Power Signal Processing: A New Perspective for Power Analysis and Optimization

Quming Zhou, Lin Zhong and Kartik Mohanram
Department of Electrical and Computer Engineering
Rice University, Houston, TX 77005
{quming, lzhong, kmram}@rice.edu

Abstract

To address the productivity bottlenecks in power analysis and optimization of modern systems, we propose to treat power as a signal and leverage the rich set of signal processing techniques. We first investigate the power signal properties of digital systems and analyze their limitations. We then study signal processing techniques for detecting temporal and structural correlations of power signals. Finally, we employ these techniques to accelerate the simulation of an architecture-level power simulator. Our experiments with SPEC2000 show that we can speed up the simulation by 100X without introducing significant errors at various resolution levels.

Categories and Subject Descriptors: J.6 [Computer- Aided Engineering]: Computer-aided design (CAD)
General Terms: Algorithms, Design
Keywords: Power, Trace, Signal Processing, Power Simulation

1. Introduction

We have seen two designer productivity challenges to power optimization of a large electronic system, being it a system-on-a-chip (SoC), a system-in-a-package (SiP), or a complete computer system. First, average power estimation is not enough. Instead, a detailed power trace is often required to identify and subsequently minimize system behavior that consumes high power. Moreover, a dynamic power trace covering a relatively long runtime is important to validate a system for performance and thermal management. For example, since performance-curbing techniques, such as clock throttling and voltage scaling, are often used to meet the thermal challenge, power behavior will have a significant impact on system performance. Unfortunately, cycle-accurate power simulation of a large system for millions of cycles is notoriously slow [1]. For example, it takes about one hour to simulate only 4000 cycles for the SPE unit on the IBM CELL processor [2]. On the other hand, techniques aiming at speed improvement often reduce to average power estimation.

Second, power simulation or measurement of large electronic systems can produce a massive amount of data. Such data contain important information for design optimization and validation. Unfortunately, it is extremely hard and counter-productive for a designer to manually examine them. Moreover, visual presentation and interactive manipulation of such massive data are also challenging. There is a great need for tools to identify suspicious power behavior from massive power data and, ideally, suggest ways to improve it.

We address these two challenges with a signal processing approach. We treat power consumption of an electronic system as a digital signal and treat that of its components as a multi-dimension signal or distributed signals. A component can be a gate, ALU, processor core, or even an entire chip on a printed-circuit board. Then, we explore advanced signal processing and pattern analysis techniques to study the power signal. We call this *power signal processing*. While signal processing techniques, such as Fourier and Wavelet analysis, have been used for micro-architecture performance [3] and supply voltage analysis [4], they have not yet been applied to power behavior, as to the best of our knowledge.

In this work, we make the following three contributions.

- Studied the properties of power signals.
- Proposed effective and efficient algorithms to detect *temporal* and *structural* correlations in power signals
- Investigated the application of power signal processing to accelerating power simulation.

We believe that power signal processing introduces a new perspective into power analysis and optimization. Our experiments with SPEC2000 show that we can speed up the simulation by 100X without introducing significant errors at various resolution levels. Our work is an initial step toward utilizing the extremely rich collection of tools from the signal processing and pattern analysis research community.

The paper is organized as follows. In Section 2, we introduce power consumption as a signal and discuss its properties. In Section 3, we introduce signal processing techniques that are relevant to power analysis and optimization. We also present techniques that make new discoveries regarding power behavior. In Section 4, we focus on power simulation acceleration. Finally, we present our experimental results in Section 5 and conclude in Section 6.

2. Power as Signal

We first provide necessary background and motivations for power signal processing as well as address the unique properties of power signals.

