A Novel and Fast Method for Characterizing Noise Based PCMOS Circuits

Anshul Singh[1,5], Satyam Mandavalli[1], Vincent J. Mooney III[2, 3, 4,5] and Keck-Voon Ling[3,5]
[1]CVEST, International Institute of Information Technology, Hyderabad, India
[2]School of ECE, Georgia Institute of Technology, Georgia, USA
[3]School of EEE, Nanyang Technological University, Singapore
[4]School of Computer Engineering, Nanyang Technological University, Singapore
[5]NTU-Rice Institute of Sustainable and Applied Infodynamics (ISAID), NTU, Singapore
E-mail: anshul.singh@research.iiit.ac.in, satyam@iiit.ac.in, mooney@ece.gatec.edu, ekvling@ntu.edu.sg

Abstract

Quick and accurate error-rate prediction of Probabilistic CMOS (PCMOS) circuits is crucial for their systematic design and performance evaluation. While still in early stage of research, PCMOS has shown potential to drastically reduce energy consumption at a cost of increased errors. Recently, a methodology has been proposed which could predict the error-rates of cascade structures of blocks in PCMOS. It requires error-rates of unique blocks to predict the error-rates of a multi-block cascade structure. The "Three stage model," which accounts for different noise filtering for different paths in a circuit, has been proposed to characterize unique blocks. While the results obtained from the three stage model produced accurate error-rates for a multi-block cascade structure, the procedure for its characterization is computationally expensive. In this paper, we propose a new method for characterizing the three stage model that not only provides accurate results but is also computationally cheap.

Keywords

PCMOS circuits, characterization, dynamic noise analysis

1. Introduction

As the feature size of CMOS devices reduces, statistical behaviors such as noise, parametric variations, defects, etc., become more prominent, thus making deep submicron devices susceptible to errors. Also, by today's usability and environmental standards, energy consumption is a top design criterion for computing devices. Voltage scaling is one of the most popular approaches to achieve low power computing, but it invariably lowers the noise margin which makes the system more susceptible to errors. To meet the challenges of the two competing needs – reliability and energy consumption – Probabilistic CMOS (PCMOS) computing proposes to include occasional errors caused by erroneous chips [1] with trade-offs between correctness of circuit operation and energy consumption [2]. It has also been shown that errors can be traded off with increased speeds of the circuit operation [3].

With the present generation low level simulators like HSPICE, which are extremely time consuming, quick and accurate prediction of probabilistic circuits becomes crucial for their systematic design and performance evaluation. Lau et al. proposed a methodology to quickly predict the error rates of cascade structures of blocks [4]. The methodology requires error-rates of constituent blocks to predict the error-rate of a whole cascaded structure of blocks. Singh et al. proposed a new model which provides the error-rates of these constituent blocks [5]. Although the results obtained from the new model are good for accurate error-rate prediction, the procedure requires SPICE simulation of the model for a large number of samples. SPICE simulation for a large number of samples makes the procedure computationally rigorous, hence, solving only one part of the problem, that is, accurate error-rate prediction. In this paper, we propose a new computationally cheap method for obtaining error-rates of those constituent blocks using the three stage model. We still use SPICE simulations, but instead of error-rates, we now obtain different information from SPICE that is not only sufficient for accurate error-rate prediction but is also computationally efficient. In short, in our new approach, we do dynamic noise analysis on the model proposed in [5] and a new statistical analysis of time domain noise to come up with a computationally cheap characterization procedure.

The rest of the paper is organized as follows. Section 2 discusses the prior work in the field of error-rate prediction of PCMOS circuits. Section 3 explains our new approach to characterize PCMOS circuits, and Section 4 compares the results obtained from HSPICE simulation and our prediction approach. Section 5 concludes the paper.

2. Prior Work

In this section we briefly go through the various works that have been done in field of PCMOS circuits and its error-rate prediction. We also look at the basics of dynamic noise margin which we use in our new approach for PCMOS circuit characterization.

2.1. Modeling PCMOS Circuits

Korkmaz et al. [6, 7] and George et al. [8] modeled a noisy PCMOS circuit by adding equivalent noise sources at the outputs of the deterministic version of the circuit. This approach goes with the assumption that the equivalent noise source at the output estimates the impact of the noise present in each gate or transistor in an actual noisy circuit. Fig. 1 shows a probabilistic full adder (PFA) built from a deterministic FA. Noise sources coupled at the outputs of a deterministic circuit are voltage controlled voltage sources (VCVSes) whose controlling voltages are obtained from the distribution of noise being modeled.

2.2. Cascade Math Model

Lau et al. proposed a methodology to predict the error rates of cascade structures of blocks as shown in Fig. 2 [4].

It consists of two main steps: (i) characterization of each unique probabilistic block and (ii) evaluation of mathematical equations that model error generation and propagation across the blocks. It has been shown in [4, 5] that the methodology can accurately and quickly predict the error-rates of the cascade structure, given the error-rates of the constituent blocks. A block is a collection of circuit elements with no restrictions on the number of inputs, outputs and types of circuit elements. Information obtained from characterization of blocks is used to predict the error-rates of a multi-block circuit. In this paper we will refer to the methodology of [4] as the cascade math model.

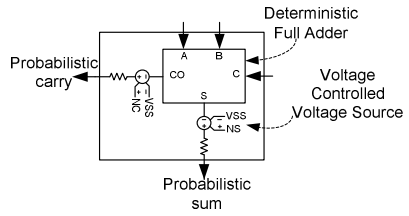

Figure 1: A Probabilistic Full Adder

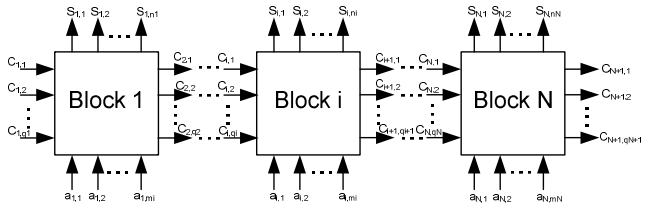

Figure 2: Cascade Structure of Blocks

In terms of circuits and circuit elements, to predict the error-rate of a *probabilistic circuit* (PrC), the cascade math model [4] requires the error-rate of each unique *probabilistic circuit element* (PCE) constituting that PrC, where a *circuit* means a combination of pull-up pull-down (pu-pd) networks constituting a higher level function of interest, such as arithmetic, and a *circuit element* means a subset of a circuit. PCEs whose error-rates are different from that of other constituting PCEs are termed *unique*. Determining error-rates of PCEs is called characterization of probabilistic circuit elements [4, 5]. Once the error rates of unique PCEs are known, the math model can predict the error rate of the PrC.

