A More Precise Model of Noise Based CMOS Errors

Arun Bhanu[§], Mark S. K. Lau[*§], Keck-Voon Ling[*§], Vincent J. Mooney III[*†§], Anshul Singh[*‡§]

{arunbhanu, marklausk, ekvling, vjmooney}@ntu.edu.sg, anshul_singh@students.iiit.ac.in

***Abstract*—In this paper we present a new model for characterization of probabilistic gates. While still not mainstream, probabilistic CMOS has the potential to dramatically reduce energy consumption by trading off with error rates on individual bits, e.g., least significant bits of an adder. Our contribution helps account for the filtering effect seen in noise based PCMOS in a novel way. The characterization proposed here can enable accurate multi-bit models based on fast mathematical extrapolation instead of expensive and slow HSPICE simulations.**

I. Introduction

In 2006 George et al. proposed a unique and novel approach to low power computation for predicted future technology generations where determinism is not guaranteed [1]. The approach – Biased Voltage Scaling or BiVoS for short – proposes using higher voltage supply (VDD) for high order bits and lower VDD for low order bits. This bias in the supply voltage was found to save significant (e.g., 5X or more) amounts of computation energy with minimal impact on signal to noise ratio (SNR) in a synthetic aperture radar (SAR) image. However, the authors simulation methodology was to use SPICE to characterize one-bit full adders and then C-based simulations to extrapolate to multi-bit Ripple-Carry Adders (RCAs) [1]. As the first paper in this topic, such an approach was reasonable, but one obvious additional step is to run SPICE simulations of multi-bit RCAs with the proposed BiVoS approach and compare with what the faster C-based simulations said would happen. This is precisely what we have investigated and will present in this paper.

II. Prior Work

In 2002 Kish predicted that frequency and voltage scaling would soon end as we know it [2], a prediction which has roughly turned out correct as microprocessor frequency scaling has in fact slowed dramatically, ushering in the current focus on multi-core processors. The main culprit cited by Kish is noise (thermal, shot and flicker) which creates unwanted disturbances that interfere with the desired logic signals on a semiconductor chip. Around the same time, Palem proposed a new approach to computing with the potential to reduce energy significantly (in a nonlinear, perhaps exponential fashion) when traded off with probability of correctness [3] [4]. This "probabilistic computing" approach requires a realignment of the fundamental logic to no longer be deterministic (e.g., deterministic Boolean logic) but probabilistic (e.g., probabilistic Boolean logic [5]).

A. Noise as a Source of Errors and a Probabilistic Switch

To implement probabilistic Boolean logic, a probabilistic switch produces a desired value as an output that is 0 or 1 with probability p, and, hence, can produce the wrong output value with a probability $(1-p)$ [3] [4]. Studies of a noise induced Probabilistic CMOS (PCMOS) inverter as a probabilistic digital switch were carried out by Korkmaz et al. [6] introducing an analytical model which relates probability of correctness of a switch to supply voltage of the switch.

*School of EEE, Nanyang Technological University(NTU), Singapore
†School of ECE, Georgia Institute of Technology, Georgia, USA
‡International Institute of Information Technology, Hyderabad, India
§Institute for Sustainable Nanoelectronics, NTU, Singapore

B. Modeling a Noisy Circuit

This section describes the methodology followed by George et al. [1] and Korkmaz et al. [6] for coupling noise at the output of a circuit. In [1] and [6] a noisy circuit has been modelled by adding an equivalent noise source at the output of a normal circuit in a known process technology. This approach assumes that the equivalent noise source at the output estimates the impact of the noise present in each of the gates or transistors in an actual noisy circuit. Fig. 1 shows how a noise source can be added to the output of a full adder (FA) circuit

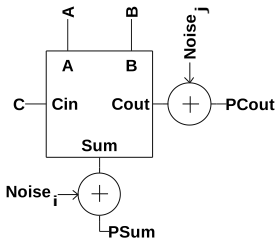

Fig. 1. Prior characterization of noised based error

to make a probabilistic full adder (PFA) circuit. In Fig. 1, a "P" prefix denotes a probabilistic output, and the noise sources Noise_i and Noise_j are independent of each other. A sampled output would have the logical state zero (one) if the output voltage is lower (equal to or higher) than half of the supply voltage [1] [6]. To determine if an error has occurred, the ideal output logic value is compared to the noisy output logic value. If the two values do not match then an error has occurred. The error probability is calculated by counting the total number of errors and dividing by the total number of input samples. It is believed that this noise model can mimic noise inherent in transistors of future technologies [2].

