Probabilistic Arithmetic and Energy Efficient Embedded Signal Processing

J. George, B. Marr, B. E. S. Akgul, and K. V. Palem
CREST, School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332–0250, USA
palem@ece.gatech.edu

ABSTRACT

Probabilistic arithmetic, where the i^{th} output bit of addition and multiplication is correct with a probability p_i, is shown to be a vehicle for realizing extremely energy-efficient, embedded computing. Specifically, probabilistic adders and multipliers, realized using elements such as gates that are in turn probabilistic, are shown to form a natural basis for primitives in the signal processing (DSP) domain. In this paper, we show that probabilistic arithmetic can be used to compute the FFT in an extremely energy-efficient manner, yielding energy savings of over 5.6X in the context of the widely used *synthetic aperture radar* (SAR) application [1]. Our results are derived using novel probabilistic CMOS (PCMOS) technology, characterized and applied in the past to realize ultra-efficient architectures for probabilistic applications [2, 3, 4]. When applied to the DSP domain, the resulting error in the output of a probabilistic arithmetic primitive, such as an adder for example, manifests as degradation in the *signal-to-noise ratio* (SNR) of the SAR image that is reconstructed through the FFT algorithm. In return for this degradation that is enabled by our probabilistic arithmetic primitives — degradation visually indistinguishable from an image reconstructed using conventional deterministic approaches — significant energy savings and performance gains are shown to be possible *per unit* of SNR degradation. These savings stem from a novel method of voltage scaling, which we refer to as *biased voltage scaling* (or BIVOS), that is the major technical innovation on which our probabilistic designs are based.

Categories and Subject Descriptors

B.7 [**Integrated Circuits**]: General; C.5.4 [**Computer System Implementation**]: VLSI Systems

General Terms

Design, Reliability

CASES'06, October 23–25, 2006, Seoul, Korea.

Keywords

DSP, Low Power, PCMOS, Probabilistic Arithmetic, Probabilistic Computing

1. INTRODUCTION

As embedded devices pervade and dominate mobile computing, portability ubiquitous to this domain often places stringent requirements on the energy consumed, or equivalently, the power dissipated by such platforms. Motivated by this significant concern, we have developed novel devices that compute "probabilistically." The associated computing platforms using these devices have a well-defined amount of error that is a *design parameter*, which can be traded for significant energy savings. As we have demonstrated in a recent paper (in DATE2006 [4]), applications that naturally embody probabilistic behavior lend themselves to orders of magnitude gains in energy and performance through architectures that use probabilistic CMOS (or PCMOS) technology. Specifically, in previous work *thermal noise* has been the basis for inducing probabilistic behavior in CMOS devices yielding PCMOS [3]. Thus noise, which was viewed as an "impediment" to sustained scaling of CMOS device feature sizes, was instead treated as a "resource" to enable ultra-low energy, probabilistic computing architectures. In this paper, we extend these insights and principles of probabilistic design along three significant dimensions to help realize embedded architectures for enabling digital signal processing (DSP) while providing an entirely new way of realizing significant energy savings.

First, we show that *probabilistic arithmetic* elements, such as adders and multipliers, can be built to realize energy-efficient computing elements, which can in turn be used to create more complex primitives for computing (such as the FFT). The insight that helps explain these energy savings gained through probabilistic elements is characterized as the *energy-probability relationship* (detailed in Section 3.1). Briefly, the probability of correctness of output bit i of an adder, denoted p_i, can be decreased to increase the associated energy savings, and vice versa. Through the energy-probability relationship, we will show that the energy savings follow from the fact that the amount of energy (ΔE) saved is significantly more than the amount of probability (Δp) traded in. We therefore exploit this *energy-probability trade-off* to realize probabilistic and extremely energy-efficient arithmetic elements.

Second, we apply this novel probabilistic design methodology to PCMOS, wherein device-level errors induce probabilistic behaviors due to *noise*. Numerous studies demonstrate the widely recognized impediments posed by noise in future CMOS devices [5, 6, 7, 8]. Specifically, these studies imply that as a result of noise, device and circuit behaviors are expected to be probabilistic around 2016 (ITRS road map [9]). Thus, a major contribution in this paper involves extending our noise-induced probabilistic devices used to realize architectures for applications with probabilistic content [4] to also encompass conventional computational steps and primitives such as adders and multipliers. *In our current context, probabilistic behaviors are induced in computing elements such as gates and circuits composed of them, by lowering the operating voltage V_{dd} to a magnitude (e.g., $1.1\,V$ in* TSMC *$0.25\mu m$ technology) comparable to noise levels available in the circuit, resulting in elements that do not compute correctly all of the time.* To reiterate, we will refer to devices that exhibit these behaviors as *probabilistic* CMOS, or PCMOS for short.

The crucial idea is that these PCMOS computing elements that fail do so in a probabilistic manner. Rather than classifying them as being faulty or correct (non-faulty), we explicitly quantify the *amount* by which they are faulty through various metrics. It is also important to note that we observe similar energy-probability trade-offs when the PCMOS devices are realized through *voltage over-scaling* [10]. In this case, V_{dd} is scaled down to induce errors due to *propagation delays*: the output of the computing element is sampled at a higher clock rate than the clock rate at which the element is expected to operate. Such a scheme would induce probabilistic behaviors in current technologies, where noise is not a serious impediment yet, realizing probabilistic computing and achieving significant energy savings in return. We compare the energy-probability trade-offs for noise-induced and over-scaled PCMOS adders and multipliers in Section 3.

Our approach is based on a novel *biased voltage scaling* (or BIVOS) scheme. In our BIVOS approach, each bit being computed has a *profit* associated with it (the bit significance) as well as an associated *investment* (energy cost). Thus, referring to Figure 1(a), the profit associated with the outputs (or inputs) of a 32-bit PCMOS adder are greater as we move from bits of lower significance to those of higher significance; it is the highest at the most significant bit. Also as shown in the figure, the operating voltage, and hence the investment, is varied through the BIVOS scheme wherein it is higher in the context of the (most significant) bits that yield higher profits and is correspondingly lower in the context of those that yield lower profit. Thus, the BIVOS approach implies non-uniform bit error rates among the individual bit positions of the adder, with the most significant bits (MSBs) having lower bit error rates and the least significant bits (LSBs) having higher bit error rates. In contrast, conventional voltage scaling [11, 12] entails lowering the voltage, and hence increasing the bit error rate, equally across all bit positions.

Third, with these innovations as our foundation, we have demonstrated that these probabilistic adders and multipliers realized using PCMOS can serve as an entirely novel approach to realizing low-energy DSP co-processors. Our architectural framework will be a *p*robabilistic *s*ystem-*o*n-a-*c*hip (PSoC) architecture [4]. Briefly, the application will be partitioned in a manner where the core control-flow such as branches will be executed on the "host" processor whereas the signal processing *kernels* will be executed on the probabilistic co-processor.