2.3. The Three Stage Model and Characterization Procedure

Singh et al. propose a three stage model for PCMOS circuit characterization which takes care of path dependent filtering effects present in circuits [5]. According to the three stage model, design of the following three components play a central role in the probabilistic response of a circuit: the three components are (1) Probabilistic Circuit Element Under Characterization (PCEUC), (2) Load of PCEUC which is called Filter Circuit (FC) and (3) The Load of the FC (LoFC). Fig. 3 shows the model.

Singh et al. describe a procedure to find unique PCEs in a PrC using the three stage model [5]. Characterization of PCEs is done by HSPICE simulations by performing the experiment shown in Fig. 4. Out of the two duplicate three stage models of the circuit element under characterization (CEUC), one is probabilistic (PCEUC) producing noisy output and the other is deterministic (DCEUC) producing deterministic outputs. FCs and LoFCs for the two duplicates are kept deterministic so that the filtering effect of the filter circuit is captured; also, the FCs are kept in such a state that there is no logical masking [5]. The two duplicates – PCEUC and DCEUC – are simulated with a large number of inputs (typically 100,000), and the outputs of the two filter circuits (FC) are compared as shown in Fig. 4. A mismatch in the digital result is treated as an error. The number of errors per output of the FC is calculated to determine the error-rate and is assigned as the error-rate of the output of PCEUC driving that FC. If an FC has, say, *n* outputs, then *n* error rates are assigned to that specific PCEUC output corresponding to each output of FC.

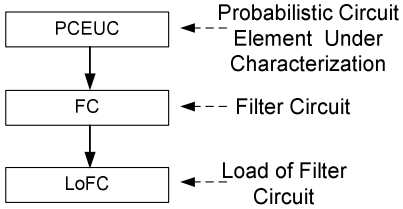

Figure 3: The Three Stage Model for a PCMOS Circuit Element

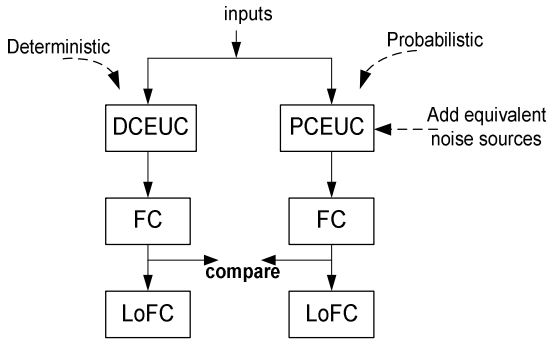

Figure 4: Characterizing PCEs

2.4. Dynamic Noise Margin

Noise margin, to quickly summarize, gives the maximum noise amplitude that can be tolerated by a circuit without affecting its correct operation. Static noise margin gives the noise margin of a circuit assuming the noise to be a DC signal [9, 10]. But, a very well known fact is that if noise is present in the form of a pulse, the noise amplitudes are allowed to be higher than the static noise margins without affecting the proper logic states of a digital circuit. For very long noise pulses the noise margins are determined by static noise analysis, but for short pulses the noise margins increase. Dynamic noise analysis provides the noise margin for noise pulses [11]. For shorter pulses the margin is higher while for longer pulses the dynamic noise margin tends to converge with the static noise margin. Fig. 5 shows the two noise margins for an inverter operating at 0.8 V. We use Synopsys 90 nm library for all our simulations.

Ding and Mazumdar extend the know-how's of obtaining static noise margins to define dynamic noise margins of

logic gates [12]. Specifically, it deals with obtaining a set of voltage values (V_{IL}, V_{IH}, V_{OL} and V_{OH}) satisfying the following two relationships

$$max(V_{in}(t)) \leq V_{IL} \Rightarrow min(V_{out}(t)) \geq V_{OH} \quad (1)$$

$$min(V_{in}(t)) \geq V_{IH} \Rightarrow max(V_{out}(t)) \leq V_{OL} \quad (2)$$

where the input ($V_{in}(t)$) and output ($V_{out}(t)$) voltages of the gate are time dependent, such that the dynamic noise margin (DNM) is maximized.

$$DNM = min\{V_{OH} - V_{IH}, V_{IL} - V_{OL}\} \quad (3)$$

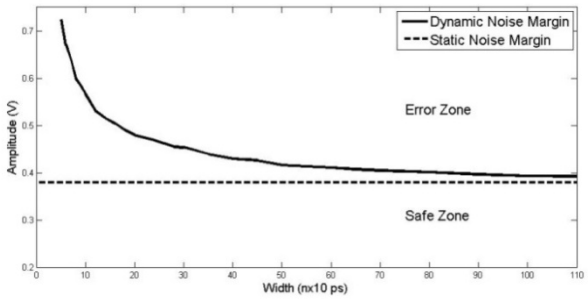

Figure 5: Static and Dynamic Noise Margin

Ding and Mazumdar also provide two methods for obtaining the four voltage values (V_{IL}, V_{IH}, V_{OL} and V_{OH}): the single gate unity gain based approach and the maximum square based approach [12]. Fig. 6 shows the voltage transfer curves of an inverter for different input pulse widths obtained by transient analysis in HSPICE, and Fig. 7 shows the values of the four voltages for an inverter obtained using the single gain unity points for a particular input pulse width. Please note that in Figures 6 and 7, H is used for the output curve in response to a low-high-low input pulse, and L is used for the output curve in response to a high-low-high input pulse. One more point that should be noted is that DNMs are obtained for single gates (single pull-up/pull-down networks) and not for a system of connected gates. DNM obtained directly for the system shows higher noise tolerance limits as compared to its actual limit.

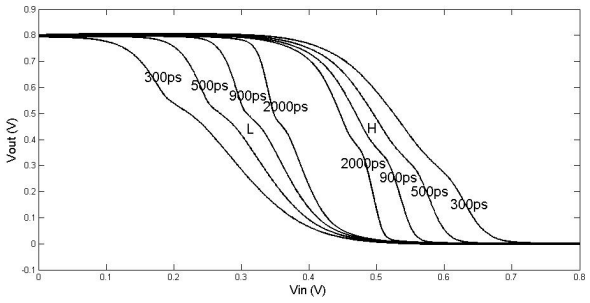

Figure 6: Voltage Transfer Curve for Different Pulse Widths for an Inverter

3. A New Method for Characterizing Noise Based PCMOS Circuits

In Section 2, we briefly discussed the three stage model and its characterization method [5]. The results obtained from the three stage model are good for reasonably accurate error-rate prediction of PrCs, but since the characterization procedure requires simulation of the model for large number of samples, it is a computationally rigorous procedure requiring computation time comparable to SPICE simulation of PrCs. In this section, we propose a new method/approach for characterizing noise based PCMOS circuits which not only provides good results, in terms of accuracy of predicted error-rates, but is also much more computationally efficient than the previous approach. In the following paragraphs we give the big picture of the method; then, in the next subsection, we go into the details of the proposed approach.

