Highly Energy and Performance Efficient Embedded Computing through Approximately Correct Arithmetic

A Mathematical Foundation and Preliminary Experimental Validation

Lakshmi N. B. Chakrapani
Department of Computer science
Rice University
Houston, Texas, USA
chakra@rice.edu

Kirthi Krishna Muntimadugu
Department of Electrical and Computer Engineering
Rice University
Houston, Texas, USA
kirthi.krishna@rice.edu

Avinash Lingamneni
Department of Electrical and Computer Engineering
Rice University
Houston, Texas, USA
Avinash.Lingamneni@rice.edu

Jason George
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, Georgia, USA
george@ece.gatech.edu

Krishna V. Palem*
Department of Computer science
Department of Electrical and Computer Engineering
Rice University
Houston, Texas, USA
palem@cs.rice.edu

ABSTRACT

We develop a theoretical foundation to characterize a novel methodology for low energy and high performance DSP for embedded computing. Computing elements are operated at a frequency higher than that permitted by a conventionally correct circuit design, enabling a trade-off between *error that is deliberately introduced, and the energy consumed*. Similar techniques considered previously were relevant to deeply scaled future technology generations. Our work extends this idea to be applicable to current-day designs through: *(i)* a mathematically rigorous foundation characterizing a trade-off between energy consumed and the quality of solution, and *(ii)* a means of achieving this trade off through very aggressive voltage scaling beyond that of a conventionally designed circuit. Through our "CMOS inspired" mathematical model, we show that our approach is better (by an exponential factor) than the conventional uniform voltage scaling approach for *comparable computational speed or performance*. We further establish through experimental study that a similar improvement by a factor of 3.4x to the SNR over conventional voltage-scaled approaches can be achieved in the context of the ubiquitous discrete Fourier transform.

*This author wishes to thank the the Moore distinguished faculty fellow program at the California Institute of Technology, which enabled pursuing this work in part.

CASES'08, October 19–24, 2008, Atlanta, Georgia, USA.

General Terms

Design,Experimentation,Performance

Categories and Subject Descriptors

B.2.0 [**Hardware**]: Arithmetic and Logic Structures—*general*

Keywords

Digital signal pocessing, voltage overscaling

1. INTRODUCTION

High performance and low energy operation are of great importance in embedded and mobile systems. A huge class of such power and performance constrained systems realize various forms of digital signal processing or "DSP" workloads. Hence techniques for energy and performance efficiency at various levels—circuit, architecture, algorithmic and application—have been sought and invented in this context. These techniques may be classified under two categories: (i) techniques which improve energy and performance with no degradation of quality of solution. For example, these techniques include using better algorithms to replace complex operations such as multiplications with simpler operations such as additions and eliminating redundant computations. At the circuit level, techniques for energy and performance efficiency typically seek more efficient circuit implementations of DSP primitives. (ii) Techniques which trade off energy for quality of solution. These are typically at the algorithmic level, where parameters such as the number of quantization levels, and the precision of coefficients are traded off for the quality of solution [8, 13, 1, 2]. These conventional techniques for low energy and high performance DSP utilize *deterministic* building blocks and

primitives—the constituent arithmetic operations are correct all the time—with energy and performance efficiency obtained through efficient implementation, or reduction of the number and size of these primitives.

As a radical departure from these techniques, George et. al [6], demonstrated how the *correctness* of arithmetic primitives may be traded off for energy consumed, while providing an acceptable or "good enough" solution. The principle that enables such an opportunity, is the relationship between energy and the probability of correctness in highly scaled, noise susceptible (future) CMOS technologies. Though impressive energy efficiencies were demonstrated empirically, and though such techniques are likely to be of great value in future noise susceptible CMOS and novel devices (such as molecular devices [15]), such techniques are not applicable to current day technology generations based on CMOS, where the noise levels are not yet sufficient to enable such a trade off.

With this as background and as a significant advance, we present two key contributions (i) A rigorous theoretical foundation to quantify such an energy-probability of correctness trade off at the arithmetic level, and (ii) A practical technique to realize this trade off in current day technology generations that is inspired by the foundational model. The probabilistic behavior that we consider is induced by aggressive voltage scaling, and hence the errors are induced due to *propagation delays*. A higher investment in energy implies operating the corresponding circuit elements at a higher voltage, and would translate into faster propagation of signal values and hence lower probability of error.

As an example consider the 8 bit binary addition of two numbers A and B where $A = 01001010$ and $B = 01000110$, where the least significant bit (the "first" bit) is written on the right and the most significant bit (the "eighth bit") on the left. We will use the "ripple carry technique" to perform such an addition. As illustrated in Figure 1(i), we notice that the addition of the second bit *generates* a carry, which is *propagated* to the third bit, which, in turn generates a carry, which when added to the fourth bit, generates a carry and sets the results of the addition of the fifth bit to 1. We shall call this a *carry chain* of length 3 originating at *position* 2. In this example, there is also a carry chain of length 1, originating at position 7. If the delay for one addition combined with the wire delay (the delay for computing and propagating this carry to the next significant bit position) is d, in conventional circuit implementations of this adder, the total delay is taken as $D = 8d$ and the circuit operated at a frequency $1/D$. This is because, in the worst case (adding 10101011 to 01010101 for example) the carry chain is of length 8 and originates at position 1. The total delay of $8d$ determines the operating frequency $f = 1/8d$ of the circuit, which in turn, determines the operating voltage V, since in CMOS circuits, $V \propto 1/d$. This in turn, determines the energy consumed, where $E \propto V^2$.

Now, let us consider the addition operation from the previous paragraph when the entire adder circuit is operated at $V' = \frac{1}{2}V$ and hence $d' = 2d$ and let the frequency of operation be f. Since the individual carry computation and propagation delay is doubled, in the time $1/f$, any carry can propagate to only half as many (or $n/2$) bit positions. Now, since the length of the larger carry chain is $3 < 8/2$, the addition would be performed correctly with this voltage lowering. Moreover, in this case, since energy consumed by CMOS circuits is quadratically related to the operating voltage, the operating energy $E' = E/4$. Hence in this case, operating voltage can be halved, and the energy consumed can be improved by a factor of 4, without compromising the correctness of operation. However, considering Figure 1 (ii) where the length of carry chain is 5, if the adder is operated at voltage V', the addition would be performed incorrectly, with 01000000 as the computed result instead of 10000000, an error magnitude of 64. However this adder achieves a factor of 4 in energy efficiency when compared to the case where the adder is operated at voltage V wherein the answer is computed correctly. This illustrates our first principle: *There is a trade off between energy consumption and error induced by propagation delay, in circuits which implement arithmetic operations, that can be exploited to garner energy savings*

Now, let us consider the case when the inputs are 00010101 and 00001011 (Figure 1(iii)). Even though the length of the carry chain is 5—the same length as the case described in Figure 1(ii)—since the carry chain originates in a less significant position and though the adder is operated at voltage V', the error is significantly lower in this case—a magnitude of 16—for the same factor of 4 in energy savings when compared to the correct operation with voltage V. This case illustrates a second principle: *Errors in bits of a higher value affect the quality of solution more than similar number of errors in bits of a lower value.*

Combining these two principles, we shall outline our technique introduced in this paper for trading energy for quality of solution. Specifically, we will operate the full adders in the more significant position with a higher supply voltage, when compared to the full adders in the less significant positions. Thus we decrease the error rate or "probability" of error in the more significant positions when compared to the less significant positions. We shall refer to our approach as the "non uniformly voltage scaled addition". A conventional ("uniform") aggressive voltage scaling on the other hand, would operate all full adders with the same (reduced) supply voltage, and hence the error rate would be the same irrespective of the bit position.