Using this PSoC framework, in this paper we show the impact of our approach at the application level in the context of *synthetic aperture radar* (SAR) imaging. In this instance, the co-processor realizes a FFT where the probability of correctness p at the device level manifests itself naturally as the signal-to-noise-ratio (SNR) at the level of the DSP primitive. In Section 6, we show dramatic energy savings and performance gains using probabilistic adders and multipliers in the context of SAR: energy savings up to 5.6X achieved with minimal degradation in SNR and the corresponding image quality. If the running time is also accounted for through the energy-performance-product (EPP), the corresponding gain is 2.25X.

1.1 Related Work

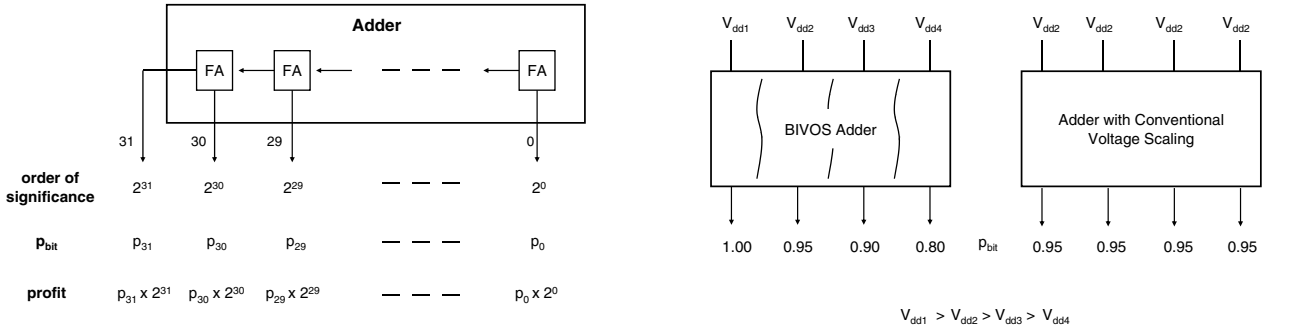
The foundations of PCMOS technology are rooted in physics of computation, algorithms, and information theory. Earlier, using techniques derived from physics of computation and information theory, Palem [13, 14] showed that the thermodynamic cost of computing a bit of information irreversibly is directly related to its probability p of being correct. Further, using an abstract model of computation, it has been demonstrated that switch level energy savings can be harnessed at the application level to construct probabilistic algorithms.

In previous work, the application of PCMOS at the device level (switch behavior) [15] and at the architecture level (probabilistic applications and PSoC) [4] was demonstrated to be feasible. The specific algorithms that were studied in this earlier work included Bayesian networks, random neural networks, and hyper-encryption. These algorithms span embedded application domains such as face and pattern recognition, spoken alphabet recognition, and computer security. This principal of probabilistic computing can be further expanded into error-tolerant applications in the form of building blocks (adder, multiplier), primitives such as the fast Fourier transform (FFT), and applications that inherently tolerate error such as image decoders and radar.

2. PROBABILISTIC ARITHMETIC

In this section, we introduce and study the novel concept of *probabilistic arithmetic*, which will serve as the theoretical basis for the signal processing primitives studied in the sequel. Informally, a *probabilistic arithmetic operation* is an operation where each the i^{th} bit of the computational primitive—addition and multiplication studied in this paper—has an associated probability of correctness, p_i. A k bit probabilistic arithmetic operation is a function $\mathcal{O}_P$ where $\mathcal{O} : \{0,1\}^k \times \{0,1\}^k \rightarrow \{0,1\}^l$ is a function and $P =< p_0, p_1, \cdots, p_{l-1} >: 0 \leq p_i \leq 1$ is the *probability parameter* where p_i corresponds to the probability that the i^{th} bit of the output is correct. The case where $P \equiv< 1 >$ corresponds to a conventional (deterministic) function.

Given a probabilistic arithmetic operation, two characteristics of interest can be studied: the *degradation* and the *gain*. Informally, the output of probabilistic arithmetic operation would be "incorrect" when compared to its deterministic counterpart; namely, there will exist bits where $p_i \neq (0 \leq i \leq l-1)$ resulting in an impact at the application level that is captured by the degradation characteristic. This application-level impact, and hence the degradation



(a) (b)

Figure 1: Adder with voltage scaling: (a) Impact of bit significance and bit probabilities on error magnitude at the output of a 32-bit adder. (b) Comparing BIVOS and conventional voltage scaling based adder examples with associated output bit probabilities and V_{dd} lines.

characteristic, is application dependent. For example, in the subsequent sections, we demonstrate that in specific digital signal processing applications, when the probabilistic addition operation is considered, certain probability parameters have no visual impact on the resulting signal. The second characteristic of interest is the *gain*, or specifically the gain in terms of energy savings. Given a monotonically increasing relationship between energy E and the probability of correctness p of a primitive one bit boolean operation, it must be the case that a probabilistic arithmetic operation is energy efficient when compared to its deterministic counterpart. The gain characteristic captures this energy efficiency. This notion of a probabilistic arithmetic operation with its associated degradation and gain characteristic will be illustrated through the *probabilistic addition function* example below.

2.1 Probabilistic Addition Function: A Case Study

Informally, the output of a k bit probabilistic addition function, $\mathcal{A}_P$, is the arithmetic sum of two k bit boolean values such that the i^{th} bit of the output is correct with a probability p_i. Given the deterministic addition operation, $\mathcal{A}_D$ and given an input $I_{2k} \in \{0,1\}^{2k}$, let the i^{th} bit of the output be y_i. If the degradation of this addition operation is defined as the difference between the output of $\mathcal{A}_P$ and $\mathcal{A}_D$, the expected degradation given an input I_{2k} is

$$\sum_{i=0}^{i=l-1} \{\bar{y}_i \times (1-p_i) - y_i \times (1-p_i)\} 2^i \qquad (1)$$

where $\bar{y}_i$ is defined to be the logical negation of y_i. The energy consumption of the probabilistic arithmetic operation can then be defined as

$$\mathbf{E}(\mathcal{A}_\mathcal{P}) = \sum_{i=0}^{i=l-1} E(p_i) \qquad (2)$$

where $E(p_i)$ relates the probability of correctness of an individual bit to the energy consumed in computing that bit. From Equation 1 and Equation 2, and for a given energy consumption, the degradation of the probabilistic adder can be reduced by altering the probability values p_i. Intuitively, since the higher order bits contribute more to the degradation than the lower order bits (since they are scaled with higher powers of 2 from Equation 1), minimal degradation can be achieved by higher probability of correctness for higher order bits. This we will refer to as "biasing" and we will quantify and expand upon in the subsequent sections.

