RZ 3829 (# Z1208-001) 08/09/2012
Computer Science 5 pages

Research Report

The Influence of Transistor Properties on Performance Metrics and the Energy-Efficiency of Parallel Computations

P. Stanley-Marbell

IBM Research – Zurich
8803 Rüschlikon
Switzerland

The Influence of Transistor Properties on Performance Metrics and the Energy-Efficiency of Parallel Computations

Phillip Stanley-Marbell
IBM Research — Zürich,
Säumerstrasse 4, 8803 Rüschlikon, Switzerland

Abstract

To sustain the improvements in transistor integration density, performance, and energy per operation witnessed over the last five decades, semiconductor device engineers are investigating several device alternatives. They are however hard-pressed to relate observed transistor-level properties of potential CMOS replacements with the performance which might be achieved when these devices are used to construct complete integrated circuits such as microprocessors.

This article addresses this challenge by developing a model linking device properties to algorithm properties (parallelism and total computational work) and system operation conditions (degree of voltage and frequency scaling). This framework is then used to provide insight into which aspects of transistor operation most influence execution time, average power dissipation, and overall energy usage of parallel algorithms executing in the presence of hardware concurrency. For the often-encountered challenge of jointly optimizing execution time and energy usage, the framework enables the expression of the appropriate form of joint energy-delay metric as a function of device properties.

The presented analysis is supported by empirical characterizations of a dozen large digital circuit designs, and is further validated using performance and power measurements of a parallel algorithm executing on a state-of-the-art low-power multicore processor.

1 Introduction

In the search for device architectures to carry computing systems through deeply-scaled CMOS and beyond, device physicists are faced with a variety of choices. In addition to the many alternative tokens for representing logic state (charge, spin, and so on), devices of a given type may be tuned for different regions of operation. Fundamental to the choice among devices and architectures are their associated performance, power dissipation, potential for dense integration, and the opportunities for tradeoffs between these (e.g., through the use of parallelism at low clock speeds to maintain compute throughput).

Several candidate devices for scaling beyond the CMOS roadmap are currently being investigated, ranging from band-to-band tunneling field-effect transistors (TFETs) [14] and nanoscale-electro-mechanical-system (NEMS) relay logic [9], to devices employing various forms of electron spin [21], and graphene [13]. These varied devices often have different characteristics from traditional bulk CMOS devices. For example, NEMS proposals have limited achievable clock speeds due to mechanical inertia; they however have very low leakage, potentially permitting designs with large transistor counts, and making up for their limited clock speeds by employing architectural parallelism. The device characteristics which should be pursued by device engineers will however ultimately depend on the existence of a meaningful set of metrics which capture the constraints (e.g., power, time, or energy) under which devices will be employed.

This article derives a set of relations linking algorithm parallelism to device properties, and to measures of performance, power, energy, and tradeoffs between them. The analysis is based on properties inherent to the class of devices that represent logic values with voltages, and in which logic transfer between stages is via the charging of a capacitive load; for devices with other state tokens (e.g., electron spin), the analysis will still serve as a basis for extension, e.g., by identifying the analogs of the properties identified herein as being critical to enabling tradeoffs for voltage-based devices. The contributions of this article include:

- **The derivation of relations between algorithm parallelism and device properties**, presented in Section 3.
- **Derivation and new insight into what metrics should be used for comparing joint energy-efficiency and performance, as a function of device characteristics**, and under what conditions they are valid (Section 4).
- **Experimental measurement of power consumption and performance** of a parallel algorithm under voltage scaling on a state-of-the-art multi-core processor (Section 5).

Section 6 concludes the article with a summary of insights.

2 Related Research

Theis and Solomon [18] outline two methods for reducing power dissipation in future devices: reducing the energy lost during logic value transitions by lowering supply voltages, and the use of adiabatic logic. Given the challenges involved in designing efficient adiabatic logic circuits, the device research community has thus far focused on finding alternative logic devices that enable a significant lowering of supply voltage, without an exponential growth in leakage current.