In the work cited before [6], the methodology is ad-hoc, a mathematical model which trades off the energy investment in each full adder to the expected *magnitude* of error is vital for a systematic exploration to guide the design of arithmetic units, and to provide an intellectual framework with sound foundation and to avoid this ad-hoc design. To this end, we introduce a theoretical characterization of an adder which combines the energy consumption with the notion of the *value* of a bit—in the binary representation of numbers, more significant bits are of a higher value than less significant bits for example—and characterize the ability to trade off energy for *quality of solution* of arithmetic operations. Here, The quality of solution is quantified through the expected magnitude of error at the arithmetic level, and is determined by the error of the constituent bits as well as their values, the latter determined by their position significance.

Our theoretical model has four independent parameters, of which, the first three are n the width of the adder, $f = 1/\mathbb{D}$ the frequency of operation of the adder, $v_i = 1/d_i$, the supply voltage for each individual full adders. For the purpose of illustration, all of our mathematical development in this paper is in the context of a ripple carry design. While the theoretical model is robust enough to characterize cases

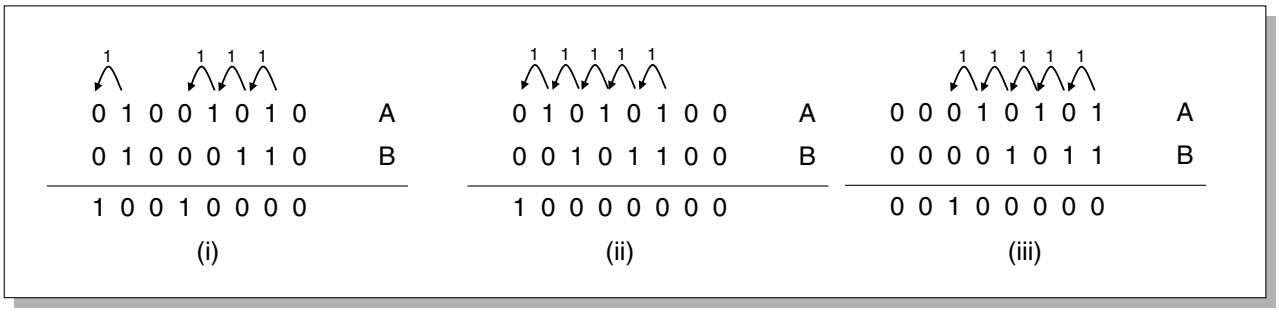

Figure 1: Carry chains and positions for addition of different Binary numbers

where the n full adders in a n bit adder, are all operated different supply voltages, practical considerations such as routing and the limitation of the number of power planes will limit the number of realizable levels of supply voltage in any CMOS based implementation. Hence, we augment our theoretical foundation with an additional independent parameter m, the number of "bins" or distinct levels of supply voltages. For example, in the uniformly voltage scaled case, there is $m = 1$ bin, since all full adders are operated at the same supply voltage. In future implementations, typically, $m \leq 4$ distinct levels of supply voltages are anticipated; a restriction that we will observe throughout the rest of this paper.

Based on this model, we derive our main result: For a n bit addition with an operating frequency f and with the same energy consumption *the ratio of the expected magnitude of error between the uniform voltage scaling scheme and the non uniform voltage scaling scheme is* $\Omega\left(2^{n/(c)}\right)$ *for a constant number of bins* m: as the number of bits $n \to \infty$ the gap (ratio) between *expected* error introduced by biased voltage scaling and conventional voltage scaling for an *equal amount of energy consumed* grows exponentially as $\Omega(2^{n/c})$. The expectation is determined by averaging over the inputs to the adder drawn uniformly from the set of all possible inputs.

Moving away from the theoretical characterization of an adder, we demonstrate that the central principle explored in this work—the trade off between error introduced by propagation delays and energy consumption, taking into consideration the value associated with the corresponding circuit elements—can be implemented using current day CMOS technologies. We evaluate a ripple carry adder, with the delays and energy consumption modeled through HSPICE, in three different configurations (a) The conventional case, where the adder is operated at the nominal voltage V and frequency f. (b) The uniformly voltage overscaled case, where the adder is operated at frequency f but at voltages lower than V and (c) The biased voltage case, where some full adders are operated at higher voltages and some full adders at lower voltages. In the third case, considering implementation constraints, we use a total of four voltage levels, that is $m = 4$. We consider the discrete Fourier transform DFT of audio and image data and build an implementation of DFT using the biased adders and show a factor of 3.4x SNR advantage in the context of DFT for comparable energy. We envision our circuits to be used in the context of *application specific integrated circuits* (ASICs) and hence the arithmetic structures are used to compute DSP operations and not as memory access units.

1.1 Related Work

Energy efficient DSP implementations is a large area of study. We shall examine prior work in two sub-areas, which are most relevant (i) Using voltage scaling and multiple voltage levels for energy efficiency (ii) Using circuit level timing speculation and voltage overscaling for energy efficiency. The former set of techniques which are *adaptive* adjust the throughput of the circuit based on the workload. Typically throughput is adjusted by operating the circuit at varying frequencies (and hence varying voltage and energy states) hence gaining energy efficiency when compared to a circuit which is operated at the peak frequency all the time. Examples of such work include [10, 11]. Non-adaptive techniques typically operate the circuit at multiple voltages—critical paths of the circuit being implemented at higher voltages when compared to the non-critical paths—or rely on circuit implementation techniques like transistor sizing for energy efficiency [3, 9, 17, 18]. A distinguishing feature of the these techniques is that the correctness of the circuit and the operations implemented by these circuits are not traded off for energy efficiency. Some techniques which use circuit-level timing speculation, do allow incorrect operation of circuit elements, which are detected and corrected. Examples include techniques where, in a voltage overscaled circuit—a circuit that is operated below the safe voltage for a given frequency of operation—a delay latch is utilized to detect circuit errors for subsequent correction [5]. In other techniques, circuit level timing errors are not corrected at the circuit level, rather techniques borrowed from signal processing is utilized to correct such errors [7, 14, 16]. A distinguishing feature of these set of techniques is that circuit errors are allowed, but detected and corrected at the circuit, architecture and application level. By contrast, in this work the associated critical path delay of a circuit is violated by design for energy and performance efficiency. As a radical departure from the traditional techniques, George et. al. [6] show how circuit level errors can be ignored if (i) the application level impact is low and (ii) if such a trade off leads to disproportionate gains in energy efficiency. However, this technique crucially relied on deeply scaled noise-susceptible CMOS devices and are not applicable to current day devices. On the other hand, our work provides a detailed theoretical model to trade off energy consumption to probability of correctness through voltage overscaling, while taking into account the number of distinct voltage levels available in practical imple-

mentations. We realize that in itself, an observation about the fact that a circuit that violates its critical path delay, can be used in a computation is straight forward if not trivial. However our contribution is being able to quantify and to show the amount of utility of such a circuit (Section 3), and through preliminary empirical observations, further establish this fact (Section 4).