To briefly highlight the opportunity of our approach, here, we compare two adder realizations based on the BIVOS approach (Figure 1(a)) and the conventional (uniform) voltage scaling approach (Figure 1(b)) with the *same* energy budget yielding equal energy savings relative to the (nominal) deterministic case. As shown in Table 1, the (average) deviation or the error magnitude at the output of the adder in case of BIVOS , corresponding to $2^{31} \times (1-p_{31}) + 2^{30} \times (1-p_{30}) + ... + 2^0 \times (1-p_0)) = 55,731$ for $p_{31..20} = 1$, $p_{19..16} = 0.95$, $p_{15..8} = 0.90$, $p_{7..0} = 0.80$ is much less than the case of conventional voltage scaling, corresponding to $(2^{31} + 2^{30} + ... + 2^0) \times (1-p_{bit}) = 214,748,364.75$ for $p_{bit} = 0.95$. Similarly, when we compare the worst case error magnitudes from the table, we see that BIVOS approach gives a smaller deviation $(2^{20}-1)$ than the conventional voltage scaling approach $(2^{32}-1)$. Note that, in our example here, p_{bit} and $p_{31}, p_{30}, ...p_0$) are configured such that the two adder types consume the same amount of energy, which is $16pJ$. The benefit to our approach is due to the fact that (as shown in Figure 1(a)) the MSBs with higher probability of correctness yield higher profit, which, to reiterate, is proportional to the bit significance: the error magnitude associated with position b_{31} can be as high as 2^{31}, whereas if the error were to occur at position b_0, the associated magnitude would be $2^0 = 1$.

Table 1: Comparing the benefits of a BIVOS based probabilistic adder implementation with a conventional voltage scaling based adder implementation for the *same* energy budget with bit probabilities $p_{31..20} = 1$, $p_{19..16} = 0.95$, $p_{15..8} = 0.90$, $p_{7..0} = 0.80$ in the BIVOS case and $p_{31..0} = 0.95$ in the conventional voltage scaling case.

	Adder with Conventional Voltage Scaling	Adder with BIVOS
Energy consumed	$16pJ$	$16pJ$
Average error magnitude	$214,748,364.75$	$55,731$
Worst case error magnitude	$2^{32}-1$	$2^{20}-1$

3. CHARACTERIZING THE RELATIONSHIP BETWEEN ERROR AND ENERGY IN PCMOS BASED COMPUTING

As a technology, PCMOS is voltage scaled CMOS technology subject to a source of noise resulting in an output that is correct with a probability p [2, 16, 17]. Thus, the quantity $(1-p)$ is an indicator of the amount of error in the output. With this in mind, certain applications (DSP among them) are capable of tolerating occasional errors caused by probabilistic device behavior. Through PCMOS we capitalize on these characteristics by trading deterministic behavior for substantial energy savings. This trade-off can be exploited to "tune" error rates or probability of correct operation in return for energy savings. In the following subsection, we briefly describe this energy-probability relationship as a basis to trade energy savings for a lower value of correctness (and vice versa) in the context of a simple inverter switch.

3.1 A Review of the Potential Energy Savings for an Inverter Switch

At the switch level, the energy per switching step grows as a function of p and is lower bounded by an exponential over the probability range of $0.5 < p < 1$ [18]. Accordingly, where $p \sim 1$ the slope of the exponential is steep and energy per switching step grows rapidly with small gains in p. In Figure 2 [19] we show this behavior, which offers significant potential for energy savings by compromising small amounts of probability of correctness, Δp, in exchange for large reductions in energy consumption, ΔE. Shown in Table 2 [19], trading 0.22% in probability of correctness (p reduced from $p = 0.9990$ to $p = 0.9968$) results in a disproportionate 23% reduction in energy per switching step. Further reducing p to 0.9827 yields 39% in energy savings at an expense of 1.4% in the value of p.

The above characterization is crucial to designing probabilistic switches and to achieving the corresponding energy gains at the switch level. A contribution of this work is to extend this relationship between the probability of correctness and the energy consumed to higher levels in the design hierarchy: at the level of arithmetic primitives and at the application level. For this purpose, in the next subsection we introduce the metrics that will be used in the rest of the paper to characterize error and quantify corresponding gains at higher levels in the design hierarchy.

3.2 Metrics for Characterizing Error in Arithmetic Primitives

In a typical digital design hierarchy within the context of digital signal processing (DSP), switches and basic gates (devices) are combined to form digital building blocks, including adders and multipliers. These building blocks are then used to realize algorithmic primitives, such as filters and transforms, which in turn form the basis for DSP . Errors correspond to different types of degradation at each level of this hierarchy. For example, the error rate at the level of an arithmetic building block is manifested as signal-to-noise ratio (SNR) at the level of an algorithmic primitive, such as a fast Fourier transform (FFT). This SNR value further propagates to the application level and causes quality degradation, such as image distortion.

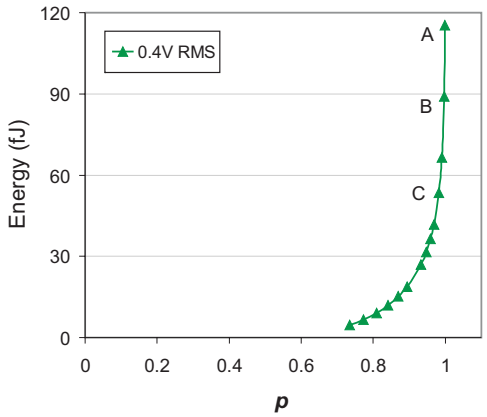


Figure 2: Energy-probability relationship of a PCMOS inverter designed in TSMC 0.25μm technology with a noise magnitude (RMS) value of $0.4V$.

To evaluate PCMOS building blocks, it is first necessary to establish a definition for probability of correctness in the context of a building block, where the errors are weighed according to their impact at the resulting output, due to bit significance. To address this issue, a definition of p at the level of a digital building block (e.g., an adder) is considered based on a threshold δ. Informally, the threshold δ designates a maximum tolerable deviation in magnitude between the expected and observed output of the primitive. Thus, errors that do not cause the magnitude of the output to vary from the correct value by more than δ in magnitude are *considered to be correct*. Considering k outputs of a primitive, the *threshold magnitude error* rate (TME) is defined simply as the ratio of the number of deviations by more than δ from the correct value to the total number of outputs. Thus, the

Table 2: Energy-probability relationship of a PCMOS inverter designed in TSMC 0.25μm technology with a noise magnitude (RMS) value of $0.4V$.

p	E	Δp	ΔE	Energy Savings	p Sacrifice
0.9990	$115fJ$	-	-	-	-
0.9968	$89fJ$	0.0022	$26fJ$	23%	0.22%
0.9827	$54fJ$	0.0141	$35fJ$	39%	1.4%

probability of correctness of the building block, denoted p_δ, is equal to $(1-\text{TME})$.