Dynamic supply voltage and frequency scaling (DVFS) in microprocessors, as a means of reducing power dissipation, has been of interest for several decades [4, 20], due to the quadratic dependence of dynamic power dissipation on supply voltage, for a given implementation circuit. Lowering supply voltages to reduce power dissipation however often leads to a loss in performance (although the overall energy usage is still usually reduced), due to the dependence of drain current, and hence, of gate delay, on supply voltage. For long-channel devices, this dependence between clock cycle time and supply voltage, captured by the Shockley model, was linear in the region of transistor operation of interest. For short-channel devices, the improved delay model of Sakurai and Newton [12] generalized the Shockley model to account for velocity saturation effects.

Given the conflicting influences of supply voltage on performance of a fixed circuit (higher is better) and energy-efficiency (lower is better), it has been of interest to jointly consider both

energy-efficiency and delay in quantifying system efficiency. For this, the energy-delay product [8] is often used; however, as noted by Pénzes and Martin [11], the energy-delay product is dependent on supply voltage, thus conclusions reached in comparing the energy-delay for two systems at one supply voltage might change when the systems operate at different supply voltages. They thus proposed the use of energy-delay^2 ($E \cdot T^2$), which they showed, empirically for a design in 0.6 μm CMOS, to be largely independent of supply voltage. This work generalizes this idea further. The concept of *parameter-independent metrics* are introduced, with the voltage-independent metric of Pénzes and Martin being a special case, and it is demonstrated how these metrics are different functions of device technology parameters.

The joint treatment of device properties, algorithm properties, and the resulting performance and energy-efficiency, provided in this article, lends new insight into prior efforts [2, 5, 10] to investigate the energy-efficiency of the use of parallelism. In particular, the analysis provides one starting point for evaluating the role of parallelism in the quest for the "next switch" [18] to replace the CMOS transistor.

3 Energy and Parallelism Models

The power dissipation of a CMOS transistor can be decomposed into the primary components of dynamic, short-circuit, gate, and subthreshold channel leakage. The analysis that follows will focus on the dynamic and subthreshold channel leakage; gate leakage has been addressed in recent years through the use of high-κ dielectrics, while short-circuit currents are typically small when signal rise and fall times are short.

3.1 Energy model

The energy for operation of a CMOS circuit at clock frequency f and supply voltage V, with effective circuit switching capacitance C_{eff}, for an execution duration T, is given by

$$E = C_{\text{eff}} \cdot V^2 \cdot f \cdot T + I_{\text{lkg}}(V, V_T, \theta) \cdot V \cdot T, \tag{1}$$

where

$$I_{\text{lkg}}(V, V_T, \theta) = K_{\text{lkg},1} \cdot e^{\frac{K_{\text{lkg},2} \cdot q(V - V_T)}{k \cdot \theta}}.$$

V_{T}, is the threshold voltage, $K_{\text{lkg},1}$ and $K_{\text{lkg},2}$ subsume several device properties, k is Boltzmann's constant, q is the electron charge, and θ is the operating temperature in Kelvin.

Supply voltage also influences the gate drive current, which in turn determines the speed at which capacitive loads can be charged/discharged, and hence the *maximum clock frequency*, f_{max}:

$$f_{\text{max}} = \phi \frac{(V - V_{\text{min}})^{\alpha}}{V}. \tag{2}$$

The constant ϕ subsumes several device and circuit parameters, and is treated as a monolithic constant in this work. For devices that operate purely in the super-threshold region, V_{min} equals V_{T}; for devices which span the sub- and super-threshold regions, V_{min} however loses its physical interpretation. The parameter α, which must be greater than or equal to unity[1], is treated in this work as a parameter with no direct physical interpretation. Although the alpha-power-law voltage-versus-frequency dependence was originally derived by Sakurai and Newton [12] to account for short-channel effects (velocity saturation) in CMOS, it is observed to capture the behavior of a wide variety of circuits, even those with mixed super- and sub-threshold modes. Figure 1 plots published voltage versus frequency "Shmoo" characterizations for 12 large programmable digital designs, along with the resulting multi-parameter fit to α of

[1] $\alpha < 1$ would permit *decreasing* power dissipation with *increasing* performance.