C. HSPICE Simulation of a Full Adder

HSPICE simulations in this paper use TSMC 180nm technology and Synopsys 90nm generic library. For the 180nm TSMC technology, the nominal voltage is 1.8 V. For Synopsys 90nm generic library, the nominal voltage is 1.2 V. All FAs used in design of the PFAs are designed using a 24-transistor mirror adder circuit [7].

Following [1], we construct a PFA by coupling noise sources to the output terminals of the carry-out and sum of a (deterministic) FA. The noise sources are independent of each other and are independent of the inputs and outputs of the PFA. The noise sources are of Gaussian distribution with zero mean and standard deviation σ. We simulate a noise source in HSPICE by use of a voltage-controlled voltage source (VCVS) with a voltage gain equal to the standard deviation σ, or root-mean-square (RMS), of the noise. Random numbers of the standard Gaussian distribution are generated by Matlab. The VCVS multiplies these random numbers by its voltage gain to obtain a Gaussian noise source of the RMS value σ. The noise samples are added to the output terminals every nano-second. Here, the noise RMS is set to 0.3 V for TSMC 180nm technology which is same as was done in [1]. In order to keep the same ratio of VDD to noise RMS, for Synopsys 90nm

technology the noise RMS is set to 0.2 V. The probabilities of carry and sum bits are now obtained by HSPICE experiments.

While we will present a set of results later, let us here consider the case of a supply voltage of 0.8 V which results in some error rate given our noise with RMS 0.3 V for TSMC 180nm and 0.2 V for Synopsys 90nm. We consider fifty thousand input vectors of $(\mathsf{A}, \mathsf{B}, \mathsf{C})$ randomly generated by Matlab. The results of HSPICE simulation of the PFA of Fig. 1 is shown in Table I.

TABLE I
ERROR PROBABILITIES OF A FULL ADDER

$Technology$	$Voltage$	$PCout$	$PSum$
180nm	0.8	0.0896	0.0933
90nm	0.8	0.0236	0.0231

III. SIMULATION OF MULTI-BIT RCAS

Our first step was to extrapolate the behavior of a multi-bit RCA from the single bit (one FA). We used a closed-form mathematical formula [8]. (We leave the closed form mathematics as a separate topic beyond the scope of this paper, but it is explained in more detail in [8].) We were able to reproduce the results of prior work [1].

A. Predicting Multi-Bit Errors

We used the mathematical model presented in [8] to predict the error probabilities of sum bits of a 4-bit PRCA using the results obtained from HSPICE simulation of Fig. 1. The results obtained for a VDD of 0.8 V are shown in Table II.

TABLE II
ERROR PROBABILITIES DETERMINED USING PRIOR MODEL [1]

$Technology$	$Voltage$	$PSum0$	$PSum1$	$PSum2$	$PSum3$
180nm	0.8	0.0933	0.1662	0.1961	0.2084
90nm	0.8	0.0231	0.0343	0.0365	0.0419

B. HSPICE Simulation of Multi-Bit Errors

A probabilistic ripple carry adder (PRCA) consists of chain of probabilistic full adders (PFA). Fig. 2 shows this simulation setup.

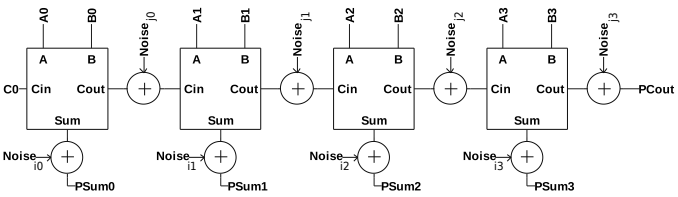

Fig. 2. Simulation setup of 4-bit PRCA using prior model

The supply voltage we use for each PFA is v volts for some $v \in \{0.8, 0.9, \ldots, 1.8\}$ for TSMC 180nm and $v \in \{0.8, 0.9, \ldots, 1.2\}$ for Synopsys 90nm technology. In this paper, all PFAs in a PRCA receive the same supply voltage. Fifty thousand 4-bit vectors of A and of B are randomly generated by Matlab. Each vector entry is a binary number of uniform distribution. The sampling period is 20 nano-seconds. That is, every 20 nano-seconds, a newly generated input vector (A, B) is input to the four-bit PRCA and outputs of the PRCA (which are due to the past input vectors) are sampled. A sampled output bit would have a logical state of zero (one) if the output voltage is lower (equal to or higher) than half of the supply voltage [1] [6]. We compare the sampled outputs of the PRCA and with the ideal expected logic output value. If they have different values, then we declare that an error has occurred. The experiments are repeated with different supply voltages $v \in \{0.8, 0.9, \ldots, 1.8\}$ for TSMC 180nm and $v \in \{0.8, 0.9, \ldots, 1.2\}$ for Synopsys 90nm.