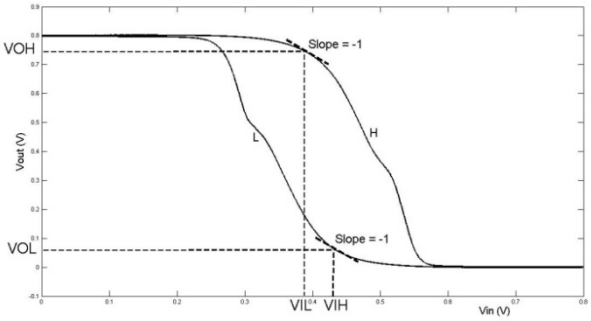

Figure 7: Single Gate Unit Gain Approach for an Inverter

According to the definition of characterization [4, 5], it is a procedure for obtaining error-rates of PCEs in a PrC, and the three stage models of the PCEs are used to characterize them [5]. By characterization of a PCE, what we are trying to measure is how many of the erroneous noise pulses output by that PCE actually cause errors in the next stage of the circuit. To obtain the number of actual errors created, we count the number of errors at the output of the load of the PCE (which we call the Filter Circuit (FC) in the three stage model) and tag this number as the error-rate of the PCE. If we look at the same procedure from the FC's point of view, what we are trying to find is its noise tolerance. The noisy pulses which can be tolerated by the filter circuit are not treated as errors, and, we say, those pulses are "filtered out" by the FC; but pulses which cannot be tolerated are converted into errors. Noise margin is a well established concept which gives the measure of this tolerance for logic gates. There are two noise margins as discussed in Section 2 – Static Noise Margin (SNM) and Dynamic Noise Margin (DNM) – which are obtained, respectively, from Static Noise Analysis (SNA) and Dynamic Noise Analysis (DNA). SNM, as mentioned in Section 2, treats noise as a DC signal, which is not true in our case. Hence, we make use of DNA in finding the error-rates of PCEs.

In our new approach, we perform DNA on the FC of the three stage model of a PCE through SPICE simulations, which is computationally very cheap as compared to SPICE simulation of the three stage model for a huge number (e.g., 100,000) of samples. We also perform a new statistical analysis on the time domain noise generated by the PCEs, which we call structural analysis, to obtain the statistical information of its various features, required to predict error-rates of PCEs using the DNA of their FCs. We first talk about the structural analysis of time domain noise, then we discuss the dynamic noise analysis of the three stage model.

We next combine the information obtained from structural noise analysis and dynamic noise analysis to come up with error-rates of PCEs. Finally, we discuss noise filtering in logic gates.

3.1. Structural Description of Time Domain Noise

Electronic noise is one of the most widely studied topics and is described in a number of ways using different approaches and methods depending upon the specific needs and requirements [13, 14]. In the domain of noise error prediction in logic circuits, static noise margin (SNM) is the most widely used measure [9, 10]. Static noise analysis assumes the noise to be a DC signal and considers only the amplitude of the noise before declaring it a potential error creator. Hence, for static noise analysis, amplitude distribution of the time domain noise is the only information needed in predicting errors in digital systems. But such an assumption regarding electronic noise, a random fluctuating quantity, is highly unreasonable. Dynamic noise analysis, on the other hand, considers the entire time varying noise waveform [12]. It not only takes into account the amplitude but also the duration of the noise pulse and is a more appropriate scheme for predicting possible noise violations, and hence, error-rates in our case.

To predict the errors using dynamic noise margin, we not only need the amplitude distribution of the noise pulses but also the distribution of the durations (widths) of the noise and a possible relationship (if one exists) between the amplitudes and widths. We have extensively searched the available literature on the statistics of Gaussian noise but are not aware of any work that describes its pulse width distribution and any relationship between pulse amplitudes and widths. Hence, in this subsection we first talk about the amplitude distribution of noise, then we propose a new metric describing the width distribution, and finally we look at a possible relationship between the amplitudes and the widths of noise. We call these descriptions of noise as *Structural Descriptions* since these metrics describe the structure of time varying noise. In this paper we restrict ourselves to Gaussian noise and use it to show the various distributions and relations. Any mention of noise in this paper refers to Gaussian noise unless otherwise stated. The amplitude, width distributions and the relationship between the amplitudes and the widths for Gaussian noise is obtained from MATLAB simulations. The data points for the noise are also obtained from MATLAB. Noise is assumed to be piecewise linear and hence we simply join the noise data points by straight lines to construct the noise. Fig. 8 shows a small sample of noise used in this paper. The data points of the noise are kept at a separation distance of 500 picoseconds (ps), and, since mean value of electronic noises, such as thermal noise, etc., is generally zero [14], the data points obtained from MATLAB are for zero mean Gaussian noise.

3.1.1. Pulse Amplitude Distribution

By a pulse we mean a transient wave above/below a certain voltage threshold. A positive pulse is a wave above 0V and a negative pulse is a wave below 0V. By pulse amplitude we mean the maximum amplitude of the positive pulse and minimum amplitude for the negative pulse. Fig. 8 shows positive and negative pulses with their respective amplitudes. Although the instantaneous value of randomly varying noise is not predictable, the probability of the amplitude of noise falling within a certain amplitude range is predictable in a statistical sense. Pulse Amplitude Probability Density Function (PAPDF) gives the probability that the amplitude of a pulse will fall within certain amplitude range. Fig. 9 shows PAPDF for the Gaussian noise for various Root Mean Square (RMS) values of noise. By RMS we mean standard deviation value of the noise. As can be seen from the figure, the PAPDF follows a normal distribution [15] and is given by function $a(V)$,

$$a(V) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{V^2}{2\sigma^2}\right)} \qquad (4)$$

where parameter σ is the standard deviation, which we call RMS in this paper, and V is the variable for amplitude (continuous variable). Please note that mean value is assumed to be zero.

