Compact Models for Estimating Microprocessor Frequency and Power

William Athas
Apple Computer
Cupertino, CA
athas@apple.com

Lynn Youngs
Apple Computer
Cupertino, CA
lyoungs@apple.com

Andrew Reinhart
Motorola
Austin, TX
r18321@email.sps.mot.com

ABSTRACT

This paper describes compact mathematical models for estimating the frequency performance and power dissipation of a microprocessor as a function of the supply voltage. The objective is to estimate the frequency and/or power performance across a wide range of supply voltages and operating frequencies using only a small number of configurable parameters and equations. These compact equations are amenable to hand calculations and spreadsheet manipulation. The configurable parameters are derived from actual measurements of microprocessor chips and are calculated using the least-squares curve-fitting method.

Categories and Subject Descriptors

C.4 [**Performance of Systems**], B.7 [**Integrated Circuits**], I.6 [**Simulation and Modeling**], G.4 [**Mathematical Software**], J.6 [**Computer-Aided Engineering**]

General Terms

Algorithms, Design, Experimentation, Performance

Keywords

Low-power, microprocessors, VLSI, ASIC, curve-fitting, delay modeling, power estimation

1. INTRODUCTION

The suitability of a microprocessor for applications in portable computing requires that it meet specific computational throughput levels at acceptable power levels. The microprocessor power dissipation has a direct impact on the battery life, size, and weight of the portable system. In this work we use a physical interpretation of the charges and currents for the individual transistors of a CMOS microprocessor to derive models for maximum frequency and power. The models take into account issues such as leakage currents and short-circuit currents. The overall behavior of the chip is extrapolated from the specific characteristics of the individual devices as they cycle charge between the capacitive circuit nodes and power rails.

For all regions of interest in the behavior of the devices a detailed physical interpretation would be infeasible to model simply. Furthermore, the estimation of frequency performance and power dissipation is not markedly improved using exact models. Instead, we use a combination of charge-based models and empirical modeling techniques to approximate device behavior across the regions of interest. The results of the modeling are equations that define the maximum operating frequency as a function of supply voltage and the power dissipation as a function of supply voltage and frequency. Together these equations describe the overall shape of the possible operating space for the microprocessor in terms of frequency performance versus power dissipation at different supply voltages.

Properties such as capacitance, circuit design techniques and styles, microarchitecture organization and pipeline design, transistor gain, and circuit activity factors are lumped into coefficients that either isolate a particular circuit's behavior or amortize all of the circuit behaviors across the entire chip. These coefficients are derived from taking measurements of sample chips while running targeted test programs and then applying the linear curve-fitting method to derive the coefficients.

A direct consequence of this approach is that the coefficients, when considered as distinctly separate quantities, can possibly offer insight into the performance of the CMOS process, the circuit techniques that were used for the implementation, and the nature of the underlying microarchitecture.

2. ESTIMATING FREQUENCY FROM SUPPLY VOLTAGE

In searching for the maximum clock frequency of a processor at a given voltage and temperature there will be one path that ultimately limits the frequency because of its delay or noise susceptibility. The basis of the estimation technique for frequency is to model that path at the device level as either charging or discharging a capacitance to either the power-supply rail or to the ground rail. The limiting path may come from either a silicon load or wire path. The delay of the basic device for the speed-limiting path is the ratio of the controlled charge to the controlling current,

$$delay = \frac{charge}{current}. \quad (1)$$

The controlled charge is simply $C \cdot V_{dd}$ where C is the lumped capacitance attached to the drain of the transistor. This capacitance is a combination of wire capacitance, parasitic capacitance, and gate capacitance.

For the controlling current we use the Newton-Sakurai analysis approach[2] to model source-drain current as a function of gate voltage and threshold voltage (V_{th}).

$$current = \beta(V_{dd} - V_{th})^{\alpha} \quad (2)$$

Gate voltage is taken to be the full supply voltage (V_{dd}). The factor β models the transconductance of the transistor. A difficulty in estimating current with a single equation is the different regions of operation in which the current changes from an exponential to lin-

ISLPED'02, August 12-14, 2002, Monterey, California, USA.

ear to quadratic dependence on the gate voltage. However, since we are only interested in the total charge transfer, we can combine the net effect into the parameter α. A simple interpretation is that "α equals one" corresponds to the linear or triode region and "α equals two" corresponds to the satuation region.