The error probability of sum and carry bits of PFA is used to characterize the probabilistic behaviour of a 4-bit PRCA through HSPICE simulations. From our HSPICE simulation of a 4-bit PRCA we have found that the model of Fig. 1 overpredicts the error probability. The results obtained using HSPICE simulation of prior model [1] for 0.8 V VDD are shown in Table III. Note that, for example, the actual error rate of $PSum3$ in 180nm technology is 10.72% and not 20.84% as predicted in Table III.

TABLE III
ERROR PROBABILITIES DETERMINED USING HSPICE SIMULATION OF PRIOR MODEL [1]

$Technology$	$Voltage$	$PSum0$	$PSum1$	$PSum2$	$PSum3$
180nm	0.8	0.0908	0.0984	0.1053	0.1072
90nm	0.8	0.0231	0.0456	0.0563	0.0614

C. Findings

In summary, we found that while the single bit error rates were valid given the assumptions described in Section II for a single bit scenario, the extrapolation to multiple bits for an RCA resulted in over predicting the error. The main reason for this appears to be the so-called "noise filtering effect" discussed in Section VI.B of [9]. In short, noise filtering occurs when the duration of the noise pulse is shorter than the propagation delay of a gate [9]. We will propose in the next section a way to deal with noise filtering effects in a practical way.

IV. NEW MODEL FOR NOISE-BASED ERROR CHARACTERIZATION OF CMOS GATES

In our HSPICE simulations of a 4-bit PRCA we have found that the model shown in Fig. 1 overpredicts the error probabilities. So, we propose a new model which takes into account the filtering effect. Fig. 3 shows our newly proposed model.

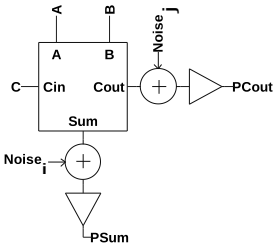

Fig. 3. Proposed characterization of noised based error

TABLE IV
ERROR PROBABILITIES OF A FULL ADDER USING PROPOSED MODEL

$Technology$	$Voltage$	$PCout$	$PSum$
180nm	0.8	0.005440	0.039600
90nm	0.8	0.008940	0.021340

In Fig. 1, samples are taken immediately after the carry and sum outputs, whereas in Fig. 3, samples are taken after buffers attached to the sum and carry outputs. The buffer attached to the carry output has the same worst-case delay as the FA to be used in the target RCA. The buffer attached to a sum output has the same worst-case delay as the master latch of the master-slave flip-flop to be used in our implementation. In this setup, the buffers model the actual loads that will be connected to PFAs when they are chained to form a PRCA. In a PRCA, a noisy carry-out signal often becomes less noisy after propagating into the next PFA. As a result, any errors appearing at the carry-out may not be "seen" as an error by the following PFA. It is hoped that the filtering effect of the buffer would be similar to that

of a PFA. Thus, by taking samples after the buffer, we can estimate the statistics of the errors "seen" by the PFA. HSPICE simulation of the new proposed PFA model is carried out using a similar procedure as described in Section II-C. The results of HSPICE simulation of our new proposed PFA model is shown in Table IV.

V. Simulation of Multi-Bit RCA Using the New Model

A. Predicting Multi-Bit Errors

We used the mathematical model presented in [8] to predict the error probabilities of sum bits of a 4-bit PRCA using the results obtained from HSPICE simulation of Fig. 3. The results obtained for a supply voltage of 0.8 V are shown in Table V.