2.1 Signal Sources: Estimation and Measurement

Dynamic power traces can be obtained through either cycle-accurate power estimation or direct power measurement. Cycle-accurate

ISLPED'07, August 27–29, 2007, Portland, Oregon, USA.

power estimation at various levels of abstraction have been widely used in industry [2]. The lower level, the more accurate but the slower is the estimation. Therefore, cycle-accurate power estimation is always concerned with tradeoffs between speed and accuracy. The most accurate estimation is running a SPICE-like simulator on a transistor-level netlist, which is too slow to be practical for large circuits. Register-transfer level power estimation can produce relative accurate traces but still suffer from its slow speed [1]. Many techniques to accelerate cycle-accurate power estimation have been studied [1, 5, 6]. Many architectural level power simulators for microprocessors have been presented in literature [7–9]. Being very fast, they are short in accuracy and are unable to guide clock gating at the RTL level [2]. For a high speed yet accurate estimation, our proposed power signal processing approach seeks to achieve multi-resolution power estimation, i.e., to run architecture-level estimation while selectively applying gate-level estimation only in interesting cycles.

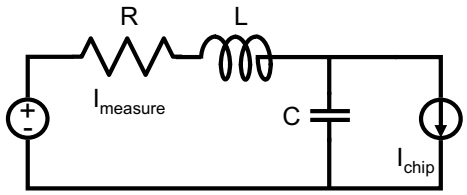


Figure 1: Second-order RLC model for the power-supply network.

Cycle-accurate power estimation is, however, limited in the accuracy to reflect the power dynamics of the real system. Most estimation technologies and simulators are memoryless, meaning that power consumption in each cycle and in each component is calculated independently. In a real system, decoupling capacitors, parasitic capacitance, and even by-pass capacitors make this untrue. Their net effect on the system power behavior is similar to a low-pass filter.

The other way to obtain dynamic power traces is direct power measurement. While direct power measurement offers absolute accuracy, it is limited in both temporal and structural resolutions. As mentioned above, due to the existence of decoupling capacitors, parasitic capacitance, and by-pass capacitance, the power consumption of a cycle or component is affected by its temporal or spatial neighbors. The power trace obtained through measurement, though accurate, is unable to offer the highest, i.e., cycle-by-cycle or component-by-component, resolutions.

2.1.1 *Inherent uncertainty in power signals*

To further examine the inherent uncertainty in power signals introduced by decoupling capacitance and by-pass capacitors, we model the power-supply network of an electronic system with a second-order resistive, inductive, and capacitive (RLC) circuit, shown in Fig. 1. In the model, the resistor represents the resistance of the power-supply network; the inductor represents parasitic inductance, e.g., that introduced by chip-die connectors [10]; the capacitor represents parasitic capacitance and on-die decoupling capacitance to curb abnormality in the power-supply network. The current draw by the system can be represented by a current source, I_{chip}. Since I_{chip} is not directly observable to power measurement, power measurement documents $I_{measure}$ instead. Unfortunately, the power-supply network will suppress much temporal dynamics in I_{chip} so that $I_{measure}$ will be at most the low-pass filtered I_{chip}. When I_{chip} is spectrally steady, the RLC circuit is low-pass filter. For an example, we use parameters from [10] for a high-performance processor with a 1GHz clock: $R = 500u\Omega$, chip-die connector inductance $L = 0.005nH$, and on-die decoupling capacitor $C = 500nF$. The circuit model for the power-supply network has a resonant frequency of 100MHz as defined by $f = \frac{1}{2\pi\sqrt{LC}}$ [11].

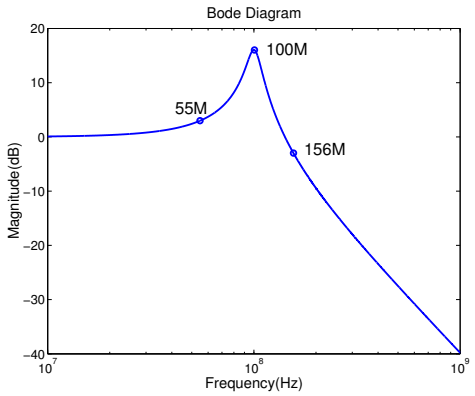


Figure 2: Bode diagram of the power-supply network

The frequency response is shown in Fig. 2. The -3dB cutoff frequency is 156MHz, 56% higher than the resonance frequency. Any harmonic frequency of I_{chip} greater than 156MHz will be attenuated. As shown in Fig. 2, the magnitude of 1GHz frequency will be reduced to 1% of the original value.