Delay is then expressed entirely in terms of supply and threshold voltage, output capacitance, transistor transconductance, and α.

$$delay = \frac{CV_{dd}}{\beta(V_{dd} - V_{th})^{\alpha}} \quad (3)$$

For the case of α equals two, Equation 3 is the same as the first-order model approximation developed by Chandrakasan[3]. With the formulation of Equation 3 we treat this factor as a configurable parameter which will later be used in the curve-fitting process to empirically compensate for device effects that are not explicitly modeled.

We take the reciprocal of this Equation 3 and combine β/C into a single fitting parameter (K_f). For consistency in notation we rename α to K_{ds} ("ds" for device saturation).

$$f = \frac{1}{delay} = K_f \frac{(V_{dd} - V_{th})^{K_{ds}}}{V_{dd}} \quad (4)$$

2.1: Finding values for the fitting parameters

To find values for the fitting parameters, we need a representative set of voltage and frequency data points for a microprocessor at a constant temperature. These points can be obtained from testing a large sample of microprocessors or from measuring a single microprocessor that has been defined to be a "typical" or baseline part. Equation 4 is transformed into a linear equation through a series of algebraic steps followed by taking the logarithm of the resulting equation.

$$\log(f \cdot V_{dd}) = K_f + K_{ds} \cdot \log(V_{dd} - V_{th}) \quad (5)$$

Equation 5 can be input to a straight-line least-squares fit except for the subtraction of V_{dd}-V_{th}. One approach would be to use a non-linear curve fitting technique to approximate the voltage offset due to the threshold voltage. However, since the range of realistic threshold voltages is small, we can exhaustively search for the best fit. For example, a range of 100mV to 1.5V with a resolution of 10mV would require 141 iterations.

To evaluate the accuracy of the model we compare data for three PowerPC™ processors from three generations of VLSI microarchitectures and CMOS fabrication technologies:

CPU94: a 0.50µm dual-issue design with 4 pipeline stages[4],

CPU99: a 0.20µm triple-issue design with 4 pipeline stages and two vector execution units[5], and,

CPU01: a 0.165µm quad-issue design with 7 pipeline stages.

The results are summarized in Table 1. The metric cited for the goodness of fit, *R*, is Pearson's product momentum correlation coefficient[1].

TABLE 1. Microprocessor Frequency Estimation Comparison

CPU94				CPU99				CPU01			
V_{th}= 1.00, K_f=0.123, K_{ds}=1.2912 *R*=0.9963				V_{th}=0.99, K_f=0.947, K_{ds}=0.9129, *R*=0.9972				V_{th}=0.83, K_f=1.444, K_{ds}s=0.8813 *R*=0.9998			
Vdd	**F/Fmax**	**Est**	**Error**	**Vdd**	**F/Fmax**	**Est**	**Error**	**Vdd**	**F/Fmax**	**Est**	**Error**
2.50	0.672	0.681	1.3%	1.70	0.794	0.799	0.6%	1.30	0.798	0.798	-0.1%
2.60	0.721	0.711	-1.4%	1.75	0.833	0.826	-0.9%	1.35	0.838	0.840	0.1%
2.70	0.738	0.741	0.4%	1.80	0.853	0.851	-0.3%	1.40	0.877	0.878	0.1%
2.80	0.754	0.769	2.0%	1.85	0.873	0.874	0.2%	1.45	0.913	0.913	-0.1%
2.90	0.803	0.796	-0.9%	1.90	0.892	0.896	0.5%	1.50	0.946	0.945	-0.1%
3.00	0.836	0.822	-1.6%	1.95	0.912	0.917	0.6%	1.55	0.974	0.974	0.0%
3.10	0.852	0.848	-0.6%	2.00	0.941	0.937	-0.5%	1.60	1.000	1.001	0.1%
3.20	0.885	0.872	-1.5%	2.05	0.951	0.955	0.4%				
3.30	0.902	0.896	-0.7%	2.10	0.980	0.972	-0.8%				
3.40	0.918	0.918	0.0%	2.15	0.990	0.989	-0.2%				
3.50	0.934	0.940	0.6%	2.20	1.000	1.004	0.4%				
3.60	0.951	0.962	1.1%								
3.70	0.984	0.982	-0.1%								
3.80	1.000	1.003	0.3%								

The results in Table 1 indicate this method can model the full data set with a high degree of accuracy. To evaluate the predictive accuracy of this model, the same curve fitting technique was applied to CPU99 using only three data points, and results compared against the full data set. The comparison is summarized in Table 2. The most accurate prediction is obtained when measurements are from the low, middle and high voltage points. The least accurate prediction occurs when all of the points are close together from the middle-most voltage points. Measurements from the low end sacrifice accuracy at the high end and vice versa.