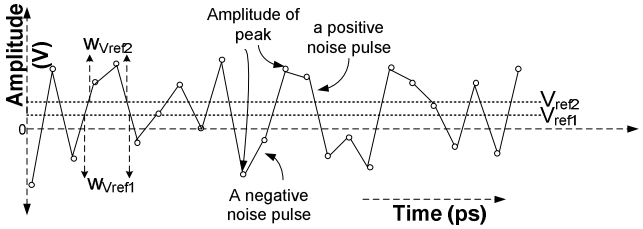

Figure 8: A Zero Mean Gaussian Noise

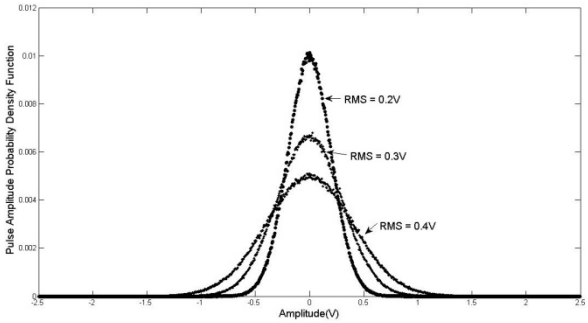

Figure 9: Pulse Amplitude Probability Density Function

3.1.2. Pulse Width Distribution

By pulse width we mean the duration of a pulse at some reference voltage, V_{ref}. Generally, the duration of a pulse is defined at a reference voltage equal to half of its amplitude [10]. But since we are working with a long sequence of noise and not a single noise pulse, we fix a V_{ref} and then look at the widths of the pulses crossing it to obtain the width distribution of noise for that V_{ref}. Since a noise pulse will give different widths for different reference voltages, as can be seen from Fig. 8, we define widths, and hence width distribution, for a reference voltage. V_{ref} can be changed but then widths would be defined for that V_{ref}. As is the case with amplitude distribution of noise, the exact value of the width of a noise pulse cannot be predicted but the probability of it falling in a certain range can be obtained in

a statistical sense. Pulse Width Probability Density Function (PWPDF) gives the probability that the width of a pulse will fall within a certain range. Please note that for all our simulations regarding widths, we have fixed this range to 10ps. If we say a pulse has a width of n, we mean the width lies between n - 4 to n + 5 ps. Since width of a pulse cannot be negative, width 0 covers the range 0 to 5ps.

We have found that RMS value of the Gaussian noise and V_{ref} are the two parameters that define PWPDF. Fig. 10 shows the variation of PWPDF with V_{ref} for a particular RMS value. Fig. 10 shows that for a fixed RMS with increasing V_{ref}, widths become smaller and distribution of the pulse widths gets restricted to a smaller range. This is obvious since we have triangular pulses whose widths decrease as V_{ref} moves towards their peak amplitudes. Fig. 11 shows the variation of PWPDF with RMS for a particular V_{ref}. Again, as the RMS values increases, the peaks of the pulses become higher and the widths become larger at a particular V_{ref}. Figures 10 and 11 show that the PWPDF is a Gaussian function. We use curve fitting techniques to obtain the PWPDF and is given by function $w(W)$,

$$w(W) = ae^{-(W-b)^2/c^2} \qquad (5)$$

where parameters a, b and c are polynomial functions of RMS value and V_{ref} and W is the variable for width. Please note that W is a discrete quantity and not a continuous variable. Hence, we will use summation instead of integration for W.

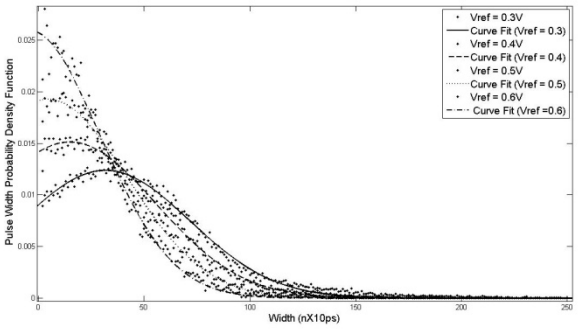

Figure 10: Dependence of PWPDF on Reference voltage, V_{ref}

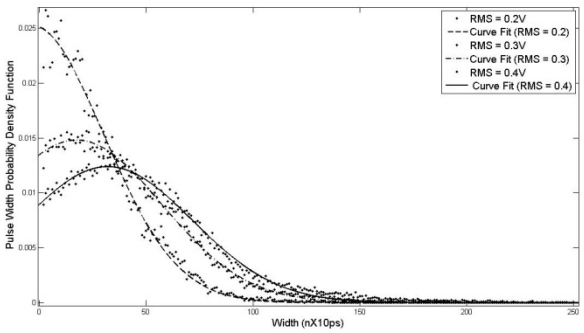

Figure 11: Dependence of PWPDF on RMS value

3.1.3. Amplitude-Width Relation

The joint pulse amplitude width probability density function (JPAWPDF) is the function which gives the probability of a noise pulse to lie within a certain amplitude range and a certain width range. We have found that this information about a noise is sufficient to predict the number of possible errors in a digital circuit using dynamic noise analysis. Obtaining (JPAWPDF) requires calculation of PAPDF of noise pulses for every width. Since the widths, in our case, vary from 0 to 2000, each with a different PAPDF, it becomes a difficult task to obtain the joint probability density function analytically from the amplitude and width PDF. We even tried using curve-fitting techniques on the noise data to find an expression for the joint amplitude width PDF but could not come with an expression which describes the variations of the data with RMS and reference voltage reasonably well. Hence, we try to find a relation between amplitude and width of noise pulses graphically.

The amplitude-width relation relates the amplitudes of noise pulses with their widths. We have found that for Gaussian noise there does exist a relation for smaller widths. Fig. 12 shows a plot of noise pulses with their widths (obtained for certain V_{ref}) and amplitudes. A dot in the plot is a noise pulse with corresponding width and amplitude. As can be seen from the plot, the maximum amplitudes of the noise pulses with smaller widths do not exceed a certain bound. But this relation does not hold true for bigger widths. Fig. 13 shows the plot of maximum amplitudes for each width obtained from the plot in Fig. 12. As can be clearly seen from the figure, noise pulses with smaller widths have maximum amplitudes below a certain bound, but for bigger widths such a condition does not hold true as the distribution becomes random. We use this fact and come up with a new metric which we found to be sufficient in predicting the error-rate using dynamic noise analysis. We assume PAPDF and PWPDF to be independent below the max amplitude bound for smaller widths (for which the correlation exists) and for higher widths we treat them independent with no bound on the maximum amplitude.

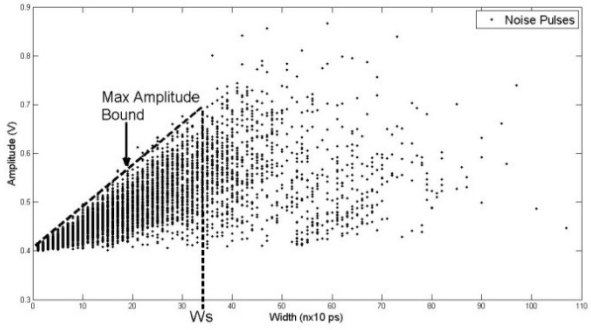

Figure 12: Amplitude-Width Distribution of Gaussian Noise

The new metric is Max-amplitude per width (MAW). For lower widths the max-amplitude shows a strong correlation with widths but becomes independent for higher widths. Hence, we obtain the max-amplitude per width for smaller widths. As the max-amplitude per width varies with RMS and V_{ref}, they are again treated as its parameters. We

use curve-fitting to obtain the expression for max-amplitudes which are valid only for smaller widths and is given by the function $m(W)$:

$$m(W) = pW + q \qquad (6)$$

where parameters p and q are polynomial functions of RMS and V_{ref}. Please note that the definition of "smaller width", which we represent by Ws (see Fig. 12), changes with V_{ref} and RMS value. For example, for Gaussian noise with 0.2V RMS and 0.4V V_{ref} the correlation exists for widths up to 350-400 but for 0.6V V_{ref} the correlation exists only till 150-200. Therefore, RMS and V_{ref} should be used as a parameter to define "smaller width" while using $m(W)$. We use a lookup table which defines smaller widths for different RMS and V_{ref}.