2. ABSTRACTING A MODEL

We first define a model for analyzing the energy-quality of solution trade-off for a ripple carry adder, analogous to the model which related energy to the probability of correctness in the CMOS domain [4]. Consider any variable x, unless mentioned otherwise, a polynomial of degree n (or size n) will be used to refer to any polynomial X which denotes $a_n x^n + a_{n-1}x + \cdots + a_1 x + a_0 x^0$ where the coefficients $a_0, a_1, \cdots, a_n \in \{0,1\}$. The index i of any coefficient (or correspondingly the degree of the corresponding monomial) will be referred to the "position" of the coefficient. The value of this polynomial at $x = 2$, denoted by $X(2)$ is defined to be its *value* or equivalently, the integer it represents. The distance between two polynomials X and Y is defined to be $|X(2) - Y(2)|$. Essentially X is the binary representation of the integer $X(2)$ and the distance is the absolute value of the difference between the integers that the polynomials X and Y represent.

Consider the Boolean function $\mathcal{C} : \{0,1\}^3 \rightarrow \{0,1\}$ where $\mathcal{C}(a,b,c)$ is defined to be $(a \wedge b) \vee (a \wedge c) \vee (b \wedge c)$, where $\vee$ and $\wedge$ represents Boolean disjunction and conjunction respectively, and $a, b, c \in \{0,1\}$. Given two polynomials $X = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0 x^0$ and $Y = b_n x^n + b_{n-1}x^{n-1} + \cdots + b_1 x + b_0 x^0$, consider the operator $\odot$ such that $X \odot Y$ is defined to be a polynomial $Z = c_{n+1}x^{n+1} + c_n x^n + \cdots + c_1 x + c_0 x^0$, where

$$c_i = \begin{cases} 0 & \text{if } i = 0 \\ \mathcal{C}(a_j, b_j, c_j) & \text{for } 1 \leq i \leq n+1, \text{ where } j = i-1 \end{cases}$$

Informally, the $\odot$ operator evaluates the carries generated by the coefficients of the polynomials X and Y under conventional binary addition.

As a slight variation of Pippenger [12], a position i said to *generate* a carry if $a_i = b_i = 1$ and a position i *propagates* a carry if exactly one of a_i, b_i equals 1. A carry chain of length k is said to originate at position i if the i^{th} position generates a carry, and k subsequent positions propagate a carry. A set of k consecutive positions $\{i-1, i-2, \cdots, i-k\}$ will be referred to as a carry block of size k at position i, if the $(i-k)^{th}$ position generates a carry and all of the remaining $(k-1)$ positions propagate a carry.

We note that if the coefficients of X, Y are chosen uniformly at random from the set $\{0,1\}$, the probability that the position i propagates a carry and there is a carry block of length k at position i is $(1/2)(1/2^{k-1})(1/4) = 1/2^{k+2}$.

We now incorporate delay with the computation of a carry at any position i. Let a *delay vector* to be a vector of length n where each element is a non-negative integer. Consider a delay vector D of length $n+1$, where D denotes $\langle d_n, d_{n-1}, \cdots, d_0 \rangle$. Given this delay vector D, and two polynomials X, Y of degree n, the result of "an addition under the delay vector D with a total delay of $\mathbb{D}$", denoted by $X \odot_{\mathbb{D},D} Y$ will be defined to be the polynomial Z' such that $Z' = c'_{n+1}x^{n+1} + c'_n x^n + \cdots + c'_1 x + c'_0$, where $c'_0 = 0$ and for $0 \leq i \leq n$, $c'_{i+1} = \mathcal{C}'(a_i, b_i, \hat{c}_i, \mathbb{D})$. The function $\mathcal{C}'$ is defined to be

$$\mathcal{C}'(a_i, b_i, \hat{c}_i, \hat{d}) = \begin{cases} \mathcal{C}(a_i, b_i, \hat{c}_i) & \text{if } \hat{d} \geq d_i \\ 0 & \text{otherwise} \end{cases}$$

where $\hat{c}_i = \mathcal{C}'(a_j, b_j, \hat{c}_j, \hat{d} - d_i)$ and $j = i - 1$.

We note that given X and Y of degree n, and a delay vector D which denotes $\langle d_n, d_{n-1}, \cdots, d_1, d_0 \rangle$, if $\mathbb{D} \geq \sum_{i=0}^{i=n} d_i$, then $X \odot Y \equiv X \odot_{\mathbb{D},D} Y$. Thus $\odot_{\mathbb{D},D}$ operator models the case where the addition is performed with a total delay of $\mathbb{D}$ (or frequency $1/\mathbb{D}$) and the computation of carry at any position i is associated with a delay d_i and the sum of the delays in the previous positions. In the conventional case of a ripple carry adder of size $n+1$, where each full adder is operated with the same supply voltage v, (and hence operates with the same delay d), the delay vector $\hat{D}$ which corresponds to this degenerate case is $\langle d, d, d, \cdots, d \rangle$, and hence for conventional correct operation of the adder, $\mathbb{D} \geq (n+1)d$. If $\mathbb{D} \geq (n+1)d$, for polynomials A, B of degree n, if $Z = A \odot B$ and $Z' = A \odot_{\mathbb{D},D} B$, $Z(2) = Z'(2)$. On the other hand if $\mathbb{D} < (n+1)d$, it might be the case that $Z(2) \neq Z'(2)$, and this corresponds to the conventional uniform aggressive voltage scaling technique of performing addition.

Given a delay vector D such that $D = \langle d_n, d_{n-1}, \cdots, d_1, d_0 \rangle$, the cost of performing $\odot_{\mathbb{D},D}$ operation (or equivalently energy cost of $\odot_{\mathbb{D},D}$), denoted as $\mathcal{E}(X \odot_{\mathbb{D},D} Y)$ will be defined as

$$\mathcal{E}(X \odot_{\mathbb{D},D} Y) = \sum_{i=0}^{i=n} \frac{1}{d_i^2}$$

This cost scheme is based on the relationship between switching time, supply voltage and switching energy. Since, for CMOS devices, switching time is inversely proportional to the supply voltage and hence $d \propto \frac{1}{v}$, and switching energy E is proportional to the square of the voltage and hence $E \propto v^2$, we have $E \propto 1/d^2$. For a conventional ripple carry adder with no associated error, if the corresponding delay vector $\hat{D}$ of length $n+1$ is $\langle d, d, d, \cdots, d \rangle$ the energy cost of an addition under $\hat{D}$ with total delay $(n+1)d$ is $(n+1)/d^2$.