The trends in values for the configurable parameters indicates that over time there are reductions in threshold voltage (V_{th}) and improvements in transistor technology and microarchitecture (K_f). The values for the threshold voltages are higher than would be nominally expected for native transistors. This discrepancy can be explained by the presence of circuits with unrestored pass gates driving the gates of other transistors. The net effect of these circuit structures would be to incur one threshold drop from the drive point to the point where the signal is used, and a second voltage drop in accordance with Equation 4. When fitting the parameters using the method, the threshold voltage would appear to be twice its intrinsic value.

The trend in K_f reflects the significant benefits of smaller feature sizes and re-organization of the microarchitecture into longer pipelines with less logic per pipeline stage. As a predictive tool, the model can be used to predict the voltage versus frequency performance of future microprocessors by adjusting K_f to account for increases in transistor gain and reductions in capacitance due to smaller feature sizes, and for improvements to the circuit structures and pipeline of the microarchitecture using, for example, a fan-out-four (FO4) performance metric[7].

The configurable parameter K_{ds} decreases slightly across the three generations of processors. One problem is that values for K_{ds} ordinarily ranges between one (linear or triode region) and two (fully saturated). An explanation for why K_{ds} is less than one value is that the simplified model of Equation 4 does not include *velocity saturation*. Consequently the curve fitting method compensates by reducing K_{ds} to a value less than we would otherwise expect from the physical nature of the devices.

3. ESTIMATING CORE POWER FROM FREQUENCY AND VOLTAGE

Deriving frequency from voltage estimates the maximum frequency at which the processor can run reliably based on a set of measured parts at a known temperature. We then seek to estimate power at the maximum frequency point for the given supply voltage or at a lower frequency for same supply voltage.

To develop a model for power we limit our investigation to core power. The power dissipated by the I/Os depends on the packaging of the chip, the wiring substrate, and circuits that directly interface to the microprocessor outputs. Furthermore, special circuits and voltage levels are often used for providing high bandwidth off-chip signaling. These differences would severely limit the ability to develop generic compact models.

The method used to estimate frequency from voltage was to model the speed limiting path in terms of a controlled charge and controlling current. With power estimation the idea is to generate the overall maximum amount of internal switching activity inside the chip, sometimes to referred to as a “smoke” test, and then to estimate the charge flow from the power supply rail to the ground rail due to switching and leakage currents.

Developing the smoke test to maximize switching activity concurrently in the data paths and caches is a difficult task. For example, long wires in the caches and datapaths are large contributors to power dissipation. Maximizing switching activity in the datapaths implies that the instructions and data are in the caches. Maximizing cache activity implies that the caches are responding to misses and thus the datapaths are stalled waiting for data. Maximizing activity in both requires detailed knowledge about the pipeline behavior, instruction scheduling, and the interactions between the different sub-systems inside the chip.

The smoke test provides an upper bound for maximum power dissipation since it encompasses all on-chip resources, excluding the I/Os. Real-world applications would typically not generate as much internal activity as the smoke test. Furthermore, it is straightforward to correlate smoke power to a lower level for typical workloads or important applications. To use a power program less stressful than a smoke test could produce misleading results since other test programs might not exercise subsystems which are significant contributors to the power dissipation and which would then be missed in the modeling.

TABLE 2. Comparison of original data to frequency model using different subsets of measurements

original data		**all voltages**		**1.70V, 1.95V, 2.20V**		**1.70V, 1.75V, 1.80V**		**1.90V, 1.95V, 2.00V**		**2.10V, 2.15V, 2.20V**	
Vdd	F/Fmax	Est	Error	Est	Error	Est	Error	Est	Error	Est	Error
1.70	0.794	0.799	0.6%	0.794	0.0%	0.797	0.2%	0.791	0.9%	0.894	-11.9%
1.75	0.833	0.826	-0.9%	0.820	1.6%	0.828	-0.2%	0.816	1.2%	0.905	-9.6%
1.80	0.853	0.851	-0.3%	0.845	0.9%	0.856	-0.7%	0.840	1.2%	0.917	-7.7%
1.85	0.873	0.874	0.2%	0.869	0.5%	0.883	-1.0%	0.865	1.1%	0.928	-6.1%
1.90	0.892	0.896	0.5%	0.891	0.2%	0.909	-1.4%	0.889	0.8%	0.939	-4.7%
1.95	0.912	0.917	0.6%	0.911	0.0%	0.933	-1.7%	0.914	0.4%	0.949	-3.5%
2.00	0.941	0.937	-0.5%	0.931	1.1%	0.956	-2.1%	0.938	-0.2%	0.960	-2.5%
2.05	0.951	0.955	0.4%	0.950	0.2%	0.978	-2.4%	0.963	-0.8%	0.970	-1.6%
2.10	0.980	0.972	-0.8%	0.967	1.4%	0.998	-2.7%	0.987	-1.5%	0.980	-0.8%
2.15	0.990	0.989	-0.2%	0.984	0.7%	1.018	-2.9%	1.011	-2.3%	0.990	-0.1%
2.20	1.000	1.004	0.4%	1.000	0.0%	1.036	-3.2%	1.035	-3.1%	1.000	0.5%