When I_{chip} is not spectrally steady, the power-supply network will further impact the accuracy of $I_{measure}$ when the RLC circuit takes time to enter a new steady state. Therefore, the power-supply network will attenuate the frequency components in I_{chip} that are higher than the resonant frequency, and more attenuation at higher frequencies. Hence, the frequency components higher than the resonant frequency in $I_{measure}$ will not accurately reflect those in I_{chip}. In another word, a sampling rate much higher than the resonant frequency will not produce a power signal with more reliable temporal dynamics. We then employ SPICE to simulate the circuit in Fig. 1 with I_{chip} running at 1GHz with a triangle shape [12], which is higher than the resonance frequency of 100MHz. Fig. 3 presents the plots for both I_{chip} and $I_{measure}$. The current $I_{measure}$ is heavily modulated by the power supply circuit as shown by its fluctuating waveform. An error will occur if directly measuring the current to estimate the cycle-accurate power. The waveform is stabilized after 70 cycles in the figure, which implies the measurable current is an average value for at least 70 cycles.

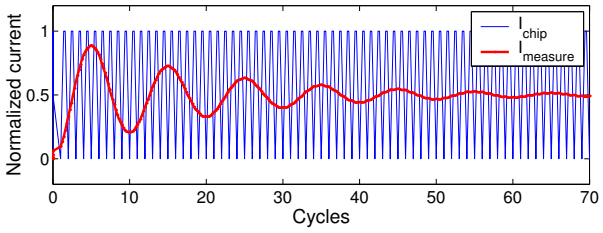


Figure 3: Cycle-accurate current (power) at 1GHz: the ringing of the measured current $I_{measure}$ disallows a cycle-accurate measurement.

In summary, the power-supply network significantly limits the temporal dynamics that power measurement can capture. As a side effect, it also suppresses security attacks based on power analysis. As long as a security-sensitive behavior happens at a higher frequency than the -3dB cutoff frequency or the resonant frequency, direct power measurement will be unlikely to uncover it.

2.2 Power signal properties

The rationale behind our proposed approach is that power traces obtained through simulation and measurement can be naturally treated as time-discrete signals, or power signals. Moreover, power signals exhibit many properties that are amenable to digital signal processing.

To illustrate the properties of a power signal, we use a cycle-accurate power trace generated by an industry RTL power simulation for an HDTV ASIC module as an example. Part of the trace is shown in Fig. 4. The figure also shows power contributed by three different types of data path units, functional units, multiplexers, and registers. Power traces typically have rich periodicity, as is apparent from Fig. 4. Knowing the periodicity of a power trace, we can recover or synthesize a power trace that approximates the original one, and potentially accelerate power simulation significantly. Fig. 4 also shows that power consumption by multiplexers and functional units are highly related. Knowing such structural relations among components, we can significantly speed up power simulation by skipping the simulation for either multiplexers or functional units.

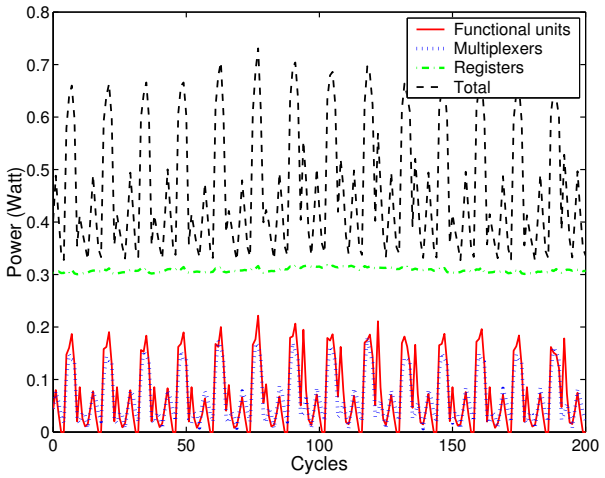


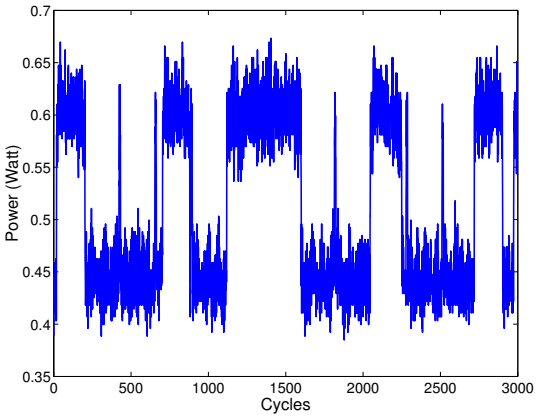
Figure 4: Cycle-accurate power traces: power traces generated from RTL-level simulation have periodicity and correlations.

