Year

These results were arrived at under the assumption of a single global clock. Both the NTRS and ITRS recognize that across chip communication poses a significant barrier to achieving multi-gigahertz clock frequencies and project both global and local clock frequencies for future technology generations. This suggests a significant departure from standard microprocessor architectures. Such an arrangement will encourage systems in which most computation is done locally (avoiding the long-interconnect problem) at a much faster rate.

In this paper, we propose an ultra-compact empirically based CPI model for projecting the throughput of future gigascale systems. The model evaluates CPI contributions from both a random logic network and the memory hierarchy. The resulting model agrees well with existing data while providing insight into the effects of the memory-logic interaction on the CPI. GENESYS [6,12] utilizes the logic memory throughput model to project the performance of two architectural families across the ITRS roadmap years from 1999 to 2011. The results of this analysis indicate that uniprocessor architectures cannot maintain the current performance growth rate. The advent of new architectures that stress the use of global and local clock frequencies to improve upon the historical rate of CPI reduction will be required to compensate for the effects of a slower improving memory interface.

- [1] Semiconductor Industry Association, *The International Technology Roadmap for Semiconductors*, 1999 edition.
- [2] G. A. Sai-Halasz, "Performance trends in high-end processors," *Proceedings of the IEEE*, vol. 83, pp. 20-36, Jan. 1995.
- [3] H. B. Bakoglu, *Circuits, Interconnections and Packaging for VLSI*, first edition. Reading, MA: Addison Wesley, ch 5.3, 1990.
- [4] J. L. Gustafson, R. Todi, "Convention benchmarks as a sample of the performance spectrum," *Journal of Supercomputing*, vol. 13, no. 3, pp. 321-42, May 1999.
- [5] Hennessy and Patterson, *Computer Architecture: A Quantitative Approach*, second edition, ch 4, San Mateo, Calif.: Morgan Kaufmann, 1996.
- [6] J. C. Eble, *A Generic System Simulator with Novel On-Chip Cache and Throughput Models for Gigascale Integration*, PhD. Thesis, Georgia Institute of Technology, November 1998.
- [7] See 5, pp. 374.
- [8] T. Yager, "Bringing Benchmarks up to SPEC," *BYTE*, vol. 21, no. 3, pp. 145-6, March 1996.
- [9] M. J. Flynn, *Computer Architecture: Pipelined and Parallel Processor Design*, first edition. Boston, MA: Jones and Bartlett, 1995.
- [10] A. J. Smith, "Cache evaluation and the impact of workload choice," *Proceedings of the Twelfth Intl. Symposium on Computer Architecture*, pp. 64-73, June 1985.
- [11] A. J. Smith, "Line (block) size choices for CPU cache memories," *IEEE Transactions on Computers*, vol. C-36, no. 9, pp. 1063-1075, Sept. 1987.
- [12] J. C. Eble, V. K. De, D. S. Wills, J. D. Meindl, "A Generic System Simulator (GENESYS) for ASIC Technology and Architecture Beyond 2001," *Proceedings of the Ninth Annual IEEE International ASIC Conference*, pp. 193-196, Rochester, NY, Sep. 1996.