Our method starts with the linear relationship between frequency (F) and current (I) while the supply voltage is held constant,

$$I = I_{dc} + F \cdot Q_{ac} \quad . \tag{6}$$

The frequency-dependent component to current (Q_{ac}) models the average amount of charge that cycles between the capacitive circuit nodes and voltage supply rail or ground rail as the processor performs a computation. The constant component (I_{dc}) models the frequency-independent leakage component. By measuring current at different frequency points, the current-measurement points should comprise a straight line. The slope of the line estimates the dynamic current and the y-intercept estimates the leakage current. Note that the derived values for I_{dc} and Q_{ac} can be expected to vary significantly when the supply voltage or temperature is varied.

For a single supply voltage, V_i, it is straightforward to find the leakage current component and charge component from the least-squares fitting method.

$$I(V_i) = I_{dc}(V_i) + F \cdot Q_{ac}(V_i) \tag{7}$$

Applying Equation 7 to a set of frequency and voltage points creates a pair of vectors for I_{dc} and Q_{ac} for different supply-voltage points. Since I_{dc} is due in part to sub-threshold conduction, this quantity varies exponentially with supply voltage[6]. Thus we can use the set of values for I_{dc} from curve-fitting Equation 7 at different voltages V_i and fit the resulting set of I_{dc} values to an exponential curve,

$$I_{dc}(V_i) = K_{sub1} \cdot e^{K_{sub2} \cdot V_i} \quad . \tag{8}$$

Equation 8 is transformed into a linear equation by taking the logarithm of both sides of the equation:

$$\log(I_{dc}) = K_{sub1} + K_{sub2} \cdot V \, . \tag{9}$$

Estimating Q_{ac} presents more challenges. To a first approximation, this quantity can be modeled as the product of the effective charged-capacitance and supply voltage (CV_{dd}). This component to the dynamic charge is modeled as the supply voltage times a constant (K_d). A second contributor is the short-circuit or "crowbar" current that flows directly from the supply rail to ground when both the pull-up and pull-down devices are active. This is a function of the supply voltage, and, to a smaller degree the threshold voltage. As the supply voltage decreases the short-circuit component becomes increasingly small and practically vanishes at twice the threshold voltage.

The objective is to model the amount of charge that flows during the interval when both the n-channel and p-channel devices are turned on. Proceeding from the analysis done by Weste and Eshragian[5], the short-circuit power is:

$$P_{sc} = \frac{\beta}{12}(V_{DD} - 2V_{th})^3 \frac{t_{rf}}{t_p} \tag{10}$$

Dividing by V_i to get current and using $1/t_p$ for frequency, the model for short-circuit charge is:

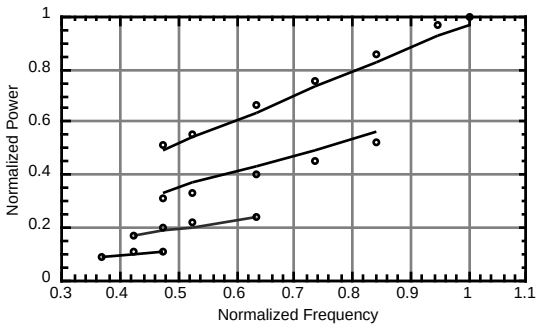
$$Q_{sc} \propto V_i^2 \left(1 - \frac{2V_{th}}{V_i}\right)^3 = \frac{(V_i - 2V_{th})^3}{V_i} \, . \tag{11}$$

After taking many measurements of different parts we found, however, that the curve-fitting method produced equally good results using the following simplification:

