Timing Error Statistics for Energy-Efficient Robust DSP Systems

Rami A. Abdallah, Yu-Hung Lee, and Naresh R. Shanbhag
University of Illinois at Urbana-Champaign, Urbana, IL-61801
[rabdall3,ylee203,shanbhag]@illinois.edu

***Abstract*—This paper makes a case for developing statistical timing error models of DSP kernels implemented in nanoscale circuit fabrics. Recently, stochastic computation techniques have been proposed [1], [2], [3], where the explicit use of error-statistics in system design has been shown to significantly enhance robustness and energy-efficiency. However, obtaining the error statistics at different process, voltage, and temperature (PVT) corners is hard. This paper: 1) proposes a simple additive error model for timing errors in arithmetic computations due PVT variations, 2) analyzes the relationship between error statistics and parameters, specifically the input statistics, and 3) presents a characterization methodology to obtain the proposed model parameters and thus enabling efficient implementations of emerging stochastic computing techniques. Key results include the following observations: 1) the output error statistics is a weak function of input statistics, and 2) the output error statistics depends upon the *one's probability profile* of the input word. These observations enable a one-time off-line statistical error characterization of DSP kernels similar to delay and power characterization done presently for standard cells and IP cores. The proposed error model is derived for a number of DSP kernels in a commercial $45nm$ CMOS process.**

I. Introduction

Present-day worst-case design methodology leads to high power consumption due to increased variations in process, temperature and voltage (PVT) [4], while a nominal-case design results in a loss of yield. Error-resiliency has emerged as an attractive approach for designing nanoscale systems. Error-resilient designs are implemented at nominal PVT corner to save power, and the resulting timing errors are corrected via logical [5] [6], architectural [7], or algorithmic techniques [8].

The robustness and energy efficiency of error-resilient designs depend upon the error statistics of the underlying hardware, even though error statistics are typically not accounted for in the design. For example, the robustness of N-modular redundancy (NMR), where the outputs of N identical kernels are majority voted upon (see Fig. 1(c)), depends upon the component probability of error and requires the error events across the replicated kernels to be independent [10]. While conventional NMR ignores error statistics, *stochastic computation* [2] advocates an explicit characterization and exploitation of component error statistics, i.e., error probability distribution, as seen at the architectural/algorithmic/system levels. Soft-NMR (see Fig. 1(d)) and *bit-level a-posteriori probability processing* (BLAPP) [3] exploit the likelihood of specific error magnitudes in order to correct the output. In fact, stochastic computation techniques, such as *algorithmic-noise tolerance* (ANT) [8], *stochastic sensor network-on-a-chip* (SSNOC)[9], *soft*-NMR [1], and BLAPP [3] exploit the statistical nature of application-level performance metrics, such as bit error-rate (BER), probability of detection, and signal-to-noise ratio (SNR), and match it to the statistical attributes of the underlying device and circuit fabrics. The benefits of such a design philosophy are the tremendous gains in robustness (14X) and energy-efficiency (40-75%) at high-degree of circuit fabric unreliability.

The authors acknowledge the support of Texas Instruments and the Gigascale Systems Research Center (GSRC), one of six research centers funded under the Focus Center Research Program (FCRP), a Semiconductor Research Corporation (SRC) entity.
978-3-9810801-7-9/DATE11/

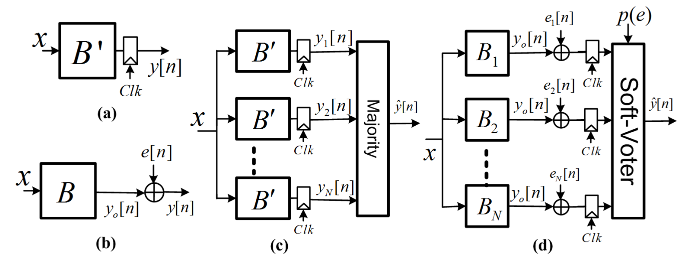


Fig. 1. A DSP kernel (B') exhibiting errors: (a) block diagram, (b) proposed additive error model, (c) NMR setup, and (d) soft-NMR.