Fig. 5(a) is a power trace of a Smartphone measured at 10K samples/sec, when the Smartphone is playing a video clip using Windows Media Player Mobile. The power trace has an apparent pattern that the trace repeats around every 670 cycles. It corresponds to a frequency of 15Hz (10K/670=15), the number of video frames per second. The frame rate can also be visualized in the frequency domain. Fig. 5(b) gives the time-frequency characteristics of the power trace, which reveals a strong frequency component at 15Hz. Additional, the observation that the dominant frequency at 15Hz is quite stable across the whole trace supports the periodicity of 670 cycles in the trace.

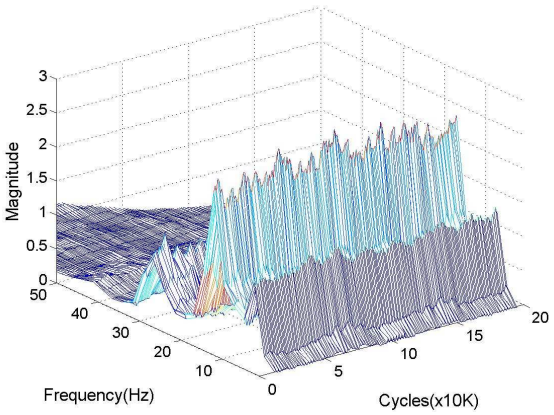
The highly predicable power trace is essentially correlated with the executed program. For example, loops in the algorithmic specification of a system create frequency components in the power trace. Nested loops create co-existing frequency components.

Moreover, finer power behavior revealed under high temporal resolution is usually introduced by lower level design features. Through power signal analysis and processing, we can relate power behavior with design features, and identify sources that introduce undesirable power behavior. Undesirable power behavior can include

- extremely high peak power,



(a) Power signal: the periodicity is 670 cycles



(b) The time-spectrum of the power signal: prominent energy at 15Hz

Figure 5: Power signal of a Smartphone playing a video at 15 frames/sec and its spectrum: the sampling rate is 10K per sec.

- long-lasting high power period,
- repeated high-power patterns, and
- power behavior that reveals implementation information.

While 1-3) are quite obvious for power and thermal management reasons, 4) is related to system security. Differential power analysis [13] has been used to attack a system by comparing power traces generated by different inputs.

2.3 Resolution of Power Signals

We use "resolution" to refer to how detailed temporal dynamics is in a power signal. If a power signal can provide the average power for any m consecutive cycles, we say that it is with a resolution level of m, the level m. Average power estimation for a whole simulation can be viewed of the level ∞; cycle-accurate power traces are of the level 1, which is the highest level. The accuracy of a power trace can be measured at different resolution levels too. In this work, we employ the following error definition for the level m:

Definition: Error at the level m: Given a power trace sequence $S = [W_1, W_2 \cdots, W_n]$, W_i being a sample window with m cycles, we have measurement (or estimation) M_i for each window W_i. The error at the level m is defined as

$$Error = \frac{1}{n}\sum_{i=0}^{n}\left|\frac{mean(M_i) - mean(W_i)}{mean(W_i)}\right|, \qquad (1)$$

where the absolution error is used to prevent the positive and negative errors from canceling each other out. The measurement M_i could be measured samples inside window W_i, or predicated values from adjacent windows if no simulation is carried out in window W_i.

By introducing the concept of error at a resolution, we are able to justify a power simulator or measurement. The error of measured current (power) consumption in Fig. 3 is 79.2% at the level 10, and reduces to 2.7% at the level 70.