We define any polynomial X to be chosen uniformly at random if the coefficients of X are chosen uniformly at random from the set $\{0,1\}$. Given a delay vector D and a total delay $\mathbb{D}$, if X and Y are polynomials of degree n chosen uniformly at random, the expected magnitude of error of the operation $\odot_{\mathbb{D},D}$ is defined to be the expectation, $\text{ExpErr}(\mathbb{D}, D) = \text{Exp}[|Z(2) - Z'(2)|]$ where $Z = X \odot Y$ and $Z' = X \odot_{\mathbb{D},D} Y$.

$$\begin{aligned} \text{ExpErr}(\mathbb{D}, D) =& \text{Exp}[|Z(2) - Z'(2)|] \\ =& \text{Exp}\left[\left|\sum_{i=0}^{n+1}(c_i - c'_i)2^i\right|\right] \qquad (1) \\ \leq& \text{Exp}\left[\sum_{i=0}^{n+1}|(c_i - c'_i)2^i|\right] \end{aligned}$$

hence

$$\text{ExpErr}(\mathbb{D}, D) \leq \sum_{i=1}^{n+1} p_i 2^i \qquad (2)$$

where p_i is the probability that $c_i \neq c_i'$. If ExpErr$(\mathbb{D}, D)$ is the expected magnitude of error of the operation $\odot_{\mathbb{D},D}$ and ExpErr$(\mathbb{D}, \hat{D})$ is the expected magnitude of error of the operation $\odot_{\mathbb{D},\hat{D}}$, then the relative magnitude of error $\mathcal{R}(\mathbb{D}, \hat{D}, D)$ is defined to be

$$\mathcal{R}(\mathbb{D}, \hat{D}, D) = \frac{\text{ExpErr}(\mathbb{D}, \hat{D})}{\text{ExpErr}(\mathbb{D}, D)}$$

We note that practical considerations such as routing and the limitation of the number of power planes, limit the number of levels of supply voltage in any CMOS based implementation. Hence, in practice, any delay vector D of length $n+1$ will be considered to contain m "bins" (where m is some constant and divides $n+1$), where $(n+1)/m$ successive elements are equal. That is, if $o = (n+1)/m$, $d_i = d_j$ if $\lfloor i/o \rfloor = \lfloor j/o \rfloor$. For example, for an addition operation of two polynomials of degree 5, under a delay vector D of length 6 with 3 bins, D would be of the form $\langle d_5, d_4, d_3, d_2, d_1, d_0 \rangle$, where $d_0 = d_1$, $d_2 = d_3$ and $d_4 = d_5$.

We consider one such delay vector D, which we refer to as *the geometric delay vector of length $n+1$ with m bins*, where D denotes $\langle d_n, d_{n-1}, \cdots, d_1, d_0 \rangle$, $o = (n+1)/m$ and $d_i = d(1-2^{-j})^{-1}$ where $j = \lfloor i/o \rfloor + 1$. Two delay vectors $D, \hat{D}$ of equal length are said to be of equal energy if for any polynomial X, Y of degree n, $\mathcal{E}(X \odot_{\mathbb{D},D} Y) = \mathcal{E}(X \odot_{\mathbb{D},\hat{D}} Y)$. We define a uniform delay vector of length $n+1$ to be a vector $\hat{D} = \langle \hat{d}_n, \hat{d}_{n-1}, \cdots, \hat{d}_1, \hat{d}_0 \rangle$ where for $0 \leq i, j \leq n$, $\hat{d}_i = \hat{d}_j$. In the experiments and results discussed in Section 4, three cases are considered: (a) the conventional correct operation of a ripple carry adder of size n. This corresponds to the case of $X \odot_{\mathbb{D},\hat{D}} Y$ where $\hat{D}$ is a uniform delay vector and $\mathbb{D} \geq \sum_{i=0}^{n-1} \hat{d}_i$, (b) uniform aggressive voltage overscaling, which corresponds to the case of $X \odot_{\mathbb{D},\hat{D}} Y$ where $\hat{D}$ is a uniform delay vector and $\mathbb{D} < \sum_{i=0}^{n-1} \hat{d}_i$ and (c) non uniform (geometric) aggressive voltage overscaling, which corresponds to $X \odot_{\mathbb{D},D} Y$ where D is a geometric delay vector with m bins and $\mathbb{D} < \sum_{i=0}^{n-1} d_i$. In these experiments, the comparisons of interest are (i) in the two cases (b) and (c) above, $\mathcal{R}(\mathbb{D}, \hat{D}, D)$ when $\mathcal{E}(X \odot_{\mathbb{D},\hat{D}} Y) = \mathcal{E}(X \odot_{\mathbb{D},D} Y)$ and (ii) the energy consumption of the two cases $\mathcal{E}(X \odot_{\mathbb{D},\hat{D}} Y)$, $\mathcal{E}(X \odot_{\mathbb{D},D} Y)$ when $\mathcal{R}(\mathbb{D}, \hat{D}, D) = 1$.

We now theoretically quantify the relative magnitude of error $\mathcal{R}(\mathbb{D}, \hat{D}, D)$ when the operators $\odot_{\mathbb{D},D}$ and $\odot_{\mathbb{D},\hat{D}}$ are of equal energy, or equivalently, $\mathcal{E}(X \odot_{\mathbb{D},\hat{D}} Y) = \mathcal{E}(X \odot_{\mathbb{D},D} Y)$.

3. THEORETICAL FOUNDATIONS

To quantify the relative magnitude of error $\mathcal{R}(\mathbb{D}, \hat{D}, D)$, we first derive a lower bound on the expected magnitude of error ExpErr$(\mathbb{D}, \hat{D})$ of addition under a uniform delay vector and an upper bound on the expected magnitude of error ExpErr$(\mathbb{D}, D)$ of addition under a geometric delay vector of length n with m bins when $\mathcal{E}(X \odot_{\mathbb{D},\hat{D}} Y) = \mathcal{E}(X \odot_{\mathbb{D},D} Y)$.

LEMMA 3.1. *If $D = \langle d_{n-1}, d_{n-2}, \cdots, d_1, d_0 \rangle$ is a geometric delay vector of length n with m bins, whereas $\hat{D} = \langle \hat{d}_{n-1}, \hat{d}_{n-2}, \cdots, \hat{d}_1, \hat{d}_0 \rangle$ is a uniform delay vector of length n such that D and $\hat{D}$ are of equal energy, then for $0 \leq i \leq n-1$ and a positive constant m,*

$$\hat{d}_i > d\left(1 - \frac{5}{6m} - \frac{25}{72m^2} + \frac{1}{m2^{m-1}}\right)^{-1}$$

PROOF. The energy cost of addition of two polynomials X, Y of degree $n-1$, under the geometric delay vector D of length $n-1$ with m bins is:

$$\begin{aligned}
\mathcal{E}(X \odot_{\mathbb{D},D} Y) &= \sum_{i=0}^{n-1} \frac{1}{d_i^2} \\
&= \frac{n}{md^2} \sum_{i=1}^{m} \left(1 - \frac{1}{2^i}\right)^2 \\
&= \frac{n}{md^2} \sum_{i=1}^{m} \left(1 - \frac{2}{2^i} + \frac{1}{2^{2i}}\right) \\
&= \frac{n}{md^2} \left(m - \sum_{i=1}^{m} \frac{2}{2^i} + \sum_{i=1}^{m} \frac{1}{2^{2i}}\right)
\end{aligned}$$

Summing the geometric series,

$$\mathcal{E}(X \odot_{\mathbb{D},D} Y) = \frac{n}{md^2} \left(m - 2 + \frac{1}{2^{m-1}} + \frac{1}{3} - \frac{1}{3.2^{2m}}\right)$$

hence

$$\mathcal{E}(X \odot_{\mathbb{D},D} Y) < \frac{n}{d^2} \left(1 - \frac{5}{3m} + \frac{1}{m2^{m-1}}\right) \tag{3}$$

Since in the uniform delay vector, $\hat{d}_i = \hat{d}_j$ for $0 \leq i, j \leq n-1$,

$$\mathcal{E}(X \odot_{\mathbb{D},\hat{D}} Y) = \sum_{i=0}^{n-1} \frac{1}{\hat{d}_i^2} = \frac{n}{\hat{d}_0^2} \tag{4}$$

From (3) and (4) and from the fact that $\mathcal{E}(X \odot_{\mathbb{D},\hat{D}}) = \mathcal{E}(X \odot_{\mathbb{D},D} Y)$,

$$\frac{n}{\hat{d}_0^2} < \frac{n}{d^2} \left(1 - \frac{5}{3m} + \frac{1}{m2^{m-1}}\right)$$

hence

$$\hat{d}_0 > d\left(1 - \frac{5}{3m} + \frac{1}{m2^{m-1}}\right)^{\frac{-1}{2}}$$

Expanding using Taylor series,

$$\hat{d}_0 > d\left(1 - \frac{5}{6m} - \frac{25}{72m^2} + \frac{1}{m2^{m-1}}\right)^{-1}$$

□

Given D, a geometric delay vector of length n with m bins and $\hat{D}$ a uniform delay vector such that $\mathcal{E}(X \odot_{\mathbb{D},\hat{D}} Y) = \mathcal{E}(X \odot_{\mathbb{D},D} Y)$, we now bound the expected magnitude of error ExpErr$(\mathbb{D}, \hat{D})$ from below.