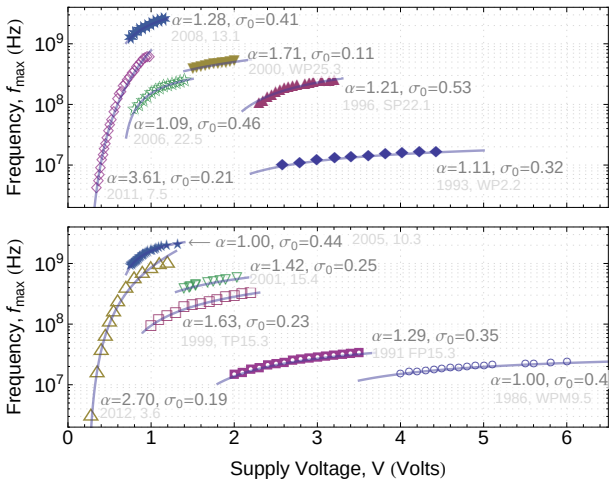

Figure 1. Empirical data from voltage-versus-frequency characterizations (points) and fits to Equation 2 (lines), for several large circuit designs published in ISSCC (1986–2012). The curves are separated into two plots for readability.

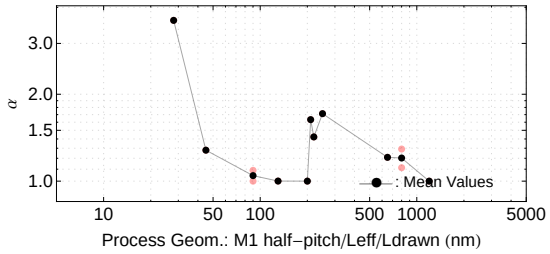

Figure 2. Fitting the V versus f_{max} characterizations of Figure 1 to Equation 2 yields no clear trend in α, although the fits in Figure 1 closely match the measured data. The parameter α may thus in principle be influenced by design choices, independent of the technology node at which a circuit is implemented.

Equation 2, and σ_0, the ratio of the fitted V_{min} to the maximum supply voltage.

The values of α vary significantly (Figure 2), with a mean across all data points of 1.59, and a standard deviation of 0.79. In what follows, α is therefore treated as a parameter that may be controlled even for a fixed technology node. One way in which such control of the overall shape of the V versus f_{max} curve (and hence of α) may be achieved, is through the facilitation of both sub- and super-threshold operation (e.g., the device "2012, 3.6" in Figure 1).

The frequency f at which a circuit operates (in Equation 1) can be chosen at will under the constraint $0 < f \leq f_{\text{max}}$; doing so while leaving supply voltage fixed however results only in a reduction in average power, but no gain in energy-efficiency. The following analyses are therefore restricted to the mode of operation where the supply voltage is always the lowest for a given target operating frequency.

V_{min} and V_{max} will be used to denote the minimum and maximum supply voltages at which a system operates (technology parameters), while σ will be used to denote the degree of voltage scaling (a system-configuration-dependent parameter). A $\sigma = 1$ indicates no voltage/frequency scaling, while $\sigma = \sigma_0$ denotes a maximally-scaled voltage, i.e.,

$$\sigma_0 < \sigma \leq 1, \tag{3}$$

and

$$\sigma_0 = \frac{V_{\min}}{V_{\max}}. \tag{4}$$

Expressed in terms of σ and σ_0, Equation 2 becomes

$$f_{\max} = \phi V_{\max}^{\alpha-1} \frac{(\sigma-\sigma_0)^{\alpha}}{\sigma}. \tag{5}$$

The supply operating point (σ) employed in a system will depend on the desired tradeoff between performance and energy-efficiency, and, importantly, on the possibility to make up for lower clock frequencies through the use of architectural parallelism.

3.2 Algorithm parallelism model

The dynamic execution of an algorithm can be represented with a data dependence graph, a directed acyclic graph (DAG) in which nodes are units of work and edges represent dependencies. These units may be instructions, basic blocks, or coarser. In the DAG model for dynamic parallelism [6], on which the following analysis is based, the units are sections of the dynamic instruction stream between points of creation or merging of parallel threads.