3. Correlation analysis

In this section, we discuss two types of correlations in power signals, temporal correlation and structural correlation. A trace signal x is temporal correlated with a time lag t_0 if $x(t) = x(t - t_0)$. Due to the noise of the trace, the equation may not be exactly valid. We consider an local periodicity of a trace. The periodicity may vary in a long term. Similarly, the structural correlation between two trace signals is also time-dependent.

3.1 Temporal correlation

Temporal correlation is the relation of a group of cycles with another group in the power signal. The most apparent temporal correlation is the periodicity. The periodicity of a trace will be revealed as peaks in the power signal spectrum. The spectrum gives the average energy of a signal at each frequency. A peak at frequency f_i is significant if

$$Magnitude(f_i) > u_p + k\sigma_p, \tag{2}$$

where u_p is the average magnitude over all frequencies, k a threshold value (typically 3), and σ_p the standard deviation in the magnitude over all frequencies. For an N-cycle power trace, we use the average power spectrum of L-point windows. A moving window of L-points with 50% overlap is applied to the N-cycle trace to from $2N/L - 1$ sections of length L. Then the spectrums of these sections are averaged. We use the largest significant frequency as the periodicity of the trace (p).

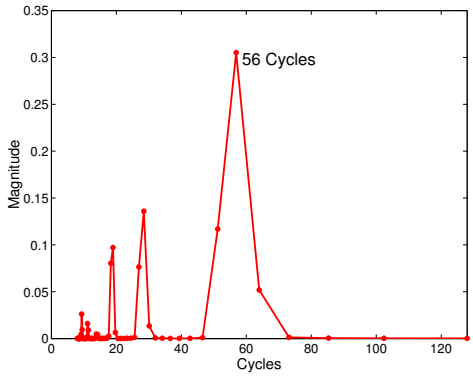


Figure 6: Power spectrum of HDTV in Fig. 4: a significant magnitude peak is detected at 56 cycles, indicating a periodicity of 56 cycles.

The spectrum of the HDTV trace is shown in Fig. 6. The significant periodicity is 56 cycles as denoted by the peak. It means that the trace repeats every 56 cycles.

3.2 Structural correlation

Structural correlation is the cross correlation between different components in a system. Fig. 4 provides an example for the correlation between the power consumption by different system components. Cross correlation is a standard method of estimating the degree to which two series are correlated. We use cross correlation analysis to explore the associations of different power components. Cross correlation can not tell the casual relationship between two components, i.e., one components determines the other. Hence, we choose one with larger power consumption as the dominant component between two correlated components.

Consider two power signals $x(i)$ and $y(i)$, where $i = 0, 1, 2...N - 1$. The cross correlation r at delay d is defined as

$$r(d) = \frac{\sum_{i=0}^{N-1} [(x(i) - u_x)(y(i-d) - u_y)]}{\sqrt{\sum_{i=0}^{N-1} (x(i) - u_x)^2} \sqrt{\sum_{i=0}^{N-1} (y(i-d) - u_y)^2}}, \tag{3}$$

where u_x and u_y are the means of corresponding series. When the index of the series is out of the range $[0, N-1]$, we use zero as the values. The denominator in the expression above serves to normalize the correlation coefficients such that $r(d) \in [-1, 1]$, the bounds indicating maximum correlation and 0 indicating no correlation. A high negative correlation indicates a high correlation but of the inverse of one of the series. The range of delay d is chosen between $[-p/2, p/2]$, where p is the detected periodicity. We use the maximum $|r(d)|$ among $d \in [-p/2, p/2]$ as the cross correlation of two series.

We employ t-test [14] to test the statistical significance of r. T-test evaluates the means of two groups are statistically different from each other. The hypotheses for the test are $H_0 : r = 0$ and $H_a : r \neq 0$. A low p-value for the test (less than 0.05 for example) indicates that there is evidence to reject the null hypothesis H_0 in favor of the alternative hypothesis H_a, or that there is a statistically significant relationship between the two series.