Therefore, it is clear that the availability of statistical error models of circuit fabrics, and developing an understanding of the factors that impact these models are essential in the investigation of next generation robust energy-efficient system design techniques. Furthermore, the availability of error statistics enables robustness analysis of existing techniques, as done in [10] for NMR. In this paper, we propose the additive error model (see Fig. 1(b)) at word level for non-recursive DSP computations. The proposed additive error model is effective in abstracting the system-level timing error behavior. Existing techniques [11] predict the probability of error for each output bit while ignoring their correlations, and thus they cannot determine the impact of error on system performance metric or derive the word-level error statistics required by error-aware resilient system techniques such as soft-NMR and BLAPP. Moreover, we show that timing error statistics under the proposed error model are weakly dependent on input statistics. This observation enables a one-time off-line characterization of error statistics for DSP kernels similar to power and delay characterization done today for standard cells and IP cores. Furthermore, we employ various DSP blocks, such as adders and FIR filters, to validate the proposed error model and its characterization.

II. The Proposed Additive Timing Error Model

We focus on non-recursive architectures to simplify the exposition and because such architectures can implement a large class of applications. We propose that the output of any DSP kernel B' with latched input and outputs (see Fig. 1(a)) exhibiting timing errors can be represented via an additive error model (Fig. 1(b)):

$$y[n] = y_o[n] \oplus e[n] = f_o(x[n]) \oplus f_1(x[n], y[n-1], A, V_{dd}, V_t, T, P) \tag{1}$$

where $y[n]$ is the corresponding output at time-index (or clock-cycle) n, $x[n]$ is the input, $y_o[n]$ is the correct (error-free) output, and $e[n]$ is the error. $y_o[n]$ is a function ($f_o()$) only of the present input $x[n]$ since the kernel is non-recursive, while $e[n]$ is a complex non-linear function ($f_1()$) of the kernel architecture (A), the input ($x[n]$), the supply voltage (V_{dd}), the threshold voltage (V_t), the temperature (T), and other physical effects (P). It is also a function of the previous output $y[n-1]$ because some or all bits of the output $y[n]$ can retain their values if the clock period is too small. As $y[n-1]$ is also a function of $x[n-1]$ and $y[n-2]$, we can express $e[n]$ as

$$e[n] = f_2(\mathbf{x}[n], A, V_{dd}, V_t, T, P) \tag{2}$$

where $\mathbf{x}[n] = (x[1], x[2], ..., x[n])$. Function $f_2()$ is complex if described in a deterministic manner. Instead, by recognizing that most emerging applications employ statistical performance metrics such as mean-square error (MSE), SNR, peak-SNR (PSNR), and error-aware resilient techniques rely on the statistics of $e[n]$ rather than the exact value of $e[n]$, we propose to treat $e[n]$ as a random variable E and characterize its probability mass function (PMF) denoted by $P_E(k) = p(e[n] = k)$, i.e., we are interested in $P_E = f_3(P_X, A, V_{dd}, V_t, T, P)$, where P_X represents the PMF of input x. Thus, given a fixed PVT corner, the output error PMF depends on the architectural implementation of the DSP computation and input statistics. The output error statistics is a strong function of the architecture A since different architectures have different path delay distributions, and thus will result in different errors for the same set of input statistics [12]. In the next section, we study the relationship between the error statistics P_E and input statistics P_X.

III. Error Analysis: Impact of Input Statistics

Many DSP applications have a typical input data set or statistics $P_{X,T}$ which can be employed to characterize the output error of a given architecture. However, this makes error characterization procedure dependent on application. Given a DSP kernel/architecture A, we wish to answer the questions:

1) If we employ a typical input PMF $P_{X,T}$ to obtain the output error PMF $P_{E,T}$, can we find a class of input PMFs $C_{X,T} = \{P_{X,i}\}_{i=1}^{M}$ such that they all have similar error PMFs as $P_{X,T}$?
2) Can we find a $P_{X,DSP}$ such that the size of the corresponding class, $|C_{X,DSP}|$, is large and its characteristics are commonly encountered in most DSP applications?

