Propagation of Delay in Probabilistic CMOS Systems

L. Oudshoorn, G.A. Gillani, A.B.J. Kokkeler
Faculty of Electrical Engineering Mathematics and Computer Science, University of Twente, Enschede, Netherlands
Email: luuk.oudshoorn@gmail.com; {s.ghayoor.gillani, a.b.j.kokkeler}@utwente.nl

***Abstract*—Future low voltage noise dominated designs render probabilistic behavior of CMOS. This is acceptable as far as applications' intrinsic error resilience allows quantified inaccuracy in results to save energy consumption, such as in applications like audio/video processing and sky image formation in radio astronomy. This introduces the trade-off between energy consumption (E) and probability of correctness (p) that provides an opportunity for inexact computing to attain higher energy efficiency. Efforts have been made in the last decade to model probabilistic CMOS (PCMOS) keeping in view the noise variance and to establish its feasibility for error resilient applications focused on the nominal voltage range. However, exploiting the near threshold voltage (NTV) range is quite a promising energy efficient design technique that operates the hardware at relatively slower pace while retaining the deterministic property of computations. We propose to take the advantage of energy efficiency at NTV while retaining the speed as constant, sacrificing p to the extent allowed by applications resilience. In this regard, we investigated the impact of NTV operation on PCMOS where more energy can be saved with less accurate results. Our simulation results of an inverter and a 4-bit ripple carry adder in Cadence showed the shortcomings of current analytical models for probability of correctness at NTV and lower voltage supplies. We further investigated the impact of delay propagation in a digital system composed of probabilistic building blocks, which provides a clear insight of timing delay affecting the higher significant computational bits more than its lower significant counterparts and hence contributing considerably to the total error.**

I. Introduction

Energy efficient computing has been motivated by the increased trend in power dissipation of microprocessor technology due to higher complexity in reduced transistor dimensions [1]. Moreover, the gigantic power requirements for data intensive scientific computations such as digital processing in radio astronomy also urge for low power designs where performance requirements exceed hundreds of PFLOPS [2].

The quadratic relation of supply voltage (V_{dd}) with power consumption substantiates the viability for low voltage design to save energy. Different levels of low voltage designs have been explored in literature; namely: ultra low or sub-threshold voltage [3], near threshold voltage (NTV) [4], and super-threshold or nominal voltage range. Energy efficiency decreases and performance increases as we go from ultra low voltage to nominal voltage approach. Moreover, dynamic voltage and frequency scaling (DVFS) techniques [5] and heterogeneous architectures [6] have also been explored to achieve better energy-delay product. The design methodology for low voltage design is to reduce the V_{dd} to a minimum viable level (V_{opt}) and compensate the performance loss by exploiting parallelism within the applications while running the parallel threads on additional cores. V_{opt} is determined by the technology size, frequency of operation and intrinsic characteristics of software application like architectural and Amdahl's overhead [7]. Recent analysis on low-voltage designs [8] suggest that the NTV operation is a promising technique to attain higher energy benefits with a reasonable performance for error-free computing.

Inexact computing—where energy can be saved further by operating the deterministic hardware for less iterations/ computational bits, or by operating non-deterministic hardware with less probability of correctness—has attracted researchers over the last decade to trade-off accuracy of results for increased energy gains. Adaptive voltage over-scaling without error correction is another kind of inexact computing that saves power more than its error-free low voltage design equivalent [9]. Inexact computing is used for error resilient applications where the input data is redundant or noisy, or the algorithms are statistical/probabilistic in nature or with self-healing iterations, or the observer of outputs has perceptual limitations [10]; audio/video processing, digital communication and sky image formation in radio astronomy are few of such examples.

Palem in [11] and [12] showed that there is fundamentally a lower limit of the energy usage of circuits that work probabilistic instead of deterministic and proposes to use PCMOS as an alternative to its deterministic counterpart. [13] has explained the tradeoffs between performance, energy and inexactness in probabilistic circuits. The PCMOS circuits are modeled by introducing noise as a metric to incorporate the desired probability of correctness and it is assumed that the E-p curve can be described by an error function [14]. Recent work in [15] compared the addition of noise at the input, output and power supply of an inverter circuit and also proposed an energy efficient way of PCMOS system design catering the propagation of error by providing higher voltages to higher significant computational bits.

Based on the fact that modern digital design targets NTV region for optimal supply voltage, we are specifically interested in the energy consumption vs probability of correctness relation, i.e. E-p curves, within the NTV circuit operation. We have investigated the impact of variations in transistor size, frequency of operation, and noise levels on the E-p curves and found a different behavior between the analytical models and our simulation results in the NTV and lower levels. This is due to the drastic increase in channel impedance at these voltage levels which brings slow behavior of the circuits. Moreover, we are presenting the effect of delay propagation on the digital systems composed of probabilistic blocks which is quite more severe than when only considering the noise propagation.