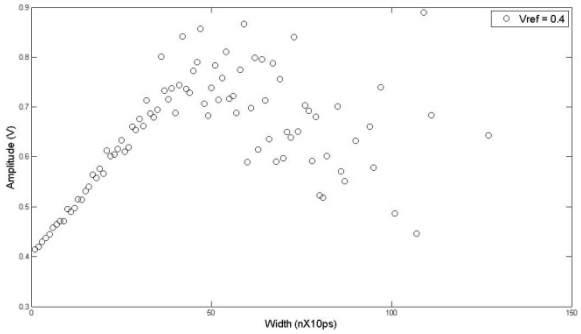

Figure 13: Maximum Amplitude for each Width

3.2. Dynamic Noise Analysis

Dynamic Noise Margin, as discussed in Section 2, provides the information whether a noise pulse with certain amplitude and width can be tolerated by a logic gate or not. Since we are trying to predict error-rate, we are interested in the noise pulses which will not be tolerated by the gate. V_{IH} and V_{IL} are two parameters of the logic gate which provide us the information whether a noisy signal will be treated as logic 0 or logic 1. The V_{IL} curve of a gate gives the maximum voltage that can be considered as logic 0 by the gate when the input is going from logic 0 to 1 for different pulse widths. Similarly, the V_{IH} curve gives the minimum voltage that can be considered as logic 1 by a gate for different pulse widths when the input is going from logic 1 to 0. We know that positive noise pulses create 0 to 1 errors and negative noise pulses create 1 to 0 errors. By 0(1) to 1(0) error we mean a signal which is at logic 0(1) but is treated as a 1(0) because of the noise present. A positive noise pulse, with width W_N, should have amplitude higher than that specified by V_{IL} curve for width W_N to potentially create a 0 to 1 error. Similarly, to create a 1 to 0 error a negative noise pulse, with width W_H, should have amplitude lower than that specified by $V_{IH} - H_{signal}$ for width W_H, where H_{signal} is the nominal voltage of signals at logic one which, generally, is VDD (power supply voltage). Fig. 14 shows the two curves (V_{IL} and $V_{IH} - H_{signal}$) and the error-zone for a logic gate. If a noise pulse has an amplitude and width which lies in the error-zone defined by the V_{IL} and V_{IH}-H_{signal} curves of a logic gate, the noise pulse, potentially, will be treated as an error by the logic gate.

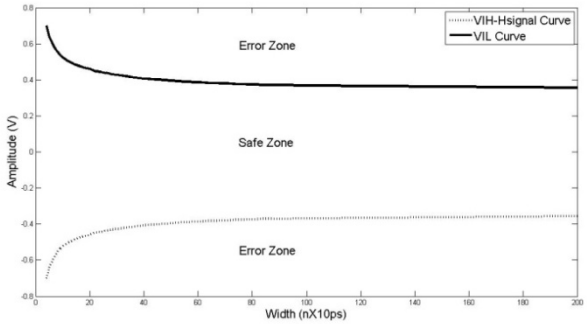

Figure 14: V_{IL} and $V_{IH} - H_{signal}$ Curves

3.2.1 Procedure for finding V_{IL} and V_{IH} Curves

We obtain the V_{IL} and V_{IH} curves for logic gates by HSPICE simulations (transient analysis). We use single gate unity gain method [12] to calculate V_{IL} and V_{IH} because it is easy to use as compared to the maximum square method [12] and also the variations of the results obtained from the two methods are negligible for widths of our interest.

To obtain V_{IL} of a logic gate, say inverter, we simulate the inverter with a positive triangular pulse (low-high-low) as input and obtain the voltage transfer curve as shown in Fig. 7. Please note the voltage transfer characteristic of logic gate depends on the logic gate, on the widths of the input pulses (Fig. 6), the load which gate is driving and also to some extent on the gate driving the logic gate. These are actually the three stages of the three stage model — the driving gate (PCEUC), the logic gate (FC) and the load of the logic gate (LoFC). Hence, to obtain the V_{IL} of an inverter, we construct the three stage model of the inverter where inverter is the FC with its driving gate and load and use a voltage controlled voltage source (VCVS) at the output of the driving gate whose controlling voltage is a positive triangular pulse of amplitude VDD (supply voltage) with a certain specified width. We obtain the voltage transfer curve of inverter (H curve in Fig. 7) and find the input voltage which gives -1 slope on the curve; this input voltage is V_{IL}. We repeat this procedure for triangular pulses with different widths to obtain the V_{IL} curve (Fig. 15). The reference voltage (V_{ref}) used for the calculation of widths of the triangular input pulses is obtained by finding out the V_{IL} for a triangular pulse of very large width (2000, in our case). The motivation for using such a reference voltage will become clear in Section 3.3. Similarly, to obtain V_{IH} we use negative triangular pulses (high-low-high) where the base of the pulse is at VDD and peak at Gnd. Again, we obtain the voltage transfer curve of the inverter (L curve in Fig. 7) and find the input voltage which gives -1 slope on the curve; this input voltage is V_{IH}. We repeat the procedure with different pulse widths to obtain the V_{IH} curve (Fig. 15). V_{ref} for the width calculation of input triangular noise pulses is obtained by finding V_{IH} for a very large width. Since we are more interested in $V_{IH} - H_{signal}$ curve than the V_{IH} curve, we subtract H_{signal} from the V_{IH} curve to obtain the $V_{IH} - H_{signal}$ curve.

We represent V_{IL} and V_{IH} curves by a function, $d(W)$

$$d(W) = \frac{eW + f}{W + g} \qquad (7)$$

where e, f and g are parameters obtained from curve-fitting and W is the variable used for width. We use $d(W)_{V_{IL}}$ and $d(W)_{V_{IH}}$ to differentiate between the two curves. Hence, V_{IL} and V_{IH} curves for a gate are represented by six curve-fitting parameters, three for the V_{IL} curve and the other three for the V_{IH} curve.