LEMMA 3.2. *If D is a geometric delay vector of length n with m bins whereas $\hat{D}$ is a uniform delay vector such that D and $\hat{D}$ are of equal energy, the expected magnitude of error of $\odot_{\mathbb{D},\hat{D}}$, is at least 2^s where*

$$s = \frac{5n}{6m} + \frac{25n}{72m^2} - \frac{n}{m2^{m-1}} - 2$$

$\mathbb{D} = nd$, and m is a positive constant.

PROOF. Let $\hat{D}$ denote $\langle \hat{d}_{n-1}, \hat{d}_{n-2}, \cdots, \hat{d}_1, \hat{d}_0 \rangle$. Then from Lemma 3.1, for $0 \leq i \leq n-1$, $\hat{d}_i > td$ where

$$t > \left(1 - \frac{5}{6m} - \frac{25}{72m^2} + \frac{1}{m2^{m+1}}\right)^{-1}$$

Let $Z = X \odot Y$ and $Z' = X \odot_{\mathbb{D},\hat{D}} Y$, where $Z = c_n x^n + c_{n-1}x^{n-1} + \cdots + c_1 x + c_0$ and $Z' = c'_n x^n + c'_{n-1}x^{n-1} + \cdots + c'_1 x + c'_0$. If X, Y are polynomials such that a_{n-1}, b_{n-1} propagate a carry and there is a carry block of length n/t, at position $n-1$, then $c'_n \neq c_n$. The probability that for polynomials chosen uniformly at random, a_{n-1}, b_{n-1} propagate a carry and there is a carry block of length n/t, at position $n-1$ is $(1/2)^{(n/t)+2}$. Hence from (1), the expected magnitude of error is at least

$$2^n \frac{1}{2^{\frac{n}{t}+2}} = 2^s$$

where $s = \frac{5n}{6m} + \frac{25n}{72m^2} - \frac{n}{m2^{m-1}} - 2$ □

We now get an upper bound for the expected magnitude of error for addition under a geometric delay vector of length n with m bins and a total delay $\mathbb{D} = nd$.

LEMMA 3.3. *If D is a geometric delay vector of length n with $m > 3$ bins, the expected magnitude of error of $\odot_{\mathbb{D},D}$, is at most*

$$\left(n - \frac{5n}{6m} + 1\right) 2^{\frac{5n}{6m}}$$

where $\mathbb{D} = nd$ and m is a positive constant.

PROOF. Let D denote $\langle d_{n-1}, d_{n-2}, \cdots, d_1, d_0 \rangle$. From the definition of a geometric delay vector of length n with m bins, if $o = (n)/m$, $d_i = d(1 - 2^{-j})^{-1}$ where $j = \lfloor i/o \rfloor + 1$. Let $\hat{\mathbb{D}} = \sum_{i=n-1}^{5n/(6m)} d_i$. Then

$$\begin{aligned}
\hat{\mathbb{D}} &= \sum_{i=n-1}^{\frac{5n}{6m}} d_i \\
&= \sum_{i=n-1}^{\frac{n}{m}} d_i + \sum_{j=\frac{n}{m}-1}^{\frac{5n}{6m}} d_j \\
&= \frac{n}{m} \sum_{i=2}^{m} d \left(1 - \frac{1}{2^i}\right)^{-1} + \frac{n}{6m} 2d \\
&= \frac{nd}{m} \left(\frac{1}{3} + \sum_{i=2}^{m} \left(1 + \frac{1}{2^i - 1}\right)\right) \\
&= \frac{nd}{m} \left(\frac{1}{3} + m - 1 + \sum_{i=2}^{m} \frac{1}{2^i - 1}\right)
\end{aligned}$$

For $m > 3$, $\sum_{i=2}^{m} \frac{1}{2^i - 1} < \sum_{i=2}^{m} 1/(3.2^{i-2})$.
Since $\sum_{i=2}^{m} 1/(3.2^{i-2}) = 1/3(2 - 1/2^{m-2})$,

$$\hat{\mathbb{D}} < \frac{nd}{m} \left(m - 1 + \frac{1}{3} + \frac{2}{3} - \frac{1}{3.2^{m-2}}\right)$$

hence,

$$\hat{\mathbb{D}} < nd$$

Hence the sum of the delays from the $5n/(6m)^{th}$ position to the $n - 1^{th}$ position is less than nd. Hence for addition under the delay vector D with total delay $\mathbb{D}$, for $5n/(6m) \leq i \leq n-1$, if $c'_i \neq c_i$ then $i - 1^{th}$ position propagates a carry and there exists a carry block of length $q \geq i - 5n/(6m) - 1$ at position $i - 1$. For two polynomials X and Y chosen uniformly at random, the probability that $i - 1^{th}$ position propagates a carry and there exists a carry block of length $q \geq i - 5n/(6m) - 1$ at position $i-1$ is at most $(1/2^{i-5n/(6m)})$.

Hence from (2), the expected magnitude of error is at most

$$\sum_{i=\frac{5n}{6m}}^{n} 2^i \left(\frac{1}{2^{i-\frac{5n}{6m}}}\right) = \left(n - \frac{5n}{6m} + 1\right) 2^{\frac{5n}{6m}}$$

□

When D is a geometric bias vector of length n with m bins and when $\mathcal{E}(X \odot_{\mathbb{D},D} Y) = \mathcal{E}(X \odot_{\mathbb{D},\hat{D}} Y)$, we have obtained a lower bound for $\text{ExpErr}(\mathbb{D}, \hat{D})$ and an upper bound for $\text{ExpErr}(\mathbb{D}, D)$. Using this the relative magnitude of error $\mathcal{R}(\mathbb{D}, \hat{D}, D)$ may be computed[1].