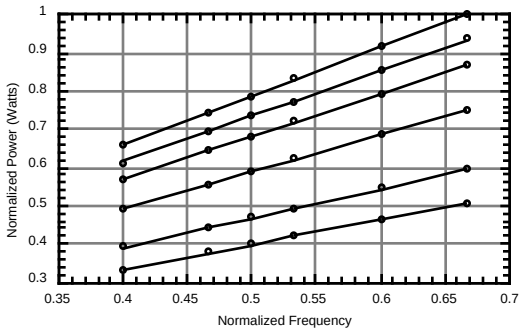
$$\frac{(V_i - 2V_{th})^3}{V_i} \cong V_i^2 \tag{12}$$

The complete equation for I_{ac} is

$$I_{ac}(V_i) = F[K_d + K_{sc} V_i^2] \quad . \tag{13}$$



(a)



(b)

Figure 1. (a) Power model versus measurement for CPU99 (a) and CPU01 (b),

We can use the least-squares method to find coefficients for I_{ac} by algebraic manipulation and substitution:

$$\frac{I_{ac}}{F} = K_d + K_{sc} X_i \qquad X_i = V_i^2 \quad . \tag{14}$$

Figure 1 plots the results of modeling the CPU99 and CPU01 processors using this power estimation method. For CPU01 we had access to a 36-element matrix of smoke power measurements comprised of 6 frequency points and 6 voltage settings using a smoke power test. The results shown in Figure 1(b) demonstrate an excellent fit between the model and the measurements.

For CPU99 we started with a sparse matrix of 9 frequency points and four voltage setting. The minimum number of frequency points per voltage setting was three and the maximum was seven. Power was measured from running typical programs. The results are shown in Figure 1(a). The CPU01 case represents the ideal case and the results are extremely accurate. The CPU99 case is typical of a less controlled set of measurements with corresponding loss of accuracy in the modeling results.

To test the predictive power of the model we conducted an experiment similar to the one used for the frequency model. In Table 3 we show the results of applying the model to only nine points of the original 36 points in the CPU01 matrix. For frequency we used the high, low, and a mid frequency and likewise for voltage. From those nine points we then estimated what the power would be for the other 27 points.

4. POWER ENVELOPES

The two methods for modeling frequency and power can be combined in a single representation for frequency, power, and voltage called a *power envelope*. From voltage we can predict frequency and from voltage and frequency we can estimate power.

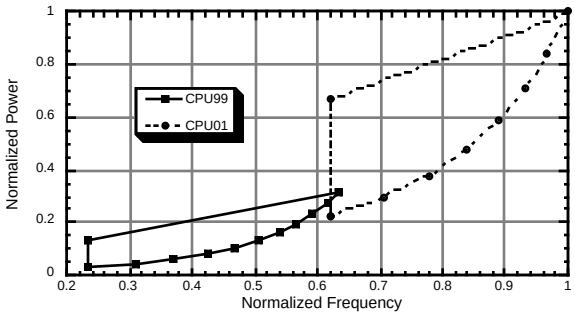


Figure 2. Power envelopes for CPU99 and CPU01

Thus for a given frequency point we can find the minimum required voltage and the consequent power level. For the same frequency, however, we could run at a higher voltage and power level, up to the supply-voltage limit for the CMOS process.

Figure 2 diagrams the power envelopes for CPU99 and CPU01. The bottom edge of each defines the most energy-efficient mode for the processor at each frequency point. For a given frequency we can find the voltage from Equation 4 and then use that voltage plus the frequency to estimate the power level from Equation 13. We can further use Equation 13 for every power level at that frequency above the minimum supply-voltage up to the maximum supply-voltage allowed by the CMOS process.

The absolute maximum frequency is estimated by the speed model from the maximum allowable supply voltage. The minimum frequency is determined by system and circuit factors, e.g., charge loss in unrestricted dynamic circuits or phase-lock-loop range tracking limitations. For each voltage and frequency point within the range the microprocessor will dissipate a power level which resides within the power envelope and can be estimated from the power model.

Using frequency as the performance metric, the power envelopes clearly and concisely demonstrate the power. benefits from improvements in CMOS process technology, new circuit styles, and microarchitecture innovations. For both CPU99 and CPU01, increasing performance through frequency and voltage near the upper end of each power envelope comes at a very high power cost.

5. SUMMARY

In this paper we have presented compact models for estimating and predicting frequency and power for microprocessors as a function of supply voltage. The models were applied to complex state-of-the-art microprocessors but the techniques presented may also be applied to ASICs and all types of synchronous digital VLSI systems.