3.2.2 V_{IL} and V_{IH} Curve for Multi-Input Gate

In the previous subsection we discussed the procedure of obtaining V_{IL} and V_{IH} curves for a gate. If a gate has multiple inputs, then V_{IL} and V_{IH} curves should be obtained for each input while keeping the other inputs in such a state that the gate responds to the input for which we are calculating the V_{IL} and V_{IH} curves. For example, for a two input NAND gate, to obtain V_{IL} and V_{IH} curves for one input, the other input should be kept at logic 1 so that there is no logical masking.

We have found that V_{IL} and V_{IH} curves slightly vary for different inputs of the same gate and there is also a dependence of the curves on the state of the other inputs of the gate. Fig. 15 shows the two V_{IL} and V_{IH} curves obtained for a 2-input NAND gate for its two inputs. For first order analysis such variations can be ignored but for very accurate results such variations should be taken into account. In this paper, we do not take into account such variations instead we take the average of the V_{IL} and V_{IH} curves obtained for different cases of a gate. For example, for a NAND gate, we take the average of V_{IL} curves obtained for input A and input B and use it as the V_{IL} curve of the NAND gate.

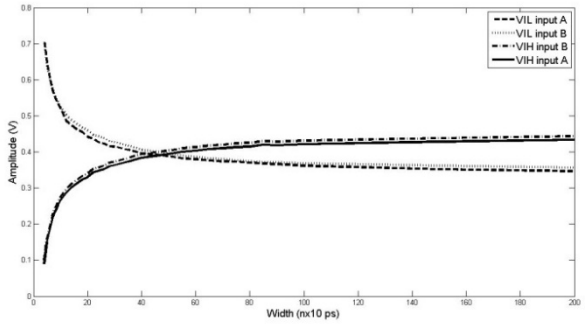

Figure 15: V_{IL} and V_{IH} Curves for 2-Input NAND Gate

3.3. Error-rate Calculation

In this section, we combine the structural information of noise and V_{IL} and V_{IH} curves for a gate to obtain the error-rate of its driver gate or the Probabilistic Circuit element under characterization (PCEUC) in the three stage model notation [5]. Fig. 16 shows the noise pulses that need to be counted as the potential 0 to 1 error creators (above the V_{IL} curve labeled as error-zone). We show how to obtain 0 to 1 error-rate from the V_{IL} curve and structural information of noise. The 1 to 0 error-rate can be calculated similarly. The only difference that exists between the two is that we use $V_{IH} - H_{signal}$ curve instead of V_{IL} curve to determine the error-zone.

To obtain the error-probability/rate for 0 to 1 errors we use three equations given below. We use superscript H for 0 to 1 errors and L for 1 to 0 errors.

$$e_1^H = \sum_{W_C^H+1}^{W_S^H} [w(W) * \int_{d(W)_{V_{IL}}}^{m(W)} a(V)dV] \qquad (8)$$

$$e_2^H = \sum_{W_S^H+1}^{W_{max}} [w(W) * \int_{d(W)_{V_{IL}}}^{\infty} a(V)dV] \qquad (9)$$

$$e^H = e_1^H + e_2^H \qquad (10)$$

where $a(V), w(W)$ and $m(W)$ are the PAPDF, PWPDF and MAW functions described in Section 3.1. $d(W)_{V_{IL}}$ is the V_{IL} curve for a logic gate and * is used for multiplication. W as mentioned before is a discrete quantity and can only take integer values in multiples of 10, i.e., 0, 10, 20 and so on. The reason for W taking such values comes from the fact that we have restricted our range for width calculation to 10ps. e_1^H calculates the probability of a pulse to be in the zone bounded by Max Amplitude per Width curve and V_{IL} (error zone 1 in Fig. 16). In equation 8, W_C^H is the crossover width where the curves $m(W)$ and $d(W)_{VIL}$ meet. There are no noise pulses above the V_{IL} curve before W_C^H. Hence, we start calculating the probability of erroneous pulses after the crossover width till the maximum "smaller width" (for which the correlation between amplitude and width exists). The maximum "smaller width" is denoted by W_S^H and is shown in Fig. 16. The limits for amplitude distribution in this zone goes form $d(W)_{V_{IL}}$ (lower bound) to $m(W)$(upper bound). e_2^H calculates the probability of a noise pulse to be in the zone which is not bounded by Max Amplitude per Width curve (error zone 2 in Fig. 16). In equation 9, we start calculating the probability after W_S^H and go till W_{max} which in our case is 2000 as there is no noise pulse with width greater than that. The limits for amplitude distribution in this zone goes from $d(W)_{V_{IL}}$ (lower bound) to ∞ as there is no maximum amplitude bound. We add the probability of error zone 1 and zone 2 in equation 10 to obtain the error-rate, e^H, of 0 to 1 errors. Similarly, we obtain the error-rate, e^L, for 1 to 0 errors. e^H and e^L are the characterization results for a three stage model and is used by the cascade math model to predict the error-rate of a PrC.

In Section 3.2.1 we discussed about obtaining $\boldsymbol{V_{ref}}$ by calculating V_{IL} for a triangular pulse with very big width. We do this to synchronize the widths of V_{IL} curve and noise pulses. V_{IL} for a very big pulse gives us the minimum value of V_{IL} for that particular V_{IL} curve. We will call it V_{ILmin}. Since we need to count noise pulses which have amplitudes greater than V_{ILmin}, we use this value as $\boldsymbol{V_{ref}}$ for calculation of all our noise parameters. Now that we are using the same $\boldsymbol{V_{ref}}$ for both V_{IL} curve and noise pulses, the widths are synchronized and they share a common axis in amplitude-width plot and hence we can use our equations to calculate the error-rate. Another question that arises is, why not fix

$\boldsymbol{V_{ref}}$ to an arbitrary small value and use it to obtain $\boldsymbol{V_{IL}}$ and noise parameters. We have tried using this approach but the results are not as good as those obtained by the above mentioned approach.

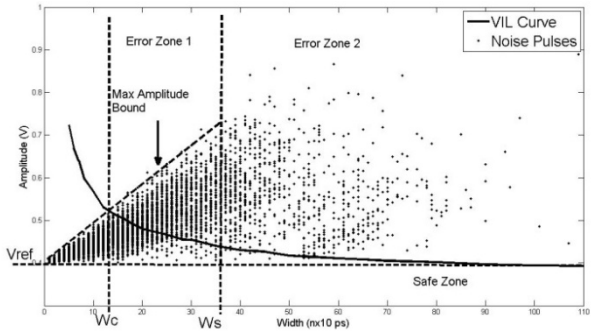

Figure 16: Error-Zone (0 to 1Errors)

3.4. Discussion on Noise Filtering

Singh et al. in [5] explained noise filtering from the point of view of delays in circuits. Noise filtering happens when the duration of a noise pulse is shorter than the propagation delay of the circuit. On looking closely we see two mechanisms happening in a gate, one mechanism decides whether a pulse possesses enough width and amplitude to create a logic change at the output, which is also known as inertial delay [16] (information that we obtain from the V_{IL} and the V_{IH} curves of a gate) and the other mechanism (transportation delay [16]) delays the output response by some time if the first mechanism produces an output response.