Informally, the threshold value δ is meant to serve as a guide to aid in designing a probabilistic computing element, and is derived from the needs of the application where the element is being used: if the application domain can tolerate significant error magnitudes, then the δ value can be high, allowing for a more aggressive BIVOS approach to achieve energy savings.

4. CASE STUDY OF A PCMOS ADDER

4.1 Adder Design Using BIVOS Approach

The objective of minimizing energy consumption, while simultaneously maximizing p_δ, can be achieved by considering the fact that *not all bit errors are created equal*. While an error in the least significant bit of an output sample minimally affects the difference in magnitude between the erroneous result and the expected result, a bit error in the most significant bit of an output value can cause a difference on the order of the range of the number set. Furthermore, even with low bit error rates, the SNR value and image distortion are significantly affected by bit errors at bits of higher significance resulting in large magnitude differences. With conventional voltage scaling, where all bits are treated with equal significance, there is little opportunity to mitigate this effect.

In contrast, the probabilistic arithmetic through the "biasing" model introduced in Section 2, which we refer to as *biased voltage scaling* (BIVOS), allows us to do significantly better. In this scheme, we apply voltage scaling within building blocks, depending on bit significance, providing the elements that calculate the more significant bits a higher voltage (and a thus higher p). The elements that compute bits of lower significance have a lower voltage (and a lower p). This concept is illustrated in Figure 3(a) for an $n-bit$, ripple-carry adder. The full-adder elements that compute each bit are assigned a supply voltage V_i based on bit position i where $V_i > V_{i-1}$. For each bit, V_i is chosen to establish a desired p_i based on experimental mapping between V and p. Inverter pairs are then inserted between bits to mitigate static current effects arising from the voltage differential. As shown in the example, gates that compute the output for a single bit are provided the same supply voltage. To reiterate, the result of this voltage biasing scheme is that bit errors are more likely to occur in the least significant bits, and more costly bit errors, occurring in the more significant bits, are less likely to occur [20]. This BIVOS approach constitutes the main technical contribution of this paper.

As a design tool for establishing supply voltage allocation schemes, we now introduce an example non-uniform (or BIVOS) supply voltage distribution where p is determined based on a geometric function of bit significance as shown in Equation (3).

$$p_i = \begin{cases} p_0 & : \quad i = 0 \\ p_{i-1} + ar^{i-1} & : \quad i \neq 0 \end{cases}$$

where

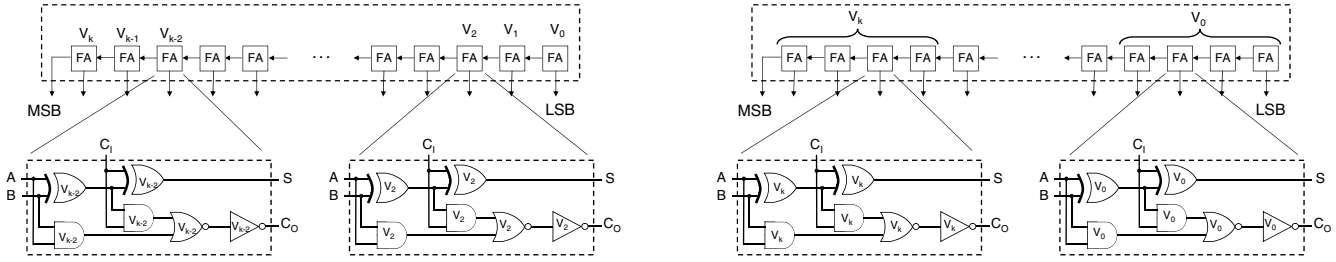
$$\begin{aligned} p_0 &= \text{probability of correctness of output bit 0} \\ p_i &= \text{probability of correctness at output bit } i \\ i &= \text{bit position} \\ a &= \text{scale constant} \\ r &= \text{ratio constant} \end{aligned} \tag{3}$$

The ability to individually tune each bit in a building block to a different voltage level can, through this scheme, yield an optimal design. However, having an unlimited number of supply voltages may not be practical as this constitutes additional cost in terms of area expense in routing the supply lines and in realizing on-die voltage conversion.

To address these significant energy considerations, we propose supply voltage binning. Rather than providing a distinct supply voltage to each bit position, we propose bits be grouped with neighbors to form bins. Informally, all the bit positions within a bin are considered to have the same significance. The supply voltage distribution among the bins is then realized such that the most significant bins receive high(er) supply voltages and the least significant bins receive low(er) supply voltages. The bins can contain a nonequal number of bits, and the energy consumed is minimized for a given error rate by optimally selecting these bins [20].

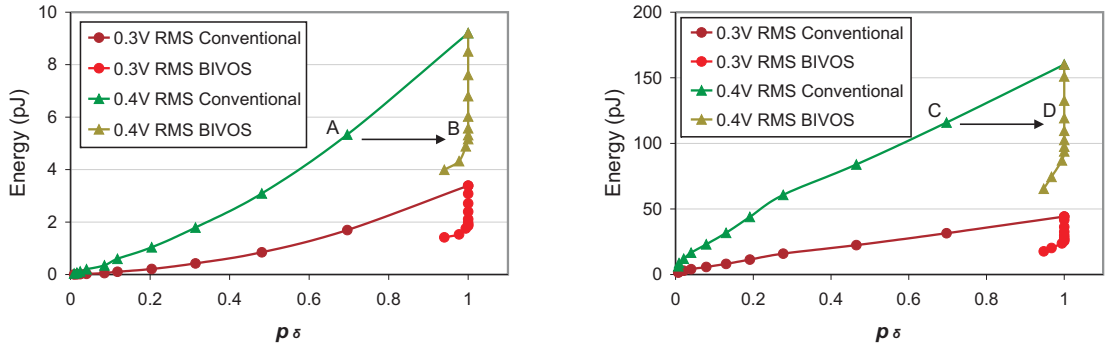
An example of supply voltage binning is shown in Figure 3(b) for an $n-bit$ ripple-carry adder. Only two four-bit voltage bins are shown in the figure for bin 0 and bin k; it is assumed that bins are defined for the remaining bits. Also here, all the gates used to compute output bits 0 through 4 are grouped into bin 0 and are provided a single supply voltage. Similarly, all the gates forming output bits at positions $n-3$ through n are grouped into bin k and provided supply voltage $V_k >> V_o$. Inverter pairs are again inserted, between bins in this case, to mitigate static current effects. This voltage binning concept ensures that bit errors are more likely to occur in the least significant bits and not in most significant bits, however, this is accomplished using a *limited number of supply voltages*.