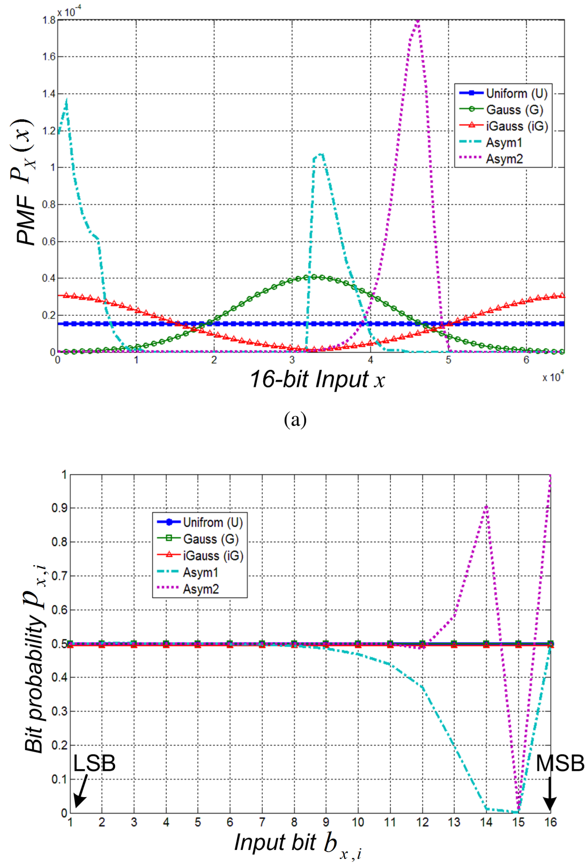
If the answer to the second question is in the affirmative then error characterization can be done once for DSP kernels/architectures employing $P_{X,DSP}$. We show that this is indeed the case. To demonstrate this fact, we study the relationship between input statistics and output error. As Boolean computation occurs at bit-level, it is expected that the output error statistics P_E will be a stronger function of *bit-level input statistics* rather than *word-level input statistics* P_X. Next, we study the relation between word-level and bit-level input statistics.

A. Bit-level vs. Word-level Statistics

Any B_x-bit signal/operand $x[n]$ in a DSP kernel consists of bits denoted by $b_{x,i}[n]$ for $i = 1, 2, \ldots, B_x$. We define the following:

- *Bit probability* of $b_{x,i}$: $p_{x,i} = p(b_{x,i}[n] = 1)$
- *Bit probability profile* (BPP) of an operand x: $\Phi_X = (p_{x,1}, p_{x,2}, \ldots, p_{x,B_x})$, i.e., the set of bit probabilities of its constituent bits.
- *Probability mass function* (PMF) of an operand x: $P_X = p(x)$

It is clear that given a PMF of x P_X, the i^{th} component of x's BPP is computed by summing P_X over x whose i^{th} bit is one. On the other hand, given a BPP Φ_X, a unique P_X cannot be obtained unless the correlations between bits $b_{x,i}$ are explicitly specified. In fact, the next property shows that the number of P_X that can be mapped to the



(a)

(b)

Fig. 2. Various 16-bit input statistics: (a) world-level distribution, and (b) their corresponding bit probability profiles.

same Φ_X is very large. Thus, to simplify and generalized statistical error characterization we can define conditions on Φ_X instead of P_X to enforce similar output error statistics for a given DSP kernel.

Property 1. *For a fixed precision B_x:*
P_x is symmetric around the mean $\mu_x = \frac{2^{B_x}-1}{2} \Leftrightarrow \Phi_x = (0.5, 0.5, ..., 0.5)$, i.e., $p_{x,i} = 0.5$ for all $i = 1, 2 \ldots, B_x$

Property 1 indicates that any PMF of x that is symmetric around $\mu_x = \frac{2^{B_x}-1}{2}$ is mapped to the same BPP where each bit is equally likely to be zero or one. Figure 2(a) and (b) show a set of different 16-bit input distributions and their respective BPPs. Symmetric distributions (U, G, and iG) with mean $\mu_x = \frac{2^{16}-1}{2}$ have the same equally-likely BPPs where each $p_{x,i} = 0.5$ unlike asymmetric distributions (Asym1 and Asym2).