THEOREM 1. *If D is a geometric delay vector of length n with a constant number $m > 4$ of bins whereas $\hat{D}$ is a uniform delay vector of length n such that D and $\hat{D}$ are of equal energy, then the relative magnitude of error $\mathcal{R}(\mathbb{D}, \hat{D}, D)$ is $\Omega(2^{n/c})$, where $c > (72m^2 2^{m-1}/(2^{m-1}25 - 72m))$ and $\mathbb{D} = nd$.*

PROOF. Immediate from Lemma 3.2 and Lemma 3.3, and the fact that the relative magnitude of error

$$\mathcal{R}(\mathbb{D}, \hat{D}, D) = \frac{\text{ExpErr}(\mathbb{D}, \hat{D})}{\text{ExpErr}(\mathbb{D}, D)}$$

□

4. EMPIRICAL RESULTS

Based on the theoretical foundations introduced in Section 3, in this section, we present practical implementation techniques, provide empirical results and relate the theoretical foundation to the DSP domain. The erroneous behavior and energy efficiency of the circuits considered in this work is due to the mismatch of the critical path delay and the actual frequency with which the circuit is operated. We first analyze the behavior of a ripple carry adder and later that of a circuit that implements the discrete Fourier transform (DFT) built from ripple carry adders. We characterize the energy and (erroneous) behavior of the ripple carry adder as well as the DFT in three contexts:

- Case (a) correct operation (the baseline), where the operating frequency is less than that determined by the critical path delay of the circuit. In the context of a n bit ripple carry adder, if the delays of the individual full adders is represented by the uniform delay vector $D = \langle d_{n-1}, d_{n-2}, \cdots, d_1, d_0 \rangle$, where $d_i = d$, this corresponds to the case where $\mathbb{D} > nd$.

[1]In the reviewed version of this paper, the relative magnitude of error was defined to be $\text{ExpErr}(\mathbb{D}, \hat{D}) - \text{ExpErr}(\mathbb{D}, D)$. Based on reviewer comments and feedback, rather than a difference measure, the relative magnitude of error has been changed into a ratio measure.

- Case (b) uniform aggressive voltage overscaling where all of the constituent full adders are operated with identical supply voltages, but the ripple carry adder itself is operated with a delay less than that of its critical path delay. That is, $\mathbb{D}$ is less than nd.

- Case (c) non uniform voltage overscaling, where some full adders are operated at a higher voltage than the others, and the entire ripple carry adder is operated with a delay less than that of its critical path delay. Hence, again $\mathbb{D}$ is less than the sum of the delays of the individual full adders. In our experiments, we consider delay vectors with 4 bins.

In all of these cases, the metrics of interest are the energy consumed and the error, quantified in terms of the average magnitude of (absolute) error in the case of a ripple carry adder, and the signal to noise ratio (SNR) in the context of the DFT. The comparisons of interest are

- Case (i) the energy consumption and the average magnitude of error of case (b) and case (c) above, when compared to the baseline.

- Case (ii) for identical energy and operating frequencies of case (b) and case (c), their average magnitude of errors (and SNR). This was defined to be the *relative magnitude* of error in the context of a ripple carry adder in the theoretical analysis.

- Case (iii) for identical operating frequency and error (or SNR) magnitude, the energy consumption of case (b) and case (c).

We call this relationship between error, energy and operating frequency, the energy-error relationship.

4.1 The Energy-Error Relationship of a Ripple Carry Adder

We consider the ripple carry adder to be composed of a chain of full adders of length n, where n is the length of the input operands. In our experiments, a ripple carry adder of length 16 is considered. The behavior of an individual full adder is modeled using HSPICE using TSMC $0.25\mu m$ libraries to derive the delay and energy consumption for supply voltages in the range $0.7v$ to $2.5v$. The behavior of a ripple carry adder is modeled using a C based behavioral simulator which utilizes the data generated by the HSPICE simulations.

To study the energy-error relationship for various operating frequencies and supply voltage configurations, the average error rate of the output and the expected magnitude of error for the three cases—case (a), case (b) and case (c) described above—is modeled using the behavioral simulator for varying operating frequencies. Three ranges of operating frequencies are of interest. If D is the delay vector of a non uniformly voltage scaled adder, and if d_{n-1} is the delay of the fastest full adder and d_0 is the delay of the slowest full adder, the first range of operating frequencies of interest is between d_{n-1} and $2d_{n-1} - \epsilon$, or equivalently, $d_{n-1} \leq \mathbb{D} \leq 2d_{n-1} - \epsilon$. We shall refer to this case as the "first range of operating frequencies". Similarly a second range of interest is the case where $4d_{n-1} \leq \mathbb{D} \leq 8d_{n-1}$ and finally the range of operating frequencies such that $d_0 \leq \mathbb{D} \leq \sum_{i=0}^{n-1} d_i$.

For these three ranges of operating frequencies, the energy and error magnitude of a 16-bit ripple carry adder is

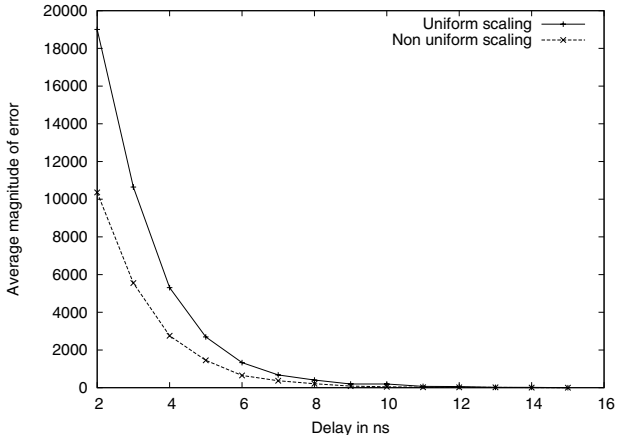

Figure 2: The change in the average magnitude of error with respect to the operating delay $\mathbb{D}$

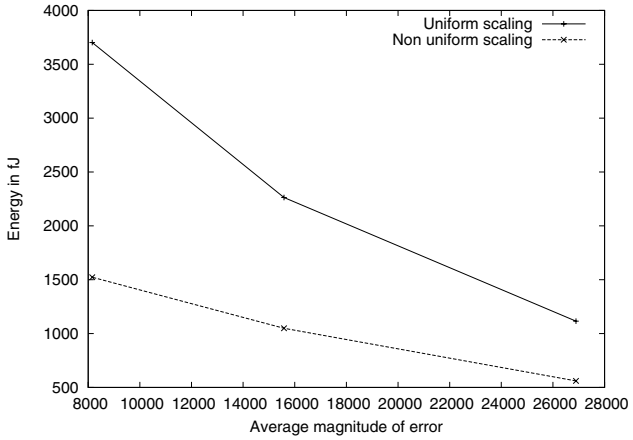

Figure 3: Average magnitude of error with respect to the energy of operation for $\mathbb{D} = 0.8ns$ (in the range $d_{n-1} \leq \mathbb{D} \leq 2d_{n-1} - \epsilon$)

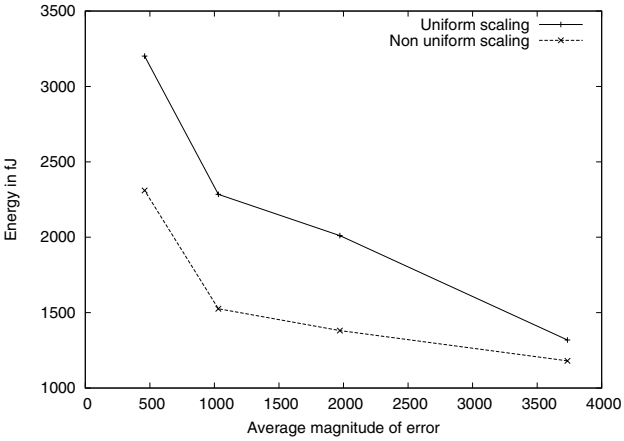

Figure 4: Average magnitude of error with respect to the energy of operation for $\mathbb{D} = 2.3ns$ (in the range $4d_{n-1} \leq \mathbb{D} \leq 8d_{n-1}$)