TABLE V
Error probabilities determined using proposed model

$Technology$	$Voltage$	$PSum0$	$PSum1$	$PSum2$	$PSum3$
180nm	0.8	0.0396	0.0446	0.0471	0.0483
90nm	0.8	0.0210	0.0296	0.0314	0.0360

B. HSPICE Simulation of Multi-Bit Errors

Fig. 4 shows the simulation setup of the 4-bit PRCA simulation setup using the proposed new model. Note the addition of extra

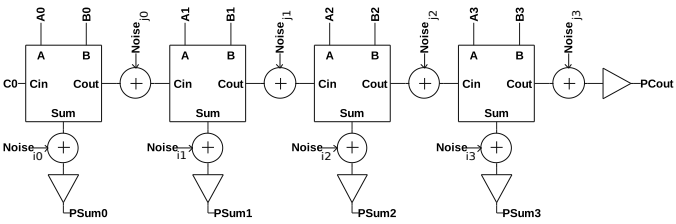

Fig. 4. Simulation setup of 4-bit PRCA using proposed model

TABLE VI
Error probabilities determined using HSPICE simulation of proposed model

$Technology$	$Voltage$	$PSum0$	$PSum1$	$PSum2$	$PSum3$
180nm	0.8	0.0365	0.0432	0.0464	0.0530
90nm	0.8	0.0213	0.0299	0.0341	0.0362

buffers to the sum and carry bits. The buffers attached to the sum bits and the final carry bit (the most significant bit of the result) have the same delay as typical flip-flop. The buffers act as the noise filters as explained in Section IV. The rest of the simulation setup remains the same as described in Section III. The results obtained using HSPICE simulation of proposed model for 0.8 V supply voltage are shown in Table VI.

VI. Discussion and Observations

The error probability results obtained using the HSPICE simulation of the setup described in [1], the results obtained using mathematical model in [8] and HSPICE simulation of proposed new model are shown in in Table VII (TSMC 180nm, note that odd supply voltage values such as 0.9 V are not shown) and Table VIII (Synopsys 90nm). Clearly, cases 3 and 4 in Table VII and Table VIII closely match, while cases 1 and 2 diverge widely. We would like to comment that we tried a number of alternatives in addition to what we present here; for example, we matched the output load capacitance (instead of delay), but the results were nowhere near as good/close. Also, the HSPICE simulation with output buffers in Fig. 4 is clearly more realistic than Fig. 2, and so we conclude that use of output buffers is the case we find to be the best approach to modeling.

In short, we have a way to model noise based error which potentially may be seen in future technology nodes. None of the prior

TABLE VII
TSMC 180nm results.Case 1: Prediction for a PRCA without buffers by the model in [1]; Case 2: Simulation of a PRCA without buffers; Case 3: Prediction for a PRCA with buffers by the proposed model; Case 4: Simulation of a PRCA with buffers.

Voltage	Case	$PSum0$	$PSum1$	$PSum2$	$PSum3$
0.8	1	0.0908	0.0984	0.1053	0.1072
0.8	2	0.0933	0.1662	0.1961	0.2084
0.8	3	0.0365	0.0432	0.0464	0.0530
0.8	4	0.0396	0.0446	0.0471	0.0483
1.0	1	0.0469	0.0541	0.0553	0.0604
1.0	2	0.0484	0.0912	0.1106	0.1194
1.0	3	0.0267	0.0316	0.0325	0.0371
1.0	4	0.0292	0.0336	0.0357	0.0368
1.2	1	0.0222	0.0262	0.0275	0.0282
1.2	2	0.0217	0.0430	0.0531	0.0580
1.2	3	0.0168	0.0194	0.0193	0.0200
1.2	4	0.0172	0.0198	0.0211	0.0217
1.4	1	0.0096	0.0114	0.0122	0.0128
1.4	2	0.0099	0.0189	0.0234	0.0256
1.4	3	0.0084	0.0095	0.0087	0.0092
1.4	4	0.0079	0.0087	0.0091	0.0093
1.6	1	0.0042	0.0048	0.0044	0.0052
1.6	2	0.0043	0.0077	0.0094	0.0102
1.6	3	0.0038	0.0043	0.0043	0.0036
1.6	4	0.0038	0.0040	0.0041	0.0041
1.8	1	0.0015	0.0017	0.0019	0.0022
1.8	2	0.0014	0.0026	0.0032	0.0035
1.8	3	0.0014	0.0017	0.0021	0.0012
1.8	4	0.0013	0.0014	0.0014	0.0014

TABLE VIII
Synopsys 90nm results.Case 1: Prediction for a PRCA without buffers by the model in [1]; Case 2: Simulation of a PRCA without buffers; Case 3: Prediction for a PRCA with buffers by the proposed model; Case 4: Simulation of a PRCA with buffers.