B. Impact of bit-level input statistics on output error

Here, we show that the output error statistics of a given DSP kernel is more dependent on the input BPP, Φ_X, instead of the word-level input PMF, P_X. Thus, condition(s) to ensure similarity of output error statistics can be placed on Φ_X instead of P_X. Any output signal y_i of a DSP kernel/architecture with input x can be viewed as a cascade of L_i *processing elements* (PEs), denoted by $\{PE_k\}_{k=1}^{L_i}$ (see Fig. 3). Each PE_k has an output signal(s) z_k, *intermediate input signal(s)* z_{k-1}, and a *direct input signal set* $x_k \subseteq x$. Note that this representation can take place at different granularity levels. For example, each PE_k can represent a single or multiple PEs or even

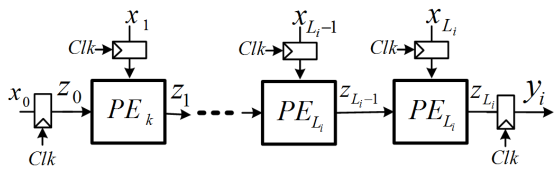


Fig. 3. An architectural model of a DSP kernel with input x, output bit $b_{y,i}$, and L_i processing elements (PE)s.

a single logic gate. In what follows, we decompose the main DSP kernel into PE_k's in such a way that z_{k-1} and x_k are independent. For example, if both z_{k-1} and x_k are generated from the same set of signals then they are correlated and in that case we have to enlarge PE_k to make z_{k-1} an internal signal. With such decomposition, if we know the logic functions implemented by all $PE_{j|j\leq k}$, then the probability of any z_k is completely determined by the BPP (Φ) of $x_{j|j\leq k}$, i.e.,

$$p(z_k) = f_k(\Phi_{x_{j|j\leq k}}) \tag{3}$$

where $f_k(\cdot)$ is a polynomial function that depends on the logic functions of $PE_{j|j\leq k}$.

Timing violations occur when the computation of the output y_i cannot complete in time. Assume that in Fig. 3 at most $L_i - 1$ PEs can compute correctly. A timing error occurs at the output if all L_i PE outputs $z_k[n]$ change their values from the previous clock cycle. If we denote the *transition event* of a signal z_k as t_{z_k}, i.e., $t_{z_k} = 1$ if $z[n] \neq z[n-1]$, then *the probability of output* y_i *being in error*, $pe_{y,i}$, is expressed as:

$$\begin{aligned} pe_{y,i} &= \sum_{\Phi_X} p(t_{z_1} = 1, t_{z_2} = 1, \ldots, t_{z_{L_i}=1}|\Phi_X)p(\Phi_X) \\ &= \sum_{\Phi_X} \prod_{k=1}^{L_i} \left[p\left(t_{z_k} = 1|\{t_{z_j} = 1\}_{j=1}^{k-1}, \Phi_X\right)\right] p(\Phi_X) \end{aligned} \tag{4}$$

However, the input signal set for each PE_k, denoted by $I_{z_k} = \{z_{k-1}, x_k\}$, shields z_k from signal transitions in preceding PEs. Thus,

$$p\left(t_{z_k} = 1|\{t_{z_j=1}\}_{j=1}^{k-1}, \Phi_X\right) = p\left(t_{z_k} = 1|t_{z_{k-1}} = 1, \Phi_{X_k}\right) \tag{5}$$

Substituting in (4), we write:

$$pe_{y,i} = \sum_{\Phi_X} \prod_{k=1}^{L_i} \left[p\left(t_{z_k} = 1|t_{z_{k-1}} = 1, \Phi_{X_k}\right)\right] p(\Phi_X) \tag{6}$$

In addition, t_{z_k} is relatively independent of $t_{z_{k-1}}$ since z_k is determined by x_k as well, i.e., transitions in z_{k-1} do not necessarily imply transitions in z_k. Thus, (7) is expressed as

$$pe_{y,i} = \sum_{\Phi_X} \prod_{k=1}^{L_i} \left[p\left(t_{z_k} = 1|\Phi_{X_k}\right)\right] p(\Phi_X) \tag{7}$$