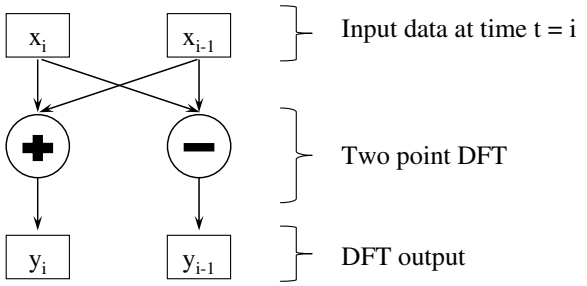

Figure 5: A two point discrete Fourier transform

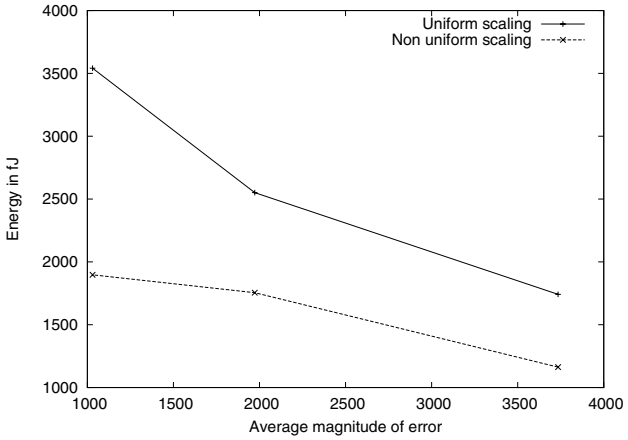

Figure 6: Average magnitude of error with respect to the energy of operation for $\mathbb{D} = 2.0ns$ (in the range $d_0 \leq \mathbb{D} \leq \sum_{i=0}^{n-1} d_i$)

derived for various voltage scaling techniques. To compute the average magnitude of error, this adder is operated on 10000 pairs of numbers derived from a uniform distribution.

4.1.1 Results and Discussion

In all the three ranges, we observe certain general trends. For a specific voltage configuration, as illustrated in Figure 2, the average magnitude of error decreases with decreasing operating frequency. For a specific voltage configuration, the average energy consumption increases with operating delay as higher operating delays allow for more transitions in the constituent full adders as carries propagate across positions. Similarly, as illustrated in Figure 3, Figure 4 and Figure 6, for a fixed operating frequency, the magnitude of error decreases with increasing energy investment, since increasing the energy investment decreases the critical path delay. For a fixed operating frequency and energy consumption, the non uniformly voltage scaled adder, achieves a lower magnitude of error than the uniformly voltage scaled adder, since errors are confined to the lower order bits and hence are of a lesser magnitude. A histogram which compares the magnitude of errors and their relative frequency of occurrence is presented in Figure 7(a) and (b), which shows that in the non uniformly voltage scaled case, errors of a lesser magnitude are quite frequent, whereas error of a higher magnitude are quite rare.

Considering the first range of operating frequencies where $d_{n-1} \leq \mathbb{D} \leq 2d_{n-1} - \epsilon$ and referring back to Figure 3, the non uniformly overscaled case (case (c)) achieves up to a 2.43x reduction in energy consumption when compared to uniformly scaled configuration (case (b)) for similar error magnitude of 8156.63 and operating frequency of $0.8ns$. When the operating frequency is in the first range, where $d_{n-1} \leq \mathbb{D} \leq 2d_{n-1} - \epsilon$, the ripple carry adder yields the best energy savings over the baseline, since fastest full adders switch only once and the rest may not have completed even one computation. In this context, since the operating frequency is much higher than that determined by the critical path delay—the ratio of the critical path delay to the operating delay $\mathbb{D}$ is 23.5 in the case considered in Figure 3—the average (absolute) magnitude of error when compared to the baseline can be as high as about 26894.

For operating frequencies in the second range, when the delay of operation of the circuit $4d_{n-1} \leq \mathbb{D} \leq 8d_{n-1}$—and is greater than the sum of the delays of the full adders in the fastest bin in the non-uniformly overscaled case—the non-uniformly voltage overscaled case has an energy savings of about 1.49x over the uniformly overscaled configuration for similar error magnitude of 1030. Since in this case the frequency of operation is lesser than the case considered above—in the case considered in Figure 4, the ratio of the critical path delay to the operating delay is about 8.5—we note that the lowest average magnitude of (absolute) error that can be achieved has decreased as shown in Figure 4.

Finally, for the third range of operating frequencies $d_0 \leq \mathbb{D} \leq \sum_{i=0}^{n-1} d_i$, considering the faster operation where $\mathbb{D} = 2ns$, case (c) achieves a 1.86x reduction in energy consumption over case (b). Since the operating frequency is higher than that of the representative example considered above, for a similar error magnitude of about 1030, the energy consumption of case (b) as well as case (c) is higher.

4.2 Applying the Lessons Learned From the Adder to Audio and Image Data in DFT

To demonstrate the utility of our voltage overscaling technique, we build a circuit which implements the discrete Fourier transform. Discrete Fourier transform is a mathematical technique that transforms a signal in time domain to the frequency domain. The input is a sequence of n complex numbers $x_0, x_1, \cdots, x_{n-1}$ and is transformed into the sequence of n complex numbers $y_0, y_1, \cdots, y_{n-1}$, where for $0 \leq k \leq n-1$,

$$y_k = \sum_{l=0}^{n-1} x_i e^{\frac{-2\pi jkl}{n}} \tag{5}$$

We have considered a 2-point DFT, where each DFT operation is performed on two subsequent values in time. Hence the DFT formula reduces to

$$y_0 = x_0 + x_1 \quad \text{and} \quad y_1 = x_0 - x_1 \tag{6}$$

and is illustrated in Figure 5

To study the energy and signal to noise ratio (SNR) of DFT operations, the behavioral simulator is extended to simulate a DFT circuit and this circuit is simulated using representative data derived from audio files and images. Each simulation of a DFT described in this section consists of about

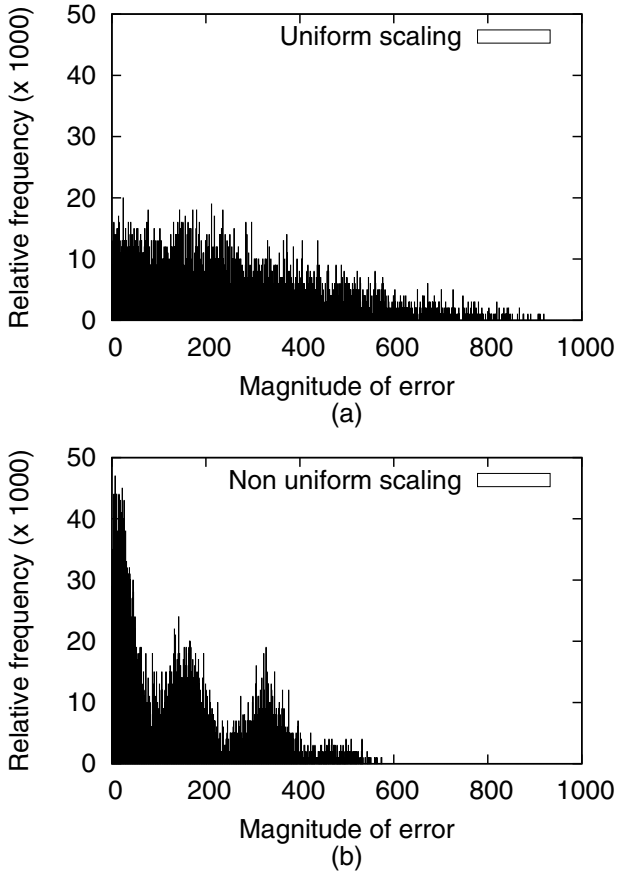

Figure 7: Error magnitude and the relative frequency for (a) the uniformly voltage scaled case and (b) the non uniformly voltage scaled case

1000000 operations of the individual full adders, and the frequency of operation of the circuit is confined to the third range of operating frequencies. In the context of case (b) and case (c) above, after the DFT is computed the inverse DFT of the output is computed using a conventional correct technique. This is compared with the original input to compute the signal to noise ratio SNR. Since $2's$ complement addition is present in the DFT data and since the operating frequency we consider is less, for increasing the efficiency of simulation, the simulation is performed in the granularity of a full adder and the internal state of a full adder as characterized by the signal values of the individual gates is not maintained.