Voltage	Case	$PSum0$	$PSum1$	$PSum2$	$PSum3$
0.8	1	0.0231	0.0343	0.0365	0.0419
0.8	2	0.0231	0.0456	0.0563	0.0614
0.8	3	0.0210	0.0296	0.0314	0.0360
0.8	4	0.0213	0.0299	0.0341	0.0362
0.9	1	0.0127	0.0187	0.0191	0.0241
0.9	2	0.0127	0.0253	0.0315	0.0345
0.9	3	0.0114	0.0158	0.0165	0.0201
0.9	4	0.0116	0.0160	0.0182	0.0193
1.0	1	0.0065	0.0094	0.0102	0.0118
1.0	2	0.0065	0.0131	0.0163	0.0179
1.0	3	0.0055	0.0081	0.0086	0.0102
1.0	4	0.0056	0.0076	0.0086	0.0090
1.1	1	0.0028	0.0045	0.0050	0.0059
1.1	2	0.0028	0.0060	0.0075	0.0083
1.1	3	0.0025	0.0039	0.0042	0.0050
1.1	4	0.0025	0.0033	0.0037	0.0040
1.2	1	0.0013	0.0021	0.0025	0.0029
1.2	2	0.0013	0.0026	0.0033	0.0036
1.2	3	0.0011	0.0017	0.0019	0.0023
1.2	4	0.0011	0.0015	0.0017	0.0017

papers [3], [4], [1], [9] figured out what we have presented in this paper. The cases shown so far are limited for the sake of illustration. A more complete set of results, which shows the predicted and simulated error probabilities (left plots) and their relative percentage errors (right plots) for TSMC 180nm and Synopsys 90nm technology is shown in Fig. 5 and Fig. 6 respectively.

We see from left plots of Fig. 5 (a) and Fig. 6 (a) that error probabilities of the prior model [1] diverge significantly from theoretical error probabilities as determined by the math model [8] where as Fig. 5 (b) and Fig. 6 (b) shows that the proposed new model follows the theoretical error probability values very closely. The right plot shows that the relative percentage error between the prior model [1] and the proposed new model. The relative error is less than 15% in

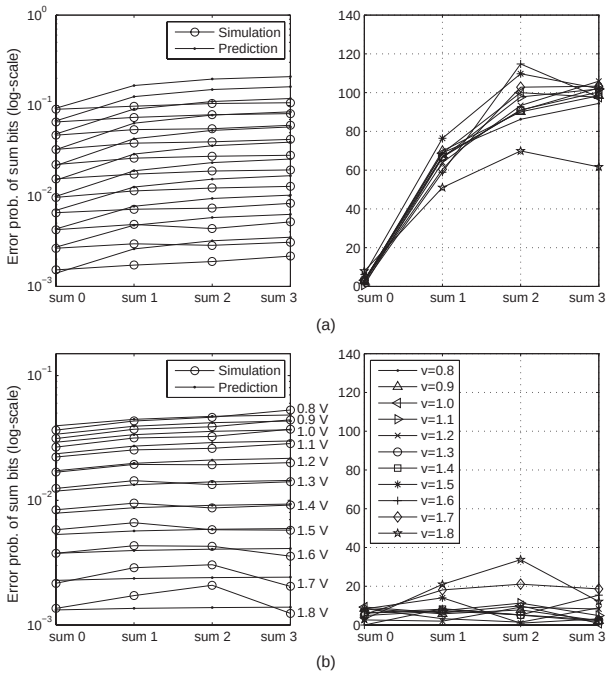

Fig. 5. Error probabilities and relative error for TSMC 180nm technology. The top two plots (a) show results for the prior model: sum bit error probabilities (top left) and relative error of the prior model versus HSPICE results (top right). The bottom two plots (b) show results for the model proposed in this paper: sum bit error probabilities (bottom left) and relative error (bottom right) for each sum bit when compared to HSPICE results. Note that in the left plots the simulated and predicted probabilities using the math model [8] are marked by circles and dots, respectively going from a VDD of 0.8 V to 1.8 V (shown explicitly in bottom left plot but not in the top left plot).