For ease of notation, we denote $(z_{k-1}[n], z_k[n])$ as $\mathbf{z}[n]$ and introduce the operator $\models$ to denote that all individual components of the two vectors $\mathbf{z}[n]$ and $\mathbf{z}[n-1]$ are not equal. In non-recursive architectures the signal transitions are independent across time and thus the conditional transition probability $p(t_{z_k} = 1|\Phi_{X_k})$ in (7) at the output of each PE_k is expressed as follows:

$$p(t_{z_k} = 1|\Phi_{X_k}) = \sum_{z_k[n-1]\neq z_k[n]} p(z_k[n-1]|\Phi_{X_k})p(z_k[n]|\Phi_{X_k}) \tag{8}$$

This means that we treat the logic state of PE_k independent of time and sum over values where both $z_k[n]$ and $z_k[n-1]$ are different. For example if z_k is 1-bit, then we sum over the tuples $(z_k[n], z_k[n-1]) \in \{(0,1),(1,0)\}$. Since the probabilities are stationary, we treat each $p(\mathbf{z}[n]|\Phi_{X_k})$ and $p(\mathbf{z}[n-1]|\Phi_{X_k})$ similarly. Substituting (3) into (8) and then (7), we obtain:

$$pe_{y,i} = \sum_{\Phi_X} \prod_{k=1}^{L_i} \sum_{z_k[n-1]\neq z_k[n]} f_{k,n}(\Phi_{x_{j|j\leq k}}) f_{k,n-1}(\Phi_{x_{j|j\leq k}}) p(\Phi_X) \tag{9}$$

This shows $pe_{y,i}$ is completely determined by Φ_X. If we assume that at most $D \leq L_i - 1$ PEs compute correctly in one clock-cycle, then, for an error to appear at the output, the last D PEs need to undergo a transition independent of preceding PEs in the chain. Otherwise the error cannot be propagated. Conditioning on $p(z_{Q_i-1})$, where $Q_i = L_i - D - 1$, will shield all $PE_{k>Q_i-1}$ from signal transitions in preceding PEs in the logic chain, and thus (6) is written as:

$$\begin{aligned} pe_{y,i} &= \sum_{\Phi_X} p(\Phi_X) \sum_{z_{Q_i-1}} p(z_{Q_i-1}) \\ &\quad \prod_{k=Q_i}^{L_i} \left[p\left(t_{z_k} = 1|t_{z_{k-1}=1}, \Phi_{X_k}, z_{Q_i-1}\right)\right] \end{aligned} \tag{10}$$

Following similar procedure from (6) to (9), $pe_{y,i}$ in (10) can also be written as a polynomial function of Φ_X. This shows that Φ_X completely determines the probability of output errors. Thus, we can modulate the probability of output error in a DSP kernel/architecture by enforcing conditions on the constituent elements of Φ_X. Next, we employ this observation to generalize the proposed error model to be independent of the application, given a DSP architecture.

C. Generalized Error Characterization Procedure

Given a DSP kernel/architecture and two input statistics $P_{X,1}$ and $P_{X,2}$ that have the same BPP, i.e., $\Phi_{X,1} = \Phi_{X,2}$, then property 1 shows that output error PMFs corresponding to the two input PMFs are equal, i.e., $P_{E,1} = P_{E,2}$. Moreover, Property 1 shows that for a DSP kernel with input precision B_x, all input PMFs that are symmetric around $\frac{2^{B_x}-1}{2}$ have a BPP where all bits are equally likely. We denote this BPP as $\Phi_{X,U}$ and define the corresponding class of PMFs as $C_{X,U}$. The uniform input distribution U can be used as a representative input distribution to characterize the DSP kernel for $C_{X,U}$. Furthermore, $C_{X,U}$ can be enlarged to $C_{X,DSP}$ consisting of any input PMF that is symmetric around any value $\mu_x \in (0 : 2^{B_x} - 1)$. The uniform input distribution U can still be used to obtain $P_{E,DSP}$ of $C_{X,DSP}$. To see this, the mean of $x' = x + \frac{2^{B_x}-1}{2} - \mu_x$ is $\mu_{x'} = \frac{2^{B_x}-1}{2}$ and thus $P_{X'} \in C_{X,U}$. Then, the error PMF of x can be obtained from the error-free DSP kernel functionality f_{DSP} via a simple translation of $P_{E,U}$ as follows: $P_E = P_{E,U} + f_{DSP}\left(\mu_x - \frac{2^{B_x}-1}{2}\right)$. Therefore, output error characterization for a DSP kernel/architecture at a given PVT corner can be done once using a uniform input distribution to obtain $P_{E,DSP}$. The obtained error PMF $P_{E,DSP}$ is applicable to any application whose input statistics is symmetric which is encountered in several DSP applications. If the input statistics in a given application is asymmetric then the error-characterization will need to be redone for the DSP kernel.