We have observed that a gate interacting with noise pulses filter out the pulses depending upon its inertial delay and the outputs of the gate are legal pulses which do not suffer from mid value amplitudes and width constraints, and, almost always create a response in the subsequent gates, that is, meet the inertial delay constraints of the gates. The transportation delay mechanism of the gates delays these pulses and shifts them by some amount every time they pass through a gate. Transportation delay does not filter out these erroneous pulses but only shifts them in time. If we sample outputs, we observe that erroneous pulses which get captured at one stage of a circuit might not get captured in the later stages because of their delayed presence, because these pulses are not as wide as the input signals. But at the same time, we also observe that erroneous pulses which were not captured previously getting captured in the later stages because of the shifting property of transportation delay. Logical masking [5] of erroneous pulses also comes into the picture while analyzing whether an erroneous pulse captured before will be captured later by the sampling unit or not. Although an important mechanism, we have observed that for first order prediction of error-rate, as will be shown in Section 4, shifting mechanism of transportation delay can be neglected. In this paper, we focus on the inertial delay (dynamic noise analysis) and logical masking and show that our approach for characterizing circuits yields reasonably accurate results.

4. HSPICE Simulations and Results

In this section we validate our methodology of characterizing PCMOS circuits by HSPICE simulations of few logic gates and a probabilistic Ripple Carry Adder (PRCA). In particular, we simulate an inverter, 2-input NAND, 2-input NOR, elements of an FA and a 12-bit probabilistic RCA (PRCA) and compare the results predicted by our approach. We use the cascade math model for error-rate prediction of PRCA which uses the results obtained from the proposed method of characterizing PCEs constituting PRCA. We also compare the computation time for characterization with the previous method of simulation and show that with the proposed approach, error-rates of noise based PCMOS circuits can be predicted with reasonable accuracy with huge savings in computation time.

4.1. Modeling Noise Based PCMOS Circuits

Since dynamic noise analysis works only for single pull-up pull-down (pu/pd) network, in this paper we slightly differ from the previous approach of modeling noise based PCMOS circuits. As mentioned in Section-2, a PCMOS circuit element is built from its deterministic counterpart by adding equivalent noise sources at its outputs. In this approach, a circuit element can have any number of pu-pd networks. In our present approach, we add noise sources after every pu-pd network. As mentioned before, we call a single pu-pd network a gate. Hence, in our case the most basic circuit element is a gate. A PFA which was previously built from a deterministic FA (see Fig. 1) is now built from its probabilistic gates which in turn are built by adding noise sources at the outputs of deterministic gates. We use Mirror-Adder as our FA in this paper, which has two pu-pd networks or 2 gates. Fig. 17 shows the Probabilistic Mirror Adder (PFA) used in this paper. In our FA, we call the carry generating pu-pd network as the Carry gate and the sum generating network as the Sum gate.

4.2. Simulating Noise Based PCMOS Circuits

As mentioned before, we construct a probabilistic circuit by adding noise sources at the output of each pu/pd network of the initial deterministic circuit. We use Synopsys 90nm generic library for simulations. A noise source is a voltage controlled voltage source (VCVS) with a small resistor in series (typically 1 ohm). The random noise data for the noise sources are generated by MATLAB and have a Gaussian distribution with zero mean and non-zero root-mean-square (RMS) value. Following [6, 7], we use 0.2 and 0.3V as the RMS value for the small logic gates and only 0.2V as the RMS value for PRCA. This is intended to model a possible thermal noise in a future transistor technology, e.g., 12nm. Inputs to the probabilistic circuits are uniformly distributed random bits and are obtained from MATLAB. We simulate all designs with 100,000 samples each. The inputs are fed to the circuits at a clock period of 2ns, and the noise samples are added to the outputs of each pu/pd in each circuit design every 0.5ns. We simulate the probabilistic designs for three voltages which are 0.8, 0.9, and 1.0V. For PRCA, we obtain the error probabilities from the sum bits for each of the three respective voltages.

4.2.1. Simulation Setup for Logic Gates

Fig. 18 shows the experimental setup used for logic gates. Since we are trying to calculate how many errors are actually seen by the various FCs, we use the characterization procedure used in [5] and shown in Fig. 4. We construct two duplicates of a three stage model where we use an inverter as the CEUC and 4-inverters (FO4) as LoFC. FCs are the various logic gates which decides whether a noise pulse generated by the noisy inverter is an error or not. Hence, we use an inverter, a 2-Input NAND, 2-Input NOR, a Carry gate and a Sum gate as FCs and obtain error-rates from their output. Inputs of the gates which are not driven by the noisy inverter are fixed to some logic value so that there is no logical masking.

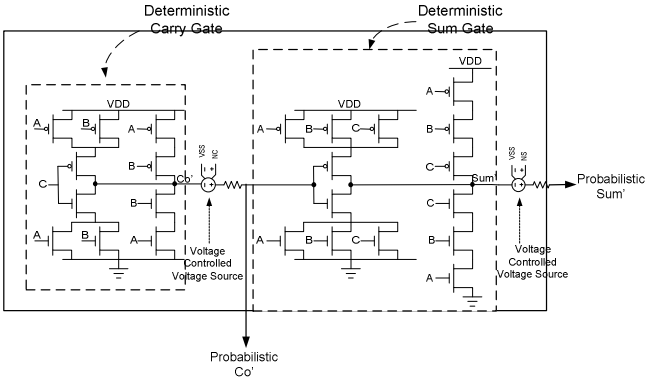

Figure 17: Probabilistic Full Adder Built from its Probabilistic Gates

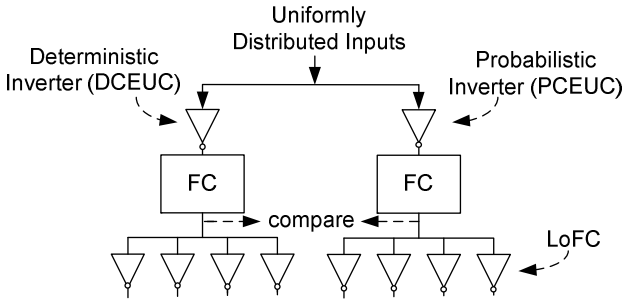

Figure 18: Experimental Setup

4.3. PCMOS Circuit Characterization

The procedure for circuit characterization is as mentioned in Section 3**.** We construct the three stage model of the PCEs and obtain V_{IL}-V_{IH} curves for the FCs. These curves are then used by the error-rate calculation equations described in Section 3.3 to produce the error-rates of PCEs. For PRCA, we obtain 4 unique PCEs which are characterized by the proposed procedure.