The number of nodes in the execution DAG constitutes the total amount of *work*, W_1, that must be completed. In a serial execution, this corresponds to the computation performed by a single processor. The length of the longest dependence chain of work units, or the *span*, is denoted by W_∞, and the average amount of parallelism, in units of work, over the course of execution, is W_1/W_∞.

In an execution employing p processors, the amount of available parallelism must be at least p in order to achieve linear speedup, i.e.,

$$1 \le p \le \frac{W_1}{W_\infty}. \tag{6}$$

The analysis that follows is restricted to computations which occur in this region, where there is sufficient algorithm parallelism for the chosen number of processors. Under these conditions, the maximum work per processor, W_p, is

$$W_p = \frac{W_1}{p}. \tag{7}$$

For the remainder of the analysis, it is assumed that communication overheads are minimal, to simplify the derivation of relations for the interaction between algorithm parallelism and device properties. For applications with significant amounts of communication, an analogous derivation of expressions for performance and power was recently developed [16]. As is demonstrated in Section 5, there are important real-world problems for which these assumptions of algorithm parallelism and communication overheads hold.

This succinct model of parallelism in the dynamic execution of algorithms can now be combined with the device-specific power and timing relations of Section 3.

3.3 Runtime, energy usage, and power dissipation of parallel algorithms

Given the definitions for clock frequency and energy in Equations 1 and 2, and per-processor parallel workload in Equation 7, it is possible to reformulate the execution time, T, for a parallel computation, as

$$T = \frac{W_1}{p \cdot f_{\max}} = \frac{W_1 \cdot V_{\max}^{1-\alpha}}{p\phi} \frac{\sigma}{(\sigma-\sigma_0)^{\alpha}}. \tag{8}$$

Substituting Equation 8 into Equation 1 yields the expression for the energy usage of the parallel algorithm execution as

$$E = \frac{W_1}{p}\sigma^2 V_{\max}^2 \left(C_{\text{eff}} + \frac{I_{\text{lkg}}(V, V_T, \theta)(\sigma V_{\max} - \sigma_0 V_{\max})^{-\alpha}}{\phi} \right). \tag{9}$$

The average power over the course of the execution, is thus also

$$P = \sigma V_{\max} \left(C_{\text{eff}} \phi (\sigma V_{\max} - \sigma_0 V_{\max})^{\alpha} + I_{\text{lkg}}(V, V_T, \theta) \right). \tag{10}$$

Equations 8, 9, and 10 encapsulate the relation between algorithm properties (W_1), hardware concurrency (p), implementation (C_{eff}), device properties ($V_{\max}$, α, ϕ, and σ_0), and system operating point (σ). Even though these relations are structured based on transistor-level equations, as will be shown by fitting data from empirical measurements to these models in Section 5, they also accurately capture the aggregate behavior of the millions of transistors making up an integrated circuit such as a microprocessor.

For devices other than CMOS, particularly those based on fundamentally different principles of operation, Equations 8 through 10 may need to be replaced as appropriate. Examples of recent work to model the energy and delay behavior of CMOS alternatives include the work of Solomon et al. [15], Behin-Aein et al. [1], and Wei et al. [19].

4 Parameter-Independent Metrics

When a single metric is of interest (e.g., only timing performance or average power dissipation), it is possible to use Equations 8 through 10 to determine which combinations of algorithms and system parameters satisfy a given time, energy, or power constraint.

In practice however, multiple metrics are often of interest. The traditional approach is to use a product of the metrics of interest, such as the energy-delay ($E \cdot T$) product proposed by Horowitz et al. [8]. Pénzes and Martin [11] previously argued that the $E \cdot T$ metric is voltage-dependent, arguing instead for $E \cdot T^2$. This concept can be generalized further to the idea of *parameter-independent metrics*, and, more importantly, using Equations 8 and 9, the appropriate form of these parameter-invariant metrics can be formulated as functions of device technology parameters.