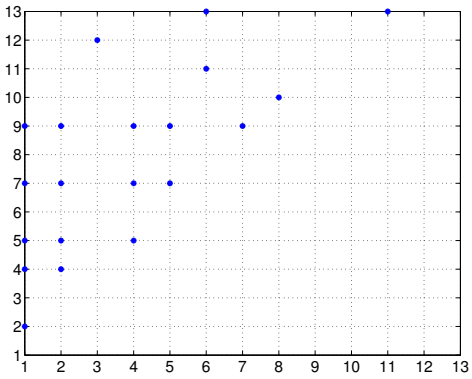


Figure 7: Power signal correlation matrix of components: components 1, 3, 6, and 8 are chosen as the major components in power simulation.

Fig. 7 gives correlations of 13 components in an architectural power simulator, Sim-Panalyzer [8] from University of Michigan. If a significant correlation with p-value= 0.01 exists between two components i and j, we mark a star at position [i, j]. Since the correlation matrix is symmetric, only the upper portion is given. Four components 1, 3, 6 and 8, being highly correlated with all other components, are chosen as the major components in power simulation. By tracking the powers of those major components instead of all components, we will speed up power simulation.

4. Adaptive acceleration of power simulation

To illustrate the applications of power signal processing, we next demonstrate how it can be applied to accelerating power simula-

tion. We show that power traces can be obtained by selectively running the power simulator without sacrificing the accuracy much.

In Section 3, we showed that loops in system behavior introduced power signal with significant harmonic frequencies. This inspired us to employ the temporal relations for selective simulation. Similarly, the inspiration for structural selection comes from the high correlations among the individual components in large systems. A power simulator usually breaks the whole architecture into many smaller functional components, each having its own power model. Depending on the program execution, the total power is the sum of involved components. In Section 3, our structural correlation analysis shows that a small number of dominating components are enough for the total power estimation. As a result, the power simulation can be faster if only simulating major components.

Based on the temporal and structural correlation detection, we devise an adaptive power simulation process, as described in Algorithm 1. In the process, we start with extracting an vector $\vec{T}^*$ for each simulated N-cycle trace, and compare it with vector $\vec{T}$. If vectors are matching, we double the skipped cycles and run another N-cycle simulation; otherwise, we simulate the successive N cycles. In step 2, a frequency of zero is used in case no significant frequency is detected as Eqn. 2. We use thresholding to determine the vector matching in step 7. Two vectors are matching if differences of all corresponding terms are less than the thresholds.

Algorithm 1 Adaptive Sampling Power Simulation

1: Run a power trace Tr_0 with N cycles
2: Calculate mean (u), variance (σ), and periodicity (p)
3: Initialize a vector $\vec{T} = [u, \sigma, p]$
4: Let the index number $ind = 1$
5: Skip $(Ind - 1) * p$ cycles and simulate N-cycle power trace Tr
6: Build another vector $\vec{T}^* = [u_r, \sigma_r, p_r]$
7: **if** $\vec{T} \approx \vec{T}^*$ **then**
8: $\quad Ind = 2 * Ind$
9: **else**
10: $\quad Ind = 0$
11: **end if**
12: Let $\vec{T} = \vec{T}^*$, and goto step 5

The power simulation employed in step 5 can employ a full power simulator including all components, or use a partial simulation based on the correlation analysis of different components. The partial simulation reduces the simulation time and data by cutting down the involved components. We describe our partial simulation version for generating an N-cycle power trace in step 5 in Algorithm 2.

Algorithm 2 Partial Simulation

1: Run L-cycle simulation fully
2: Analyze structural correlation
3: Determine major and non-major power components
4: Simulate major components for $N - L$ cycles
5: Add average power of non-major components from previous L cycles

The structural correlation analysis is used to identify which component is highly associated with another. For two highly correlated components, if one is much less than the other in the average power, the power of the small one can be simplified into a constant value without utilizing its detailed and time-consuming power model. This was addressed in Section 4.

5. Adaptive sampling results

We employ SPEC2000 [15] as our benchmarks to evaluate the effectiveness of the adaptive acceleration based on power signal processing. We run Sim-Panalyzer [8] on SPEC2K applications with the default inputs. Sim-Panalyzer models an ARM processor architecture and performs cycle-accurate power simulation. Although the accuracy of most architectural power simulation is often disputable, we view Sim-Panalyzer as a system itself, instead of the ARM processor it attempts to model. We collect power traces of all 13 components for five million cycles and use them as the baseline to apply our adaptive sampling and partial simulation techniques presented in Section 4. Table 1 summarizes the accelerated results and their errors.