4.4. Results

Table 1 (end of the paper) compares the results obtained from simulation and prediction for various logic gates. As can be seen from the table the characterization results obtained from simulation and prediction are very close. The average relative deviation is around 6% for the logic gates. The characterization results of the unique PCEs of the PRCA are used to predict the error-rates of a 12-bit RCA by the cascade math model. The implementation of the cascade math model is done in MATLAB. The results from prediction and HSPICE simulations are shown for the 12-bit RCA in Fig. 19. The results obtained from prediction using the proposed method are labeled as Prediction and the HSPICE results are labeled as Simulation in the figure. As can be seen from Fig. 19, the predicted results from the proposed methodology are reasonably close to the simulated results. For our 12-bit RCA, the average relative deviation of the predicted results from HSPICE results is under 4.5%.

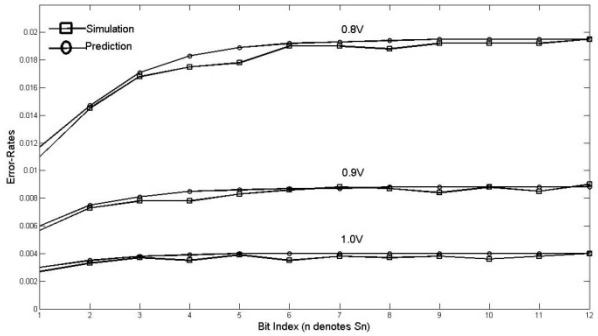

Figure 19: 12-bit PRCA Error-Rates

Table 2 compares the computation time for characterization of various logic gates using the previous approach of simulation and the proposed method. As tabulated, computation time for the proposed approach is very small as compared to the previous approach and grows less rapidly with the complexity of the three stage model as compared to the previous approach.

Table 2: Comparison of Computation Time

Approach / Gate	Previous (second)	Proposed (second)
Inverter	483	7.2
Nand	618	7.2
Nor	620	7.8
Carry	1080	8.1
Sum	1134	8.1

5. Conclusion

In this paper, we proposed a new method of characterizing noise based PCMOS circuits. We proposed new matrices for structural description of time domain noise and used V_{IL} and V_{IH} curves of a gate to obtain the error-rates of a PCE. We have showed that the new procedure is not only accurate as the previous approach but is also computationally efficient.

6. Acknowledgement

This work was supported by NTU-Rice Institute of Sustainable and Applied Infodynamics at Nanyang Technological University (Project Ref: M58B30001).

7. References

[1] K. V. Palem, “Energy aware computing through probabilistic switching: A study of limits,” IEEE Transactions on Computers, vol. 54, no. 9 pp. 1123-1137, 2005.

[2] J. George, B. Marr, B. E. S. Akgul and K. V. Palem, "Probabilistic arithmetic and energy efficient embedded signal processing," Proceedings of CASES, 2006, pp. 158-168.

[3] L. N. Chakrapani, K. K. Muntimadugu, A. Lingamneni, J. George and K. V. Palem, "Highly energy and performance efficient embedded computing through approximately correct arithmetic: A mathematical foundation and preliminary experimental validation," Proceedings of CASES, 2008, pp. 187-196.

[4] M. Lau, K. V. Ling, A. Bhanu, and V. J. Mooney III, "Error rate prediction for probabilistic circuits with more general structures", The 16th Workshop on Synthesis And System Integration of Mixed Information technologies (SASIMI2010), 18-19 October, 2010, Taipei, Taiwan, pp. 220-225.

[5] A. Singh, A. Basu, K. V. Ling and V. J. Mooney III, "Modeling multi-output filtering effects in PCMOS", VLSI-DAT 2011, Hsinchu, Taiwan (accepted)

[6] P. Korkmaz, B. E. S. Akgul, K. V. Palem and L. N. Chakrapani, "Advocating noise as an agent for ultra-low energy computing: probabilistic complementary metal-oxide-semiconductor devices and their characteristics," Japanese Journal of Applied Physics, vol.45, no.4, pp. 3307–3316, April 2006.

[7] P. Korkmaz, B. E. S. Akgul and K. V. Palem, "Energy, performance, and probability tradeoffs for energy-efficient probabilistic CMOS circuits," IEEE Transactions on Circuits and Systems I, vol. 55, no. 8, pp. 2249-2262, September 2008.

[8] J. George, B. Marr, B. E. S. Akgul and K. V. Palem, "Probabilistic arithmetic and energy efficient embedded signal processing," Proceedings of CASES, pp. 158-168, 2006.

[9] C. F. Hill, "Noise margin and noise immunity in logic circuits," Microelectronics, vol. 1, pp. 16-21, 1968.

[10] J. Lohstroh, "Static and dynamic noise margins of logic circuits," IEEE Journal of Solid-State Circuits, vol. SC-14, no. 3, pp. 591-598, 1979.

[11] J. M. Zurada, et al., "Dynamic noise margins of MOS logic gates," in Proc. Int. Symp. Circuits and Systems, pp. 1153-1156, 1989.

[12] Li Ding and Pinaki Mazumder, "Dynamic noise Margin: Definitions and Model," 17th International Conference on VLSI Design, pp.1001-1006, 2004.

[13] P. J. Fish, Electronic noise and low noise design, McGraw-Hill, 1993.

[14] G. Vasilescu, Electronic noise and interfering signals, Springer, 2005.

[15] A. Papoulis and S. U. Pillai, Probability random variables and stochastic processes, McGraw-Hill, 2001.

[16] S. H. Unger, The essence of logic circuits, Ed. Prentice-Hall International, Inc. 1989

Table 1: Simulation and Prediction Results for Various Logic Gates

	HSPICE Simulation						Prediction					
RMS	0.2 V			0.3 V			0.2 V			0.3 V		
VDD / Gates	0.8 V	0.9 V	1.0V	0.8 V	0.9 V	1.0 V	0.8 V	0.9 V	1.0 V	0.8 V	0.9 V	1.0 V
Inverter	0.0109	0.0052	0.0022	0.0647	0.0452	0.0305	0.0113	0.0059	0.0024	0.0662	0.0468	0.0317
NAND	0.0112	0.0054	0.0026	0.0648	0.0454	0.0309	0.0117	0.0062	0.0030	0.0671	0.0473	0.0321
NOR	0.0105	0.0052	0.0023	0.0639	0.0449	0.0300	0.0112	0.0056	0.0025	0.0658	0.0464	0.0311
Carry (cin-cout')	0.0065	0.0030	0.0013	0.0519	0.0356	0.0235	0.0069	0.0033	0.0014	0.0536	0.0369	0.0245
Sum (cin-sum')	0.0047	0.0021	0.0009	0.0445	0.0306	0.0202	0.0050	0.0023	0.0010	0.0457	0.0324	0.0210