As shown in Figure 4(a), the BIVOS approach based on binning significantly improves the $E-p$ relationship of an adder, specifically when compared to conventional voltage scaling. The BIVOS approach drastically improves p_δ for a fixed energy budget, resulting in a p_δ value of 1 for almost any energy level.



(a) (b)

Figure 3: Biased voltage scaling (or BIVOS) example: (a) an $n-bit$, ripple-carry adder employing BIVOS with distinct voltage supplies for each 1-bit full adder and (b) an $n-bit$, ripple-carry adder employing BIVOS with voltage binning.



(a) (b)

Figure 4: Improvement in the $E-p_\delta$ relationship through BIVOS: (a) $E-p_\delta$ relationships of conventional voltage scaling and BIVOS based adders for noise RMS values of $0.3V$ and $0.4V$. Points A and B, respectively, refer to a fixed energy budget of $5.3pJ$ with a $p_\delta = 0.7$ in the case of a conventional voltage scaling based adder and a $p_\delta = 1.0$ in the case of a BIVOS based adder. (b) $E-p_\delta$ relationships of conventional voltage scaling and BIVOS based multipliers for noise RMS values of $0.3V$ and $0.4V$. Points C and D, respectively, refer to a fixed energy budget of $115pJ$ with a $p_\delta = 0.7$ in the case of a conventional voltage scaling based multiplier and a $p_\delta = 1.0$ in the case of a BIVOS based multiplier.

4.2 Experimental Framework

As described in Section 1, currently the probabilistic behavior of PCMOS devices is derived from interactions of primary input/output signals with noise sources. In order to model thermal noise, we followed the approach described in [3, 21] by using a noise source that is a random variable where the magnitude follows a Gaussian distribution with a mean of zero and a standard deviation σ, referred to as the root-mean-square (RMS) value. This noise was coupled at the sum and carry bits of the adders to induce probabilistic behavior. The resulting signals were then compared with the expected values to determine p.

Energy consumption for the arithmetic unit was calculated per clock cycle as shown in Equation (4).

$$E = \frac{I_{avg} \cdot V_{dd} \cdot \Delta t}{s}$$

where

$$\begin{aligned} I_{avg} &= \text{average current} \\ V_{dd} &= \text{supply voltage} \\ \Delta t &= \text{time interval} \\ s &= \text{sample count (or number of clock cycles)} \end{aligned} \tag{4}$$

Characterization of the probabilistic behavior of a 12-bit, ripple-carry adder in the conventional or uniform voltage scaling context was performed through HSpice simulation using libraries for the TSMC $0.25\mu m$ technology. Random noise sequences with a Gaussian noise distribution were first generated using Matlab. Input sequences with a uniform input distribution of $\{0, 1\}$ were also generated using Matlab. Each building block was then simulated in HSpice over a range of $1.1V$ to $2.5V$ supply voltages with noise RMS values ranging from $0.2V$ to $0.4V$.The noise sampling rate was chosen to be larger than two times the longest propagation delay of the worst case component used in any building block. In our case, the component with the largest propagation delay (with $1.1V$ supply voltage) was a Type II full adder in the array multiplier.

The output sampling rate was correspondingly chosen to allow 15 noise samples for every data transition. This translated to a noise sampling rate of $50MHz$ $(1/20ns)$ and an output sampling rate of $3.33MHz$ $(1/300ns)$. All simulations were performed with input sequences of 1000 data points at each input and 15000 noise points at each noise source. Each simulation run was then post processed to determine the corresponding p_δ and energy consumption.

Characterization of the probabilistic behavior of the 12-bit, ripple-carry adder in the BIVOS context was performed through behavioral C simulations based on HSpice data. Simulation was performed over a range of geometric supply voltage distributions, chosen by varying the constant r (Equation 3) from 1 to 20 in increments of 1 and by varying a from 0 to 0.2 in increments of 0.001 (also Equation (3)).

The associated energy consumption was calculated by extrapolating the values derived from HSpice simulations for the uniform voltage scaling case. Since the uniform voltage scaling characterization data provided discrete $E - p$ pairs, it was necessary to interpolate the $E - p$ relationship to match the non-uniform supply voltage equations (Equation 3). This was accomplished by performing an exponential curve fit to the uniform voltage scaling $(E - p)$ data using Matlab.

The $E - p$ equations were then used to determine the total energy consumption of the adder at specific values of p. As shown in Equation (5), the energy consumption due to a specific bit in the ripple-carry adder was simply based on the energy consumption ratio for that bit, where the energy consumption ratio is defined as the number of elements (1-bit full adders inside the ripple carry adder in our case) needed to determine the bit value divided by the total number of elements in the adder.

$$E = \sum_{i=0}^{k} e_i \cdot \left(\frac{g_i}{N}\right)$$

where

e_i energy consumption for the entire component at p for bit i
g_i is the number of elements computing bit i
N is the total number of elements in the component
k is the total bits in the component

(5)

5. PCMOS MULTIPLIERS AND DSP PRIMITIVES

Characterization of the probabilistic behavior of a 6-bit, tri-section, two's complement array multiplier [22] was performed through HSpice and C simulation using TSMC $0.25\mu m$ technology in the same fashion as the ripple-carry adder (described in Section 4.2). As in the case of the ripple-carry adder, biased voltage overscaling drastically improves the probability of correctness, p_δ, over conventional voltage overscaling (Figure 4(b)).

Characterization of the probabilistic behavior of a 6-bit, 4-point FFT primitive was performed through behavioral C simulation based on the arithmetic building block characterizations derived from HSpice. The FFT algorithm was decomposed into building blocks constituted of adders and multipliers. Bit errors were then injected on the output of each building block at a rate determined by HSpice simulation. The results from each building block were then combined and propagated through the other building blocks of the primitive. This was repeated for each instance of supply voltage and noise RMS values simulated in building block characterization. To determine p_δ, each simulation was repeated 1000 times using uniformly distributed random data points at each input, and observed probabilistic outputs were compared with the expected (deterministic) outputs.