To minimize *both* energy and delay, independent of a given parameter (e.g., supply voltage), an appropriate parameter-independent metric is of the form $E^a \cdot T^b$, picking a and b nonzero, and such that all terms of the parameter in question cancel in the product. For $\beta = \frac{b}{a}$, the preceding product can be written as $E \cdot T^{\beta}$.

Rather than simply interpreting $E \cdot T^{\beta}$ as a relative weighting of E and T, as was the focus of prior work, the analyses in the following focuses on finding values of β that make the product independent of a particular parameter. Furthermore, expressing β as a function of device properties, when possible, provides new insight into how device engineering will influence energy versus delay tradeoffs.

4.1 $V_{\max}$-independent metric

The maximum supply voltage, $V_{\max}$, at which a design operates, may be constrained, e.g., due to power supply design, supply noise, supply current, or circuit reliability concerns. It is thus of interest to be able to compare algorithms paired with hardware designs, independent of specific values of $V_{\max}$.

From Equation 9, (dynamic) energy is a function of $V_{\max}^2$, while delay (Equation 8) is a function of $V_{\max}^{1-\alpha}$. Thus, the $V_{\max}$-independent energy-delay product is achieved when

$$-2 \cdot a = (1-\alpha) \cdot b,$$

with both a and b nonzero. One valid solution is achieved with $a = 1$ and $b = \frac{2}{\alpha-1}$, i.e., $\beta = \frac{2}{\alpha-1}$. For $\alpha = 2$ (Shockley model), the $V_{\max}$-independent metric is therefore $E \cdot T^2$. As seen in Figure 1 however, devices in practice have a wide range of values for α, with no clear trend over technology generations. As α approaches unity, jointly minimizing energy and performance, independent of $V_{\max}$, requires placing more effort on minimizing delay.

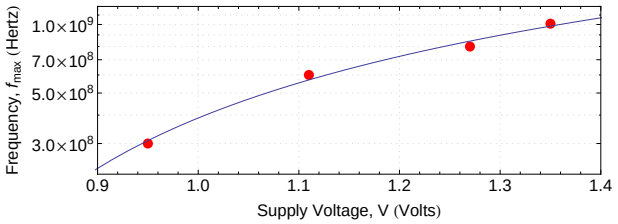

Figure 3. Voltage versus frequency dependence measured at `VDDCORE1` of the TI Omap 4430 ARM Cortex-A9 (points), fit to the model of Equation 2 (line).

4.2 W_1-independent metric

A W_1-independent metric is desired when comparing the steady-state behavior of an algorithm and platform combination, regardless of the total amount of computational work (W_1). One example of such a scenario is when studying the steady-state behavior of a streaming application.

From Equations 8 and 9, both energy and delay are functions of W_1 raised to the same exponent (unity). The W_1-independent energy-delay product is therefore achieved when $\beta = -1$. This is intuitively pleasing, as it corresponds to the average power dissipation, which is indeed independent of W_1.

4.3 Nonexistent independences

The (dynamic) energy for a parallel computation is a function of σ^2, while the delay is a function of $\sigma/(\sigma - \sigma_0)$. For $\sigma \gg \sigma_0$ (i.e., when far above the minimum supply setting in a highly voltage-scalable system), delay becomes independent of the degree of supply voltage scaling, σ. This makes a truly σ-independent $E \cdot T$ metric unattainable: i.e., jointly minimizing energy and delay cannot be made independent of the degree of voltage scaling, σ.

A metric that is jointly independent of both $V_{\max}$ and W_1 has values of the exponents a and b of E and T respectively that satisfy the system of simultaneous equations

$$-a = b,$$
$$-2 \cdot a = (1 - \alpha)b,$$

which however has no valid solutions given the constraint that $\alpha \geq 1$. Thus, one cannot jointly minimize energy and delay independent of both the total amount of computational work W_1 (algorithm dependent) and the maximum supply voltage $V_{\max}$ (technology dependent).