The models are useful for estimating and predicting frequency and power with high accuracy across a wide range of supply voltages and operating frequencies. Only a few measurement points are required to achieve accuracy to within a few percent.

There are some limitations to the modeling approach due to the simplifications made to achieve compact representations. The most significant limitations are the lack of a temperature parameter, and the neglecting of the physical effects of velocity saturation and gate current currents.

The role of velocity saturation demonstrates an important principle in the balance of maintaining a consistent physical interpretation versus the goal of achieving the best possible curve fit. From curve fitting to Equation 5 the effect of velocity saturation is compensated for by adjustments of the other parameters. This compensation occurs automatically as part of the curve-fitting method. We could, explicitly introduce a new configurable parameter, K_{vs}, and an additional component to Equation 4 to account for velocity saturation.

$$f = K_f \frac{(V_{dd} - V_{th})^{K_{ds}}}{V_{dd}} \left(2 - \frac{V_{dd}}{K_{vs}}\right) \frac{V_{dd}}{K_{vs}} \qquad (15)$$

Curve fitting to Equation 15 produces better estimates for all of the test cases we have tried but the configurable parameters take on physically impossible or inconsistent values. This example illustrates the well-known adage from statistics the value of being “approximately correct” versus “precisely wrong.”

We have done preliminary work to model the effect of temperature on frequency by using an approach similar to the two-step approach that was used to model power. Since the relationship between temperature and mobility is approximately a three-halves power[8] we perform a series of frequency curve fits at different temperatures and then curve fit K_f, and, if necessary the other parameters to temperature. The initial results are promising and the major impediment has been insufficient measurement data to evaluate the accuracy.

Temperature is more problematic for power because of the exponential relationship between temperature and sub-threshold leakage current[6]. Modeling temperature for power would require a three-step curve fitting procedure. However, due to ever thinner gate oxides, significantly more of the static current will be due to gate-leakage current which only depends very weakly on temperature.

TABLE 3. Error for each power point using only three 9 of the original 36 points

F/Fmax	1.40V	1.50V	1.60V	1.70V	1.80V	1.90V
0.600	0.17%	-0.24%	-0.25%	0.18%	0.25%	0.05%
0.699	-0.84%	-0.37%	-0.39%	0.18%	-0.25%	0.34%
0.750	-0.56%	-0.26%	0.08%	0.03%	0.00%	0.35%
0.799	-0.47%	-0.32%	-0.34%	-0.26%	0.06%	-0.32%
0.900	-0.16%	-0.28%	-0.30%	-0.08%	-0.19%	-0.38%
1.000	0.09%	-0.24%	-0.27%	0.07%	-0.39%	0.00%

6. ACKNOWLEDGMENTS

The authors thank Allan Ovrom, Bob Mansfield, and Michael Johnson for their comments, advice, and discussions, Eric Miller for his help with the GUI Cocoa interface to the curve-fitting software, and Ruby Loch for editing and proofreading the manuscript.

7. REFERENCES

[1] T. Porkess, *The HarperCollins Dictionary of Statistics*, HarperPerenial, New York, N.Y., 1991.

[2] T. Sakurai, A.R. Newton, Delay Analysis of Series-Connected MOSFET Circuits, *IEEE Jnl. of Solid-State Circuits*, Feb. 1991, pp. 122-131.

[3] A. Chandrakasan, et. al.,Low-Power CMOS Digital Design, *IEEE Jnl. of Solid-State Circuits*, Apr. 1992, pp. 473-484.

[4] G. Gerosa, et. al., A 2.2W, 80 MHz Superscalar RISC Microprocessor, *IEEE Jnl. of Solid-State Circuits*, Dec. 1994, pp. 1440-1454.

[5] C. Nicoletta, et. al., A 450-MHz RISC Microprocessor with Enhanced Instruction Set and Copper Interconnect, *IEEE Jnl. of Solid-State Circuits*, Nov. 1999, pp. 1478-1490.

[6] N. Weste, K. Eshraghian, *Principles of VLSI Design: A Systems Perspective, 2nd Edition*, Addison-Wesley, Reading, Mass., 1993, p. 236.

[7] D. Allen, et. al., Custom Circuit Design as a Metric of Microprocessor Performance, *IBM J. Res. Develop.*, Vol. 44, No. 6, Nov. 2000, pp. 799-822.

[8] L. Glasser, D. Dobberpuhl, *The Design and Analysis of VLSI Circuits*, Addison-Wesley, Reading, Mass., 1985, p. 105.