Given a DSP kernel/architecture and an error-free operating frequency f_{op}, the generalized error characterization flow is:

1) Generate a uniformly distributed input data set $D_{x,U}$ and obtain the corresponding error-free output $y_o[n]$ using an RTL or fixed-point simulation.
2) Synthesize the design at a PVT corner to obtain a gate-level netlist of the DSP kernel that can operate error-free at f_{op}.

TABLE I

KL DISTANCE BETWEEN ERROR PMFS OF 16-BIT ADDERS UNDER VARIOUS INPUT STATISTICS AND ERROR PMF P_{E_U}.

	16-bit RCA				16-bit CBA				16-bit CSA			
K_{VOS}	E_U,E_G	E_U,E_{iG}	E_U,E_{Asym1}	E_U,E_{Asym2}	E_U,E_G	E_U,E_{iG}	E_U,E_{Asym1}	E_U,E_{Asym2}	E_U,E_G	E_U,E_{iG}	E_U,E_{Asym1}	E_U,E_{Asym2}
0.95	0	0	0.062	0.04	0	0	0.08	0.05	0	0	0.07	0.07
0.90	0	0	0.15	0.06	0	0	3.93	0.06	0	0	1.29	0.53
0.82	0.01	0.01	1.15	0.20	0	0	24.3	0.72	0	0	40.7	0.40
0.73	0.07	0.07	8.86	1.33	0.02	0.01	32.6	1.83	0.01	0	129	15.7
0.65	0.30	0.28	52.0	8.48	0.01	0	142	14.5	0.1	0.02	308	96.5

3) Back-annotate the synthesized gate-level netlist with timing information (standard-delay format (SDF) file) at PVT corners worse than the synthesis PVT corner in step 2.
4) Generate the erroneous output $y[n]$ at different PVT corners by employing an RTL-level simulation of the synthesized gate-level netlist in step 2 using the same input data set $D_{x,U}$ as step 1 and the SDF files generated in step 3 while fixing the operating frequency at f_{op}.
5) Error PMF P_E is obtained at different PVT corners by comparing $y_o[n]$ in step 1 to $y[n]$ in step 3.

IV. Simulation Results

To validate the error analysis, modeling, and characterization, we employ *voltage overscaling* (VOS) in order to generate timing violations and thereby emulate PVT variations. In VOS, the supply voltage is reduced below a critical supply voltage $V_{dd-crit}$, which is the lowest voltage at which the system operates error-free, while keeping the frequency of operation fixed at f_{op}. We define $V_{dd}/V_{dd-crit}$ as the voltage overscaling factor K_{VOS}. In what follows a commercial $45nm$ CMOS process is employed and error PMF P_E of a given DSP kernel/architecture is obtained at each voltage following the characterization flow outlined previously. In certain cases, when we want to study the effect of different input statistics on output error, we use the respective input statistics instead of a uniform one. We focus on adder and multiplier units as these are widely used in DSP designs and form most of the data path in circuits benchmarks. We employ *Kullback-Leibler distance* (KL) to quantify the difference between error PMFs for different input statistics. Given two PMFs P_{E_1} and P_{E_2} of two random variables E_1 and E_2, the KL distance is:

$$KL(P_{E_1}, P_{E_2}) = \sum_e P_{E_1}(e) \log_2 \frac{P_{E_1}(e)}{P_{E_2}(e)} \quad (11)$$

KL measures the distance between two distributions so that $KL(P_{E_1}, P_{E_1}) = 0$ if and only if $P_{E_1} = P_{E_1}$. Usually, two PMFs are quite similar if $KL < 1$.