To estimate primitive energy consumption the average switching energy of each building block was first calculated. The average energy consumption for a building block over an entire simulation run (as determined in HSpice characterization) was divided by the number of output samples over that run to generate the average switching energy. The energy consumption for each building block was considered to be additive and the average switching energy of the primitives was derived based on the building block composition

of each primitive (Equation (6)).

$$E_{primitive} = \sum_{i=0}^{k} x_i \cdot E_i$$

where

i denotes a specific building block (adder, multiplier, etc.)
x_i is the count of i building blocks employed in the primitive
k is the number of different building blocks employed in the primitive
E_i is average switching energy for building block i

(6)

Characterization of the probabilistic behavior of the FFT for non-uniform supply voltage distributions was also performed through behavioral C simulations (as described in Section 4.2). After determining the corresponding p values for each bit position of the building block outputs, bit-errors were injected on correct results accordingly. The results from each building block were then combined and propagated through the other building blocks of the primitive. Each simulation was run over 1000 random, uniformly distributed data points at each input, and observed outputs were compared with the expected (deterministic) outputs to determine the resulting SNR.

6. PUTTING IT ALL TOGETHER

To analyze the value of probabilistic arithmetic in an application context we have performed experiments using *synthetic aperture radar* (SAR) imaging. A satellite image of Los Angeles County was used for experimentation wherein an ERS-1 satellite captures a SAR image, and the data received by the image processing unit is approximated by the 2-dimensional convolution of the image with the carrier frequency of the satellite [23]. The transmitted carrier waveform, s(t), as a function of time from the ERS-1 satellite is given by:

$$s(t) = e^{(\mathrm{j}(\omega_0 t - \beta t^2))} \; for \mid t \mid \leq \tau_p \tag{7}$$

where the frequency is:

$$f(t) = \frac{\omega_0 - 2\beta t}{2\pi} = 5.29GHz \tag{8}$$

and the bandwidth is:

$$B = \frac{\beta \tau_p}{\pi} = 15.5MHz \tag{9}$$

The outer product of this transmitted frequency modulated (FM) waveform approximates a 2-dimensional signal, called a *chirp*, that is convolved with the image data resulting in an approximation of the input data received by a SAR processor. The SAR processor itself is then approximated by a simple matched filter [1].

Thus, a simple experiment was performed where the data, created using the ERS-1 satellite parameters, was processed with CMOS based conventional arithmetic in one trial and PCMOS based probabilistic arithmetic in the other. The matched filter operation is simply convolution between the input data and a *matched* filter, where convolution is implemented by multiplication in the frequency domain. The output of this matched filter is the recovered image. The matched filter operation was implemented by a applying a FFT on the synthetic data, multiplying this data by the frequency representation of our filter, and by applying an inverse FFT operation on this product to recover the original image. The adder and multiplier units that comprise the FFT were implemented with PCMOS. In the case of probabilistic arithmetic, addition and multiplication were implemented with the BIVOS technique described in this paper.

6.1 Results

The notable improvement in p_δ, detailed in Sections 4 and 5 in the context of probabilistic arithmetic primitives, in turn yields noticeable SNR and image distortion improvements at the application level. Evident in SAR , BIVOS based probabilistic arithmetic and DSP implementation results in significant energy savings with minimal impact on application quality (Figure 5). In Figure 5(a) an original image of downtown Los Angeles was derived from standard SAR processing. Figure 5(b) shows this same image derived using SAR processing employing conventional voltage scaling. In Figure 5(c), we show the same image derived with BIVOS based probabilistic (PCMOS) arithmetic. The standard SAR processing and BIVOS based results are visually indistinguishable, whereas the latter yields 5.6x in energy savings.

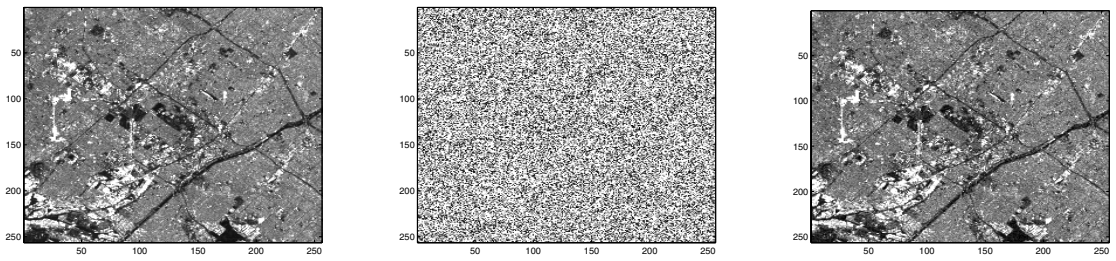
Additionally, we also examine the efficiency of our approach through the metric of *e*nergy (measured in Joules) *p*erformance (measured in seconds) *p*roduct (EPP), as well as the EPP per dB gain, and summarize our findings in Table 3: non-uniform voltage scaling is far less expensive in terms of the EPP cost per dB, denoted EPP / SNR. Shown in Table 3, non-uniform voltage scaling yields an EPP / SNR expense of 15.7×10^{-3}, compared to $+\infty$ in the case of uniform voltage scaling.

7. PROBABILISTIC ARITHMETIC IN CURRENT AND FUTURE TECHNOLOGIES

As evident from the influential ITRS road map, considerations of noise will pervade the design of CMOS circuits and concomitant computing platforms starting in 2016. Delays due to voltage over-scaling, on the other hand, mostly concern today's technologies where circuits are intentionally operated at low supply voltages with the objective of saving energy. In this section, we show that our BIVOS methodology can be applied to PCMOS computing elements where probabilistic behavior is induced not only by noise—the subject of this paper—but also due to voltage over-scaling with the example of a probabilistic adder.

7.1 Future Technologies and Noise

As transistor feature sizes scale down into the nano-regime, CMOS circuits become increasingly susceptible to error due to noise [6, 8, 24, 25]. Judging by ITRS projections for future supply voltages and thermal, switching, and cross-talk noise levels projected to be present, the noise to signal ratio (NSR)—previously defined by Cheemalavagu et. al [26]—will be high enough to cause significant errors by 2016 [9]. For example, switching noise, also known as power supply noise, is as high as $30mV$ [6] in current technology. Furthermore, as studied by Elgamel [25], cross talk can be as high as $1.4V$ when power and ground lines are not used to shield interconnect. Finally, the ITRS road map projects that supply voltages will scale down to $0.5V$ ($\pm 10\%$ or more) by



(a) (b) (c)

Figure 5: Application level impact of our approach on SAR: (a) Original image of Downtown Los Angeles, (b) image of Downtown Los Angeles with conventional voltage scaling yielding 2.5X energy savings with an SNR value of 0dB, and (c) image of Downtown Los Angeles with BIVOS based probabilistic arithmetic yielding an acceptable SNR of 28 dB and 5.6X energy savings.