In Table 1, the second column denotes the accelerated ratio. It is the ratio of the total cycles to the simulated cycles based on the adaptive sampling and all power components. The third column *Num* denotes the average number of major power components. The fourth column denotes the acceleration using the partial simulation to estimate the total power. We use the partial simulation for maximal acceleration and compare the results with the baseline at three different resolution levels: ∞ (average power over the whole trace), 100, and 1000.

To validate the efficient of our power simulation based on the adaptive sampling, we compare its results with other two sampling methods, periodic [16] and random. In both cases, the whole trace is still divided into windows with m-cycle each. The periodic sampling chooses the first cycle from every window; the random sampling uniformly chooses a random cycle from every window. When m= 100, 100X speedup over the cycle-accurate simulation can be achieved. The error at the level 100 for both periodic and random samplings are reported.

The table clearly demonstrates that the adaptive sampling is able to accelerate the simulation up to 96.7X with negligible errors. The performances of the periodic sampling and the random sampling are comparable and both highly depend on the benchmark. The standard deviation of approximation errors across the eighteen benchmarks are 1.7% for the adaptive sampling, much smaller than 9.9% of the periodic or random sampling. It clearly shows that the adaptive sampling achieves a much lower estimation error over all cases, making it more suitable for simulation acceleration.

6. Conclusions

In this paper, we first investigated the power signal properties of digital systems and analyzed the limitations power signal sources: cycle-accurate simulation and direct measurement. We then investigated signal processing techniques for discovering temporal and structural relationships of power signals. To demonstrate the applications of power signal processing, we applied these techniques to accelerating an architecture-level processor power simulator. Our experiments with SPEC2000 showed that power signal processing can improve power simulation speed by 100X with a negligible impact on power signal properties.

Our study shows that cycle-accurate at a system level is not necessary for many design tasks, such as power management and simulation. First, a well designed power supply network with decoupling capacitance will suppress cycle-accurate current so that it can not be detected accurately. Second, simulation-based power traces are highly predictable. Our accelerating 100X in SPEC2K benchmarks motivates a power simulator being able to support various tradeoffs between resolutions and speeds is more desirable. Power signal processing readily supplies basic techniques for such a simulator.

Beyond accelerating power simulation, future applications of power

Table 1: Simulation acceleration speed-up (X) and errors at different resolution levels (%)

Bench	Adaptive full simulation	Adaptive partial simulation					Traditional sampling	
	Speed-up	Num	Speed-up	Error (%)			Error at level 100(%)	
				level ∞	level 100	level 1000	Periodic	Random
ammp	57.9	2.0	227.1	0.3	5.7	3.3	43.3	43.2
applu	22.9	2.0	102.7	0.5	6.4	2.9	19.7	19.7
apsi	88.2	2.0	324.9	0.1	4.0	4.2	2.2	2.2
art	42.8	2.8	113.8	0.3	4.7	0.9	6.6	6.7
bzip	39.4	3.0	110.0	0.1	5.4	4.1	5.5	5.6
craf	26.0	3.9	54.1	2.4	2.7	0.6	14.2	14.1
equa	31.2	3.9	62.7	1.8	5.0	2.6	13.3	13.3
gal	9.6	3.0	26.2	2.6	3.9	2.3	29.1	29.2
gap	24.3	3.0	59.8	0.2	1.1	1.9	3.0	3.1
gcc	9.9	3.4	24.7	0.4	6.1	3.1	12.9	12.9
gzip	17.3	3.2	42.8	1.2	4.2	2.5	13.3	13.2
luca	40.8	2.1	173.4	0.1	4.7	1.0	8.9	9.0
mcf	44.3	2.1	159.5	0.9	1.8	3.1	5.3	5.3
mesa	19.5	3.3	43.2	0.6	1.2	1.0	18.5	18.5
mgrid	21.4	4.3	49.9	1.2	2.2	1.3	13.4	13.6
swim	18.9	3.0	53.8	0.5	4.8	3.2	15.0	14.9
twolf	26.4	3.3	69.1	0.9	6.3	3.8	8.6	8.6
vpr	15.8	3.2	42.3	0.4	5.1	4.1	13.1	13.0
Aver	**30.9X**	**3.0**	**96.7X**	**0.8**	**4.2**	**2.5**	**13.7**	**13.7**