To verify the relation between word-level (PMF) and bit-level (BPP) input statistics and output error statistics, Table I shows the KL-distance between the error PMFs corresponding to different input PMFs (G, iG, Asym1, and Asym2) and the error PMF P_{E_U} obtained using a uniform input distribution in different 16-bit adders. The error PMFs corresponding to symmetric input PMFs, G and iG, have very small KL distance with P_{E_U}. On the other hand, error PMFs corresponding to asymmetric input PMFs, Asym1 and Asym2, are close to P_{E_U} only at high K_{VOS} where the voltage of the adder is not reduced enough to produce a large number of output errors. As voltage is reduced further, the error PMF of asymmetric input distributions starts to have a very large KL distance compared to P_{E_U}. Note that, $KL(P_{E_U}, P_{E_{Asym1}})$ is greater than $KL(P_{E_U}, P_{E_{Asym2}})$. This is because the Asym1 PMF is more asymmetric than Asym2 PMF (see Fig. 2(a)). Similar trend is observed in Table II for different types of 16-tap FIR filters where error PMFs of symmetric input distributions are close to P_{E_U} while those of asymmetric distributions are quite different. These results support the error analysis and modeling procedure presented in this paper, and specifically, the fact that input distributions with similar input BPPs produce similar output error statistics.

TABLE II

KL DISTANCE BETWEEN ERROR PMFS OF A 16-TAP FIR FILTER UNDER VARIOUS INPUT STATISTICS AND ERROR PMF P_{E_U}.

K_{VOS}	E_U,E_G	E_U,E_{iG}	E_U,E_{Asym1}	E_U,E_{Asym2}
	Direct-Form FIR			
0.95	0.06	0.04	21.6	0.05
0.90	0.94	0.15	63	3.57
0.82	0.92	0.14	33	3.10
0.73	0.03	0.82	227	209
	Transposed-Form FIR			
0.95	0.49	0.13	70	0.53
0.90	0.91	0.38	62	5.78
0.82	0.31	0.08	56	3.41
0.73	0.03	0.89	203	163

References

[1] E. Kim, R. Abdallah, and N. Shanbhag, "Soft NMR: exploiting statistics for energy-efficiency," in *Proc. Int. Symp. System-on-Chip*, Oct. 2009, pp. 52-55.

[2] N. Shanbhag, R. Abdallah, R. Kumar, and D. Jones, "Stochastic computation," in *Proc. of Design Autom. Conf.*, June 2010, pp. 859-864.

[3] R. Abdallah and N. Shanbhag, "Robust energy-efficient DSP systems via output probability processing," in *Proc. of Int. Conf. on Computer Design*, Oct. 2010.

[4] S. Borkar et al., "Parameter variations and impact on circuits and microarchitecture," in *Proc. of Design Autom. Conf.*, June 2003, pp. 338-342.

[5] R. Bahar, J. Mundy, and J. Chen, "A probabilistic-based design methodology for nanoscale Computation," in *Proc. of Int. Conf. on CAD*, Nov. 2003, pp. 480-486.

[6] W. Qian, M. Riedel, K. Barzagan, and D. Lilja, "The synthesis of combinational logic to generate probabilities," in *Proc. of Int. Conf. on CAD*, Nov. 2009, pp. 367-374.

[7] T. Austin and V. Bertacco, "Deployment of better than worst-case design: solutions and needs," in *Proc. of Int. Conf. on Computer Design*, Oct. 2005, pp. 550-558.

[8] N. Shanbhag, "Reliable and energy-efficient digital signal processing," in *Proc. of Design Autom. Conf.*, June 2002, pp. 830835.

[9] G. Varatkar, S. Narayanan, N. Shanbhag and D. Jones, "Stochastic networked computation", *IEEE Trans. on VLSI*, pp. 1-13, 2010.

[10] S. Mitra, N. Saxena, and E. McCluskey, "A design diversity metric and analysis of redundant systems," *IEEE Tran. on Computers*, vol.51, no.5, pp. 498-510, May 2002.

[11] L. Wan and D. Chen, "DynaTune: circuit-level optimization for timing speculation considering dynamic path behavior", in *Proc. of Int. Conf. on CAD*, Nov. 2009, pp.172-179.

[12] Y. Liu, T. Zhang, and K. Parhi, "Computation error analysis in digital signal processing system with overscaled supply voltage", *IEEE Trans. on VLSI*, vol. 18, no. 4, pp. 517-526, Apr. 2010.