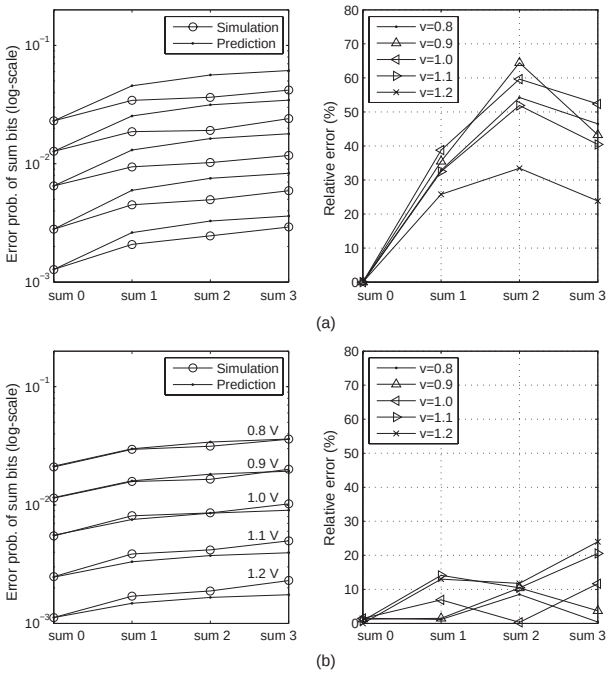

Fig. 6. Error probabilities and relative error for Synopsys 90nm technology. The top two plots (a) show results for the prior model: sum bit error probabilities (top left) and relative error of the prior model versus HSPICE results (top right). The bottom two plots (b) show results for the model proposed in this paper: sum bit error probabilities (bottom left) and relative error (bottom right) for each sum bit when compared to HSPICE results. Note that in the left plots the simulated and predicted probabilities using the math model [8] are marked by circles and dots, respectively going from a VDD of 0.8 V to 1.2 V.

most cases for the new proposed model for TSMC 180nm technology, where as the relative error of the prior model [1] is above 60% most of the time. Note that the case of sum bit 2 for the new model with VDD of 1.8 V is a statistical anomaly due to a tiny error rate - 104 errors out of 50000 samples.

Similar results can be observed for Synopsys 90nm shown in Fig. 6. The relative error is less than 15% in most cases for the new proposed model for Synopsys 90nm technology, where as the relative error of the prior model [1] is above 50% most of the time. We also have experiments for 8-bit PRCAs which confirm the same results; due to lack of space, plots of these results are not included.

VII. Conclusion

We have presented a much more accurate way to model probabilistic CMOS (PCMOS) where noise is the dominant cause of the probabilistic behavior. We have found that to properly account for the filtering effect, we can simply add buffers with worst-case delays equal to those expected in the final target VLSI implementation. Using this approach, an error model at the gate level can be used to accurately predict errors at the level of a multi-bit adder. Such predictive power is critical in high-level VLSI design space exploration where, for example, may different adder logic structures and topologies may be considered. This paper is the first to present such an accurate modeling approach which closely match HSPICE.

Acknowledgment

This work was supported by the Institute for Sustainable Nanoelectronics, Nanyang Technological University (Project Ref: M58B30001).

References

[1] J. George, B. Marr, B. E. S. Akgul, and K. V. Palem, "Probabilistic arithmetic and energy efficient embedded signal processing," in *Proceedings of the 2006 International Conference on Compilers, Architecture and Synthesis for Embedded Systems*, October 2006, pp. 158–168.

[2] L. B. Kish, "End of Moore's law: Thermal (noise) death of integration in micro and nano electronics," *Physics Letters A*, vol. 305, no. 3-4, pp. 144–149, 2002.

[3] K. V. Palem, "Energy aware algorithm design via probabilistic computing: From algorithms and models to moore's law and novel (semiconductor)devices," in *Proceedings of the International Conference on Compilers, Architecture and Synthesis for Embedded Systems (CASES'03)*, October 2003, pp. 113–116.

[4] ——, "Energy aware computing through probabilistic switching: A study of limits," *IEEE Transactions on Computers*, vol. 54, no. 9, pp. 1123–1137, September 2005.

[5] L. N. B. Chakrapani and K. V. Palem, "A probabilistic boolean logic and its meaning," Rice University, Department of Computer Science Technical Report, No. TR08-05, Tech. Rep., June 2008.

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

[7] J. M. Rabaey, A. Chandrakasan, and B. Nikolić, *Digital Integrated Circuits: A Design Perspective*, 3rd ed. Prentice Hall, 2003.

[8] M. S. K. Lau, K. V. Ling, Y. C. Chu, and A. Bhanu, "Modeling of probabilistic ripple-carry adders," in *Proceedings of the 5th IEEE International Symposium on Electronic Design, Test & Applications (DELTA)*, January 2010.

[9] 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.