4.4 Implications of device properties

As a function of a semiconductor process configuration's voltage versus frequency characteristics (and hence α), Sections 4.1 and 4.2 provided formulations of the joint energy-delay metrics to be used under two different system usage models. That study of parameter-independent metrics yields three main insights.

First, the metric of interest (e.g., $E \cdot T^{\frac{2}{\alpha-1}}$ or $E \cdot T^{-1}$) depends on the system's evaluation and usage criteria. Second the $V_{\max}$-independent metric is a function of device properties, and is influenced by α. The W_1-independent $E \cdot T^{-1}$ is however independent of device properties. Finally, jointly minimizing dynamic energy[2] and delay cannot be made independent of the degree of voltage scaling.

5 Empirical Measurements

The preceding sections outlined a model for capturing the interaction between device properties (α, σ_0, ϕ, $V_{\min}$, and $V_{\max}$), algorithm properties (W_1), implementation/architecture (C_{eff}), and the system operating point (σ). Although empirical values of device-level parameters were provided to support the argument, one question remains: Do the performance, energy, and power models of Equations 8, 9, and 10 truly reflect the behavior of complete integrated circuits executing parallel algorithms? To address this question, performance and power measurements of a parallel algorithm executing on a multi-core platform were carried out.

For the evaluation, a cache-oblivious [7] parallel matrix-matrix multiplication (MMM) application was employed. Parallel matrix-matrix multiplication was chosen as a benchmark as it forms a crucial subroutine in many compute-intensive scientific and commercial data analytics workloads. The application, which was written in the Cilk dynamic multithreading language [3], was run over the Cilk 5.4.6 runtime, which was ported to the ARM architecture to facilitate the experiments. For input data, 4 M-entry product matrices were employed, populated with uniformly distributed random data in the range of 0.0 to 1.0, to maximize switching activity in the processor datapath.

5.1 Measuring α, ϕ, σ_0, $V_{\min}$, and $V_{\max}$

For empirical measurements, an OMAP4430 dual-core ARM Cortex-A9 processor [17] from Texas Instruments was employed. The hardware used in the measurements was modified to enable isolating the power consumption of the processor cores from that of other on-chip and board-level peripherals. The processor, implemented in 45 nm CMOS, supports execution at clock frequencies of 300 MHz, 600 MHz, 800 MHz, and 1008 MHz. The processor contains a hardware subsystem ("SmartReflex") which cooperates with an external voltage regulator to set supply voltages, based on the requested clock frequency. On the test hardware platform, core supply voltages at these aforementioned frequencies, of 0.95 V, 1.11 V, 1.27 V, and 1.35 V were measured. The leakage current, $I_{\text{lkg}}(V, V_T, \theta)$, which was estimated from measurements, is treated as a constant, $\overline{I_{\text{lkg}}}$, and encapsulates all sources of idle power dissipation.

Figure 3 plots the measured supply voltage at the processor core (`VDDCORE1` on the OMAP4430) across operating frequencies. Fitting the measurements to Equation 2 yields values of the device technology parameters $\phi = 2.6 \times 10^9$, $\alpha = 1.69$, $V_{\min} = 0.67$, $V_{\max} = 1.35$, and $\sigma_0 = 0.49$. Fitting a set of active and idle power measurements at different operating frequencies to Equation 10 yields $C_{\text{eff}} = 298.84\,\text{pF}$ and $\overline{I_{\text{lkg}}} = 359.74\,\text{mA}$.

5.2 Model, Measurements, and Observations

The models of Equations 9 and 10 for the energy, E, and power, P, were derived based on transistor-level relations. These relations however also hold for entire integrated circuits, with the aggregate behavior of the millions of transistors making up a die influencing the size of the multiplicative constants in the expressions.

Figures 4 and 5 show a series of measurements of total energy and average power, for single- and dual-core configurations (i.e., $p = 1$ and $p = 2$). In both figures, the points represent measurements, and the dashed lines are the trends predicted by Equations 9 and 10 for the model constants estimated in Section 5.1.