Table 3: SAR Performance

Voltage Scaling Scheme	SNR	Energy	Running Time	EPP	EPP / SNR
BIVOS	$28dB$	1/5.6X	2.5X	0.44X	15.7×10^{-3}
Uniform Voltage Scaling	$0dB$	1/2.5X	1.41X	0.56X	$+\infty$

2016 [9]. Coupled with this scaling of supply voltage, noise levels will rise in future technology generations [8]. In a recent work [27], noise levels are conservatively projected at $60mV$ based on simulation at a feature size of $70nm$ where the experiments are conducted with supply voltage of $0.15V$. Based on an NSR value of $\frac{60mV}{0.5V} = 0.12$ in the case of a V_{dd} value of $0.5V$ or NSR of $\frac{60mV}{0.15V} = 0.4$ in the case of a V_{dd} value of $0.15V$, we have studied the relative effect of noise on future technology generations. This translated to a supply voltage range of $1.1V$ to $2.5V$ and a noise RMS ranges of $0.3V$ to $0.4V$. Under these operating conditions, namely, with a noise RMS value of $0.3V$ and $0.4V$, the $E-p$ relationship of a noise-induced PCMOS adder is shown in Figure 6(b). In conclusion, as CMOS transistor sizes scale down, they will naturally behave as PCMOS devices resulting in the corresponding primitive having a probabilistic output.

7.2 Current Technologies and Over-scaling

We also investigate propagation delay as a source of noise that is applicable to today's technologies. Consider a 32-bit, ripple-carry adder as an example. The output for a given bit is the result of data propagating through a series of full adders, up to 32 depending on the input set and the significance of the bit in question. In conventional CMOS design, this propagation delay along the *critical path* determines the upper bound for the ripple-carry adder's clock frequency. Rather than determining the clock frequency based on critical path delay (which will be the delay for only a small fraction of the input set), we propose setting the clock frequency such that outputs will switch within the given clock period with a probability, p. Thus propagation delay becomes the source of error, and PCMOS technology can be used to tradeoff energy consumption versus error rate through the novel approach of computing at a clock rate that is higher than the speed at which devices might be switching.

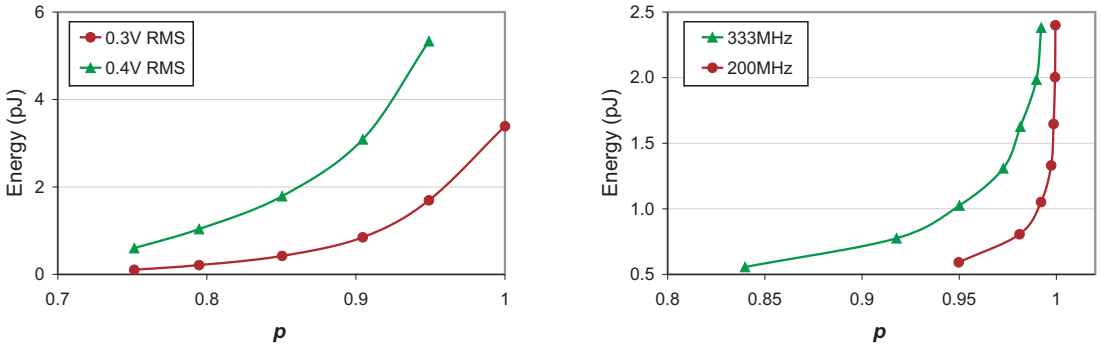
Therefore, analogous to the case of noise-induced PCMOS devices (as seen from Figure 6(a)), PCMOS devices can also be made 'probabilistic" due to voltage over-scaling. From Figure 6(b), we see that an increased performance constraint for a fixed energy budget (moving from point C to D as shown in the figure) causes the probability of correctness to decrease, namely, the $E-p$ curve to shift to the left. This effect is due to the fact that at a higher clock rate (moving from 200MHz to 333MHz), the circuit will yield more errors due to the switching speed of the arithmetic primitives being slower than the clock speed, and hence the probability of correctness will decrease.

8. REMARKS AND NEW RESEARCH DIRECTIONS

This work introduced an entirely novel notion of probabilistic devices that are controlled by voltage scaling, yielding a novel class of PCMOS devices and building blocks. Based on this, the concept of probabilistic arithmetic was introduced and shown to be effective in realizing energy efficient signal processing, specifically for an FFT. This led to the novel BIVOS approach for designing PCMOS based probabilistic arithmetic primitives.

We have compared and shown the connection between two phenomenons, namely noise-induced probabilistic behavior and delay-induced probabilistic behavior, in realizing energy-efficient PCMOS designs in Section 7.2. Although this paper has established the viability using noise-induced models as opposed to using models for delay-induced errors due to over-scaled CMOS , our study shows the potential for over-scaled CMOS to realize energy-efficient designs in today's technologies while we wait for the noise-induced phenomenon in future technologies.

While delay, area, and power consumption are all design metrics that offer tradeoffs, BIVOS based PCMOS designs must further consider propagation paths of building blocks in particular when propagation *delay* is the source of probabilistic behavior. Comparing ripple-carry and carry-skip adders for instance, carry-skip adders offer faster propagation delays, but at the expense of power consumption and die



(a) (b)

Figure 6: Comparing the energy-probability trade-offs in case of (a) a noise-induced PCMOS adder and (b) a PCMOS adder induced probabilistic due to voltage over-scaling. Moving from point A to B, i.e., from an adder operating at $200MHz$ to that at $333MHz$, probability of correctness degrades, which is similar to moving from point C to D, i.e., from an adder operated under noise RMS value of $0.3V$ to that of $0.4V$.

area [28]. Carry-skip adders, however, have the opportunity to skip entire sections of carry propagation and mitigate the potential for a propagation error. Further, there are many other building block designs that require consideration for the viability of BIVOS based PCMOS that we leave for future research.

Additionally, different building blocks have vastly different propagation characteristics. Array multipliers, as an example, have a long potential propagation delay through the middle of the structure. Ripple-carry adders, on the other hand, have the longest potential propagation delay path at the most significant bit. When propagation delay is the source of errors, these varying propagation paths between building blocks can impact p on a per-bit basis. Further study is needed to address how these bit errors along the propagation path impact the error rates at high order bits.

Finally, when implementing BIVOS based voltage scaling there is a practical concern of the cost associated with the additional voltage generation and supply routing necessary to provide multiple supply voltages. Grouping bits into voltage bins was proposed as a solution to this problem, however, there is a trade-off between the cost of generating additional supply voltages and the ability to precisely tune a BIVOS scheme to meet performance characteristics. A study of binning schemes remains as a topic for future research.

Thus, our work is a first step in investigating an entire area of probabilistic design for arithmetic logic and its value to energy efficient embedded computing. In this context, we envision signal processing enabled video, image processing, and audio processing to be studied as candidate domains of applicability.