4.2.1 Results and Discussions

Based on the insights obtained from the experiments with the ripple carry adder, which indicate the conditions under which the greatest energy savings of case (c) over case (b) can be achieved for a low average magnitude of error of case (c) over the baseline, we perform the DFT experiment to quantify the energy savings as well as the improvement of SNR of the non-uniformly voltage overscaled case over a uniformly overscaled case.

For a operating delay of $12ns$ for the entire circuit which implements the DFT operation, in the context of the DFT, the non uniformly voltage overscaled case achieves a SNR of $19.63dB$ whereas for identical operating frequency and energy consumption, the signal to noise ratio in the context of uniform aggressive voltage overscaling is $5.79dB$. Thus non-uniform voltage overscaling yields an improvement of 3.4x in the SNR when compared to conventional uniform voltage scaling. If the energy investment in the non-uniform voltage scaling context is lowered such that its SNR matches that of case (b), the non uniformly overscaled technique achieves a 27.2% savings in energy consumption when compared to the uniformly scaled case.

Figure 8: (a) Conventional correct operation (baseline) (b) Non uniformly voltage overscaled case, with saving of 21.7% in energy when compared to the baseline (c) Conventional uniformly voltage overscaled case for the same energy as (b)

In Figure 8, we have presented a visual representation of an image processed with DFT and inverse DFT operations. Figure 8(a) represents the conventional correct operation and serves as the baseline, Figure 8(b) represents the non uniformly voltage overscaled case, which achieves a saving of 21.7% in energy when compared to the baseline, with the same operating frequency and a slight degradation in the quality of the image. Finally for a 21.7% savings in energy and the same operating frequency, Figure 8(c) represents the conventional uniformly voltage overscaled case.

5. CONCLUSIONS AND FUTURE DIRECTIONS

We have demonstrated the value of probabilistic design which combines two key techniques (i) voltage overscaling and circuit level timing speculation, and (ii) multiple supply voltages, for energy efficient computing. Our technique combines good *expected case* behavior with the value of bits (and hence the circuit elements computing them), to arrive at solutions with "somewhat" erroneous arithmetic for energy efficient computation without any degradation in performance. The salient features of this work are a thorough theoretical characterization with energy and error guarantees, and a practical implementation methodology in current day technology generations. This work can be extended along three key directions (i) A theoretical analysis for *optimal* assignment of supply voltage and a technique to perform such optimization given a circuit, the list of available supply voltage levels, a characterization of the workload and a notion of the value computed by the underlying circuit, would be of great utility and importance. (ii) An extension of this technique to other adder architectures and arithmetic primitives and based on this, (iii) An extension of this technique to study other DSP primitives such as the *discrete cosine transform* and other multimedia compression and decompression techniques would be of value.

6. REFERENCES

[1] V. Ananthashayana. Comments on 'Fourier analysis and signal processing by use of the mobius inversion formula' by I.S. Reed et. al. *IEEE Transactions on Signal Processing*, 40(3):676, Mar 1992.

[2] G. Boudreaux-Bartels and T. Parks. Discrete Fourier transform using summation by parts. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 12, pages 1827–1830, apr 1987.

[3] J. Chang and M. Pedram. Energy minimization using multiple supply voltages. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 5(4):436–443, Dec. 1997.

[4] 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 *Proceedings of the IFIP International Conference on Very Large Scale Integration (VLSI-SoC)*, pages 452–457, 2005.

[5] D. Ernst, N. S. Kim, S. Das, S. Pant, T. Pham, R. Rao, C. Ziesler, D. Blaauw, T. Austin, and T. Mudge. Razor: A low-power pipeline based on circuit-level timing speculation. In *Proceedings of the 36th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pages 7–18, October 2003.

[6] J. George, B. Marr, B. E. S. Akgul, and K. Palem. Probabilistic arithmetic and energy efficient embedded signal processing. In *Proceedings of the The IEEE/ACM International Conference on Compilers, Architecture, and Synthesis for Embedded Systems*, pages 158–168, 2006.

[7] R. Hegde and N. R. Shanbhag. Soft digital signal processing. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 9(6):813–823, Dec. 2001.

[8] M. Lamoureux. The Poorman's transform: approximating the Fourier transform without multiplication. *IEEE Transactions on Signal Processing*, 41(3):1413–1415, Mar 1993.

[9] A. Manzak and C. Chaktrabarti. Variable voltage task scheduling algorithms for minimizing energy/power. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 11(2):270–276, Apr. 2003.

[10] S. M. Martin, K. Flautner, T. Mudge, and D. Blaauw. Combined dynamic voltage scaling and adaptive body biasing for lower power microprocessors under dynamic workloads. In *Proceedings of the International Conference on Computer Aided Design*, November 2002.

[11] T. Pering, T. Burd, and R. Brodersen. The simulation and evaluation of dynamic voltage scaling algorithms. In *Proceedings of the International Symposium on Low Power Electronics and Design*, pages 76–81, 1998.

[12] N. Pippenger. Analysis of carry propagation in addition: An elementary approach. Technical report, University of British Columbia, Vancouver, BC, Canada, Canada, 2001.

[13] I. Reed, D. Tufts, X. Yu, T. Truong, M.-T. Shih, and X. Yin. Fourier analysis and signal processing by use of the mobius inversion formula. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 38(3):458–470, Mar 1990.

[14] B. Shim, S. R. Sridhara, and N. R. Shanbhag. Reliable low-power digital signal processing via reduced precision redundancy. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 12(5):497–510, May 2004.

[15] J. M. Tour and D. K. James. Molecular electronic computing architectures: A review. In I. Goddard, W. A., D. W. Brenner, S. E. Lyshevski, and G. J. Iafrate, editors, *Handbook of Nanoscience, Engineering and Technology, Second Edition*, pages 5.1–5.28. CRC Press, New York, 2007.

[16] L. Wang and N. R. Shanbhag. Low-power filtering via adaptive error-cancellation. *IEEE Transactions on Signal Processing*, 51:575–583, Feb. 2003.

[17] Y. Yeh and S. Kuo. An optimization-based low-power voltage scaling technique using multiple supply voltages. In *Proceedings of the IEEE International Symposium on Circuits and Systems*, volume 5, pages 535–538, May 2001.

[18] Y. Yeh, S. Kuo, and J. Jou. Converter-free multiple-voltage scaling techniques for low-power CMOS digital design. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 20(1):172–176, Jan. 2001.