The models of Equations 9 and 10 enable a number of interesting insights. For example, when considering the isolated metric of energy usage, for the parameter values extracted in Section 5.1, the model of Equation 9 predicts lower average power and lower total energy usage across all degrees of voltage scaling σ, if $p = 2$ (as opposed to $p = 1$); this is corroborated by the measurements in Figure 4. Similarly, due to leakage, the model predicts a minimum in energy for both the single-core and dual-core cases, at or below 600 MHz, which is again validated by the measurements.

6 Summary and Insights

The search for new devices to replace CMOS poses many challenges to device engineers. Some candidate devices, such as field-effect transistors based on band-to-band tunneling (TFETs) and nanoscale-electro-mechanical-systems (NEMS) may potentially be

[2] For TFETs and NEMS, leakage is small compared to CMOS.

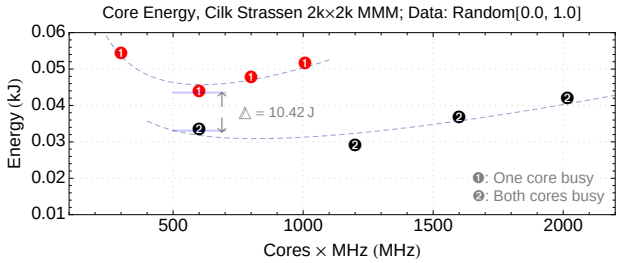

Figure 4. Measured core-only energy (points) and fit to the model of Equation 9 (dashed lines). The compared configurations have approximately equal runtimes.

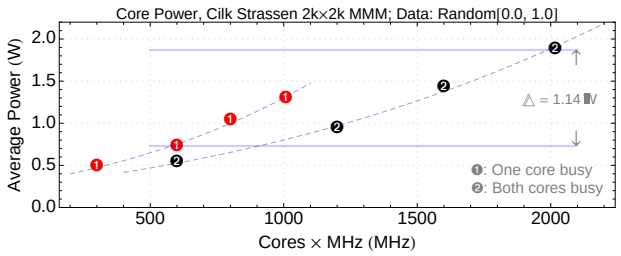

Figure 5. Measured core-only power (points) and fit to the model of Equation 10 (dashed lines). The compared points are approximately iso-energy in Figure 4.

limited in maximum clock frequencies. Due to their low leakage however, they might be integrated on-die in large numbers, making up for the limited clock frequencies with architectural parallelism, which must be exploited by algorithms. This article derived a set of relations between the properties of parallel algorithms, properties of the device technologies of the architectures on which they execute, and the resultant performance, power, and energy-efficiency. Using performance and power measurements on a dual-core ARM Cortex-A9, it was demonstrated that the derived relations accurately capture the behavior of real systems in today's semiconductor technologies.

When energy and delay are required to be jointly optimized, the *parameter-invariant energy-delay metrics* introduced in Section 4 specify the precise form of the appropriate joint energy-delay metrics, as a function of device properties. The relations presented linking algorithm and device properties, together with the parameter-invariant energy-delay metrics, provide an analytic basis for understanding the role of algorithm parallelism in the search for an energy-efficient CMOS successor.

7 Acknowledgements

The article was influenced by discussions with Volker Strumpen, who provided the idea of the formulation of the interrelation of V and $f_{\max}$ in terms of σ and σ_0, and proposed the expression of work and span in terms of instructions (W_1, W_∞) as opposed to time (T_1, T_∞). Christoph Hagleitner and various anonymous reviewers provided valuable suggestions on improving the exposition.

8 References

[1] B. Behin-Aein, A. Sarkar, S. Srinivasan, and S. Datta. Switching energy-delay of all spin logic devices. *Applied Physics Letters*, 98:123510, 2011.

[2] B. D. Bingham and M. R. Greenstreet. Modeling energy-time tradeoffs in vlsi computation. *IEEE Transactions on Computers*, 61:530–547, 2012.

[3] R. D. Blumofe, C. F. Joerg, B. C. Kuszmaul, C. E. Leiserson, K. H. Randall, and Y. Zhou. Cilk: an efficient multithreaded runtime system. In *Proceedings of the fifth ACM SIGPLAN symposium on Principles and practice of parallel programming*, PPOPP '95, pages 207–216, New York, NY, USA, 1995. ACM.