signal processing can lead to tools that automatically analyze massive power data, detect undesirable power behavior for higher resolution simulation, and identify suspicious system components and behaviors. We believe power signal processing provide a new perspective into automatic power analysis and optimization that will help address the two design productivity bottlenecks highlighted in Section 1.

7. References

[1] L. Zhong, S. Ravi, A. Raghunathan, and N. K. Jha, "RTL-aware cycle-accurate functional power estimation," *IEEE Trans. Computer-Aided Design*, vol. 25, pp. 2103–2117, Oct. 2006.

[2] D. Stasiak, R. Chaudhry, D. Cox, S. Posluszny, J. Warnock, S. Weitzel, D. Wendel, and M. Wang, "Cell processor low-power design methodology," *IEEE Micro*, vol. 25, pp. 71–78, Dec. 2005.

[3] T. Sherwood, E. Perelman, and B. Calder, "Basic block distribution analysis to find periodic behavior and simulation points in applications," in *Proc. Int. Conf. Parallel Architectures and Compilation Techniques*, pp. 3–14, 2001.

[4] R. Joseph, Z. Hu, and M. Martonosi, "Wavelet analysis for microprocessor design: Experiences with wavelet-based di/dt characterization," in *Proc. Int. Symp. High Performance Computer Architecture*, pp. 36–46, 2004.

[5] N. R. Potlapally, A. Raghunathan, G. Lakshminarayana, M. Hsiao, and S. T. Chakradhar, "Accurate power macro-modeling techniques for complex RTL components," in *Proc. Int. Conf. VLSI Design*, pp. 235–241, 2001.

[6] S. Ravi, A. Raghunathan, and S. Chakradhar, "Efficient rtl power estimation for large designs," in *Proc. Int. Conf. VLSI Design*, pp. 431–439, 2003.

[7] D. Brooks, V. Tiwari, and M. Martonosi, "Wattch: a framework for architectural-level power analysis and optimizations," in *Proc. Int. Symp. Computer Architecture*, pp. 83–94, 2000.

[8] Sim-Panalyzer: The SimpleScalar-Arm Power Modeling Project, *http://www.eecs.umich.edu/ panalyzer/*.

[9] W. Ye, N. Vijaykrishnan, M. Kandemir, and M. J. Irwin, "The design and use of simplepower: a cycle-accurate energy estimation tool," in *Proc. Design Automation Conf.*, pp. 340–345, 2000.

[10] M. Powell and T. Vijaykumar, "Exploiting resonant behavior to reduce inductive noise," in *Proc. Int. Symp. Computer Architecture*, pp. 288–299, 2004.

[11] R. A. DeCarlo and P. M. Lin, *Linear circuit analysis: time domain, phasor, and Laplace Transform approaches*. New York: Oxford University Press, 2001.

[12] J. Kozhaya, S. Nassif, and F. N. Najm, "A multigrid-like technique for power grid analysis," *IEEE Trans. Computer-Aided Design*, vol. 21, pp. 1148–1160, Oct. 2002.

[13] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," *Lecture Notes in Computer Science*, vol. 1666, pp. 388–397, 1999.

[14] S. M. Ross, *Introduction to probability and statistics for engineers and scientists*. New York: Elsevier Academic Press, 2004.

[15] J. L. Henning, "SPEC CPU2000: measuring cpu performance in the new millennium," *Computer*, vol. 33, pp. 28–35, July 2000.

[16] J. J. Yi and D. J. Lilja, "Simulation of computer architectures: simulators, benchmarks, methodologies, and recommendations," *IEEE Trans. Computer*, vol. 55, pp. 268–280, March. 2006.