Acknowledgments

This work is supported in part by DARPA under contract #F30602-02-2-0124, by the DARPA ACIP program under contract #FA8650-04-C-7126 through a subcontract from USC-ISI, and by awards from Hewlett-Packard and Intel Corporation.

9. REFERENCES

[1] M. Richards, *Fundamentals of Radar Signal Processing*. McGraw Hill Publishing, 2005.

[2] K. V. Palem, L. N. Chakrapani, B. E. S. Akgul, and P. Korkmaz, "Realizing ultra low-energy application specific soc architectures through novel probabilistic CMOS (PCMOS) technology," in *Proc. of the International Conference on Solid State Devices and Materials*, Tokyo, Japan, Sept. 2005, pp. 678 – 679.

[3] S. Cheemalavagu, P. Korkmaz, K. V. Palem, B. E. S. Akgul, and L. N. Chakrapani, "A probabilistic cmos switch and its realization by exploiting noise," in *Proc. of In IFIP-VLSI SoC*, Perth, Western Australia, Oct. 2005.

[4] L. Chakrapani, B. E. S. Akgul, S. Cheemalavagu, P. Korkmaz, K. Palem, and B. Seshasayee, "Ultra-efficient (embedded) SOC architectures based on probabilistic cmos (PCMOS) technology," *Proc. of Design Automation and Test in Europe (DATE)*, Mar. 2006.

[5] N. Sano, "Increasing importance of electronic thermal noise in sub-0.1mm Si-MOSFETs," *The IEICE Transactions on Electronics*, vol. E83-C, pp. 1203–1211, Aug. 2000.

[6] S. Lalgudi, J. Mao, and M. Swaminathan, "Parasitic extraction and simulation of simultaneous switching noise in on-chip power distribution networks," in *12th Electromagnetic Compatibility Conference 2005*, vol. 12, no. 5, May 2005, pp. 497 – 510.

[7] J. D. Meindl, "Low power microelectronics: Retrospect and prospect," *Proceedings of the IEEE*, vol. 83, no. 4, pp. 619 – 635, Apr. 1995.

[8] K. L. Shepard, "Conquering noise in deep-submicron digital ics," *IEEE Design and Test of Computers*, vol. 15, no. 1, pp. 51 – 62, Jan. - Mar. 1998.

[9] "ITRS 2005 edition. http://www.itrs.net/common/2005itrs/home2005.htm."

[10] R. Hedge and N. R. Shanbhag, "A voltage overscaled

low-power digital filter IC," *Digital Object Identifier*, pp. 388–391, 2004.

[11] A. Forestier and M. Stan, "Limits to voltage scaling from the low power perspective," in *Proceedings on Integrated Circuits and Systems Design*, Sept. 2000, pp. 365–370.

[12] A. Andrei, M. T. Schmitz, P. Eles, Z. Peng, and B. M. A. Hashimi, "Quasi-static voltage scaling for energy minimization with time constraints," in *Design, Automation and Test in Europe*, 2005, pp. 514–519.

[13] K. V. Palem, "Energy aware computing through probabilistic switching: A study of limits," *IEEE Trans. Computer*, vol. 54, no. 9, pp. 1123–1137, 2005.

[14] ——, "Proof as experiment: Probabilistic algorithms from a thermodynamic perspective," in *Proc. Intl. Symposium on Verification (Theory and Practice)*, Taormina, Sicily, June 2003.

[15] P. Korkmaz, B. E. S. Akgul, K. V. Palem, and L. N. Chakrapani, "Advocating noise as an agent for ultra-low energy computing: Probabilistic CMOS devices and their characteristics," *Japanese Journal of Applied Physics, SSDM Special Issue Part 1*, pp. 3307–3316, Apr. 2006.

[16] S. Cheemalavagu, P. Korkmaz, and K. V. Palem, "Ultra low-energy computing via probabilistic algorithms and devices: CMOS device primitives and the energy-probability relationship," in *Proc. of the International Conference on Solid State Devices and Materials*, Tokyo, Japan, Sept. 2004, pp. 402 – 403.

[17] K. V. Palem and B. E. S. Akgul, "The explicit use of probability in CMOS designs and the ITRS roadmap: From ultra-low energy computing to a probabilistic era of Moore's law for CMOS," SRC, Cavins Corner, Tech. Rep., Sept. 2005.

[18] P. Korkmaz and K. V. Palem, "The inverse error function and its asymptotic "order" of growth using O and ω," Georgia Institute of Technology, Tech. Rep. CREST-TR-06-02-01, Feb. 2006.

[19] S. Cheemalavagu, P. Korkmaz, K. V. Palem, B. E. S. Akgul, and L. N. Chakrapani, "A probabilistic CMOS switch and its realization by exploiting noise," *Proc. of IFIP International Conference on VLSI SoC*, Oct. 2005.

[20] K. V. Palem, B. E. Akgul, and J. George, "Variable scaling for computing elements," Invention Disclosure, Atlanta, GA, Feb. 2006.

[21] K.-U. Stein, "Noise-induced error rate as a limiting factor for energy per operation in digital ICs," *IEEE J. Solid-State Circuits*, vol. 12, pp. 527–530, Oct. 1977.

[22] M. Lu, *Arithmetic and Logic in Computer Systems*. Hoboken, NJ: John Wiley & Sons, Inc., 2004.

[23] C. Oliver and S. Quegan, *Understanding Synthetic Aperture Radar Images*. Raleigh, NC: SciTech Publishing, 2004.

[24] L. B. Kish, "End of Moore's law: Thermal (noise) death of integration in micro and nano electronics," *Physics Letters A*, vol. 305, pp. 144–149, Dec. 2002.

[25] M. Elgamel and M. Bayoumi, "Interconnect noise analysis and optimization in deep submicron technology," *IEEE Circuits and Systems Magazine*, vol. 3, no. 4, pp. 6 – 17, 2003.

[26] S. Cheemalavagu, P. Korkmaz, and K. V. Palem, "Ultra low-energy computing via probabilistic algorithms and devices: CMOS device primitives and the energy-probability relationship," in *Proc. of The 2004 International Conference on Solid State Devices and Materials*, Tokyo, Japan, Sept. 2004, pp. 402 – 403.

[27] K. Nepal, R. Bahar, J. Mundy, W. Patterson, and A. Zaslavsky, "Designing logic circuits for probabilistic computation in the presence of noise," in *Proc. of Design Automation Conference*, June 2005, pp. 486 – 490.

[28] M. Allam and M. Elmasry, "Low power implementation of fast addition algorithms," in *Proc. of IEEE Canadian Conference on Electrical and Computer Engineering, 1998*, vol. 2, May 1998, pp. 645 – 647.