[4] T. D. Burd and R. W. Brodersen. Design issues for dynamic voltage scaling. In *Proceedings of the 2000 international symposium on Low power electronics and design*, ISLPED '00, pages 9–14, New York, NY, USA, 2000. ACM.

[5] A. Chandrakasan, S. Sheng, and R. Brodersen. Low-power CMOS digital design. *IEEE Journal of Solid-State Circuits*, 27(4):473–484, 1992.

[6] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, Third Edition*. The MIT Press, 2009.

[7] M. Frigo, C. E. Leiserson, H. Prokop, and S. Ramachandran. Cache-oblivious algorithms. In *Proceedings of the 40th Annual Symposium on Foundations of Computer Science*, FOCS '99, pages 285–, Washington, DC, USA, 1999. IEEE Computer Society.

[8] M. Horowitz, T. Indermaur, and R. Gonzalez. Low-power digital design. In *IEEE Symposium on Low Power Electronics*, pages 8–11, Oct 1994.

[9] J. Jeon, V. Pott, H. Kam, R. Nathanael, E. Alon, and T. Liu. Perfectly complementary relay design for digital logic applications. *Electron Device Letters, IEEE*, 31(4):371–373, 2010.

[10] V. A. Korthikanti and G. Agha. Analysis of parallel algorithms for energy conservation in scalable multicore architectures. In *Proceedings of the 2009 International Conference on Parallel Processing*, ICPP '09, pages 212–219, Washington, DC, USA, 2009. IEEE Computer Society.

[11] P. I. Pénzes and A. J. Martin. Energy-delay efficiency of VLSI computations. In *Proceedings of the 12th ACM Great Lakes Symposium on VLSI (GLSVLSI-02)*, pages 104–111, New York, Apr. 18–20 2002. ACM Press.

[12] T. Sakurai and A. Newton. Alpha-power law MOSFET model and its applications to CMOS inverter delay and other formulas. *IEEE Journal of Solid-State Circuits*, 25(2):584–594, 1990.

[13] F. Schwierz. Graphene transistors. *Nature nanotechnology*, 5(7):487–496, 2010.

[14] A. Seabaugh and Q. Zhang. Low-voltage tunnel transistors for beyond cmos logic. *Proceedings of the IEEE*, 98(12):2095–2110, 2010.

[15] P. Solomon, D. Frank, and S. Koswatta. Compact model and performance estimation for tunneling nanowire fet. In *Device Research Conference (DRC), 2011 69th Annual*, pages 197–198. IEEE, 2011.

[16] P. Stanley-Marbell. Parallelism, performance, and energy-efficiency tradeoffs for in situ sensor data processing. *IEEE Embedded Systems Letters Journal*, 3(1):16–19, 2011.

[17] Texas Instruments Incorporated. OMAP4430 Technical Reference Manual (Literature Number: SWPU231J), July 2010.

[18] T. Theis and P. Solomon. In quest of the "next switch": Prospects for greatly reduced power dissipation in a successor to the silicon field-effect transistor. *Proceedings of the IEEE*, 98(12):2005 –2014, dec. 2010.

[19] L. Wei, S. Oh, and H. Wong. Performance benchmarks for Si, III–V, TFET, and carbon nanotube FET — re-thinking the technology assessment methodology for complementary logic applications. In *Electron Devices Meeting (IEDM), 2010 IEEE International*, pages 16–2. IEEE, 2010.

[20] M. Weiser, B. Welch, A. Demers, and S. Shenker. Scheduling for reduced cpu energy. In *Proceedings of the 1st USENIX conference on Operating Systems Design and Implementation*, OSDI '94, Berkeley, CA, USA, 1994. USENIX Association.

[21] S. Wolf, D. Awschalom, R. Buhrman, J. Daughton, S. Von Molnar, M. Roukes, A. Chtchelkanova, and D. Treger. Spintronics: A spin-based electronics vision for the future. *Science*, 294(5546):1488, 2001.