Rest of the paper is organized as follows. Section II briefly dis-

cusses the theory and analytical models of PCMOS. In section III, we discuss the inverter simulation setup, assumptions, and the results for various frequencies, noise levels and transistor widths. Section IV presents the delay propagation in PCMOS systems with simulation results of a ripple carry adder as an example and section V concludes our work.

II. Theory of Probabilistic CMOS

To understand why probabilistic behavior can lead to fundamentally lower energy usage, the process of switching can be analayzed from a thermodynamic perspective. Palem [11] has shown the energy gain of PCMOS as kTln(1/p); where p is the probability of correctness, T is the temperature and k is the Boltzmann constant. Analytical discussions in [14] and [16] suggest to model a probabilistic switch, for instance an inverter, with a noise coupled output as,

$$p = P(X \leq \frac{V_{dd}}{2}) = \frac{1}{2} + \frac{1}{2} erf(\frac{V_{dd}}{2\sqrt{2}\sigma}) \tag{1}$$

where σ is the noise RMS. We will compare Eq. 1 with our simulation results in section III to discuss the relation in behavior of E-p curves at NTV and lower voltages. Moreover, it is important to calculate the error propagation within the probabilistic system to find the total error which is bounded by the applications intrinsic resilience. In case of a probabilistic ripple carry adder, the probability of correctness of the sum output of stage $i+1$ depends on the probability of correctness of the adder block itself and also on the probability that its input carry from stage i is correct. Keeping in view this effect, M. Lau [17] calculated the propagation error within a 4-bit ripple carry adder for each sum and carry output. We further derive for the probability of correctness of sum outputs, i.e. the probability that the probabilistically calculated sum ($s^{'}$) at any stage (i) equals its deterministic counterpart (s) as,

$$P(s^{'}_{i+1} = s_{i+1}) = \frac{1}{2} + (p^s_{i+1} - \frac{1}{2}) \times [\prod_{j=1}^{i}(p^c_j - \frac{1}{2}) + \sum_{k=1}^{i}\prod_{l=k}^{i}(p^c_l - \frac{1}{2})] \tag{2}$$

In section IV, we will show from our simulation results, the impact of delay propagation in addition to error propagation as modeled in Eq. 2 and compare them to emphasize the importance of considering delay propagation in the PCMOS system design.

III. Simulation Setup and Resulting E-p Curves

In the previous section a theoretical basis for probabilistic CMOS has been established. In order to get a more realistic view on the practical use of PCMOS, simulation results for E-p curves are presented in this section. The CMOS inverter is chosen as the circuit to be simulated for the sake of simplicity. In the next section, this knowledge will be used for analyzing the more complex circuits, for instance, a 4-bit ripple carry adder.

TABLE I.
Simulation Parameters

Parameter	Value
MOS type	Umc65ll N/P_12_llrvt
NMOS gate length	60nm
NMOS gate width	80nm
PMOS gate length	60nm
PMOS gate width	160nm
V_{dd}	Range: 0-2V
V_{in}	Alternating between 0V and V_{dd}
C_{out}	10fF
Temperature	27°C
Noise amplification	50x/100x/200x/400x
Bit duration	1ns
# Bits simulated	800

A. Simulation Setup

We have used the umc65 library in Cadence IC for the simulations. Our approach is to simulate 65nm technology with increased intrinsic noise due to channel resistance to represent the much smaller future transistors. A parameter called 'noise scale' is used to amplify the noise levels. The points on the E-p curves are estimated by simulating many bit periods for a supply voltage setting and counting the number of correct and incorrect samples. A long transient simulation is performed using Cadence, which can simulate time domain noise. The results are then exported to Matlab where the sampling and processing of the data is done.
The simulation in Cadence is performed using a default set of parameters with variably increased noise amplification scales as shown in Table I. In order to get all the points for the E-p curve, a parameter sweep is performed for the V_{dd} parameter. To keep the simulation time acceptable, the number of points (the supply voltage step size) for the E-p curves is kept relatively low. Although this results in a less smooth curve, it gives a reasonable representation of facts.

B. Inverter E-p Curves

The CMOS inverter circuit has been simulated to show the variations in E-p curves at low voltage levels. In our simulations, the noise can be scaled by a factor, which multiplies all generated noise by the chosen amount. Simulated E-p curves for noise scaled by a factor of 50 to 400 times are shown in Fig. 1. Though noise scaling factors of over 100 are not realistic for the contemporary CMOS feature sizes, these numbers are chosen in order to better show the qualitative influence of noise on the E-p curves. The dotted line in Fig.1 represents the theoretical performance for a certain noise standard deviation (here we assume 100mV) according to Eq.1. A large difference between the predicted shape and the simulation results is the sudden drop around the threshold voltage (0.5V) while lowering the supply voltage. This suggests that the inverter makes errors that are caused by malfunctioning rather than output misinterpretation due to noise. This is because the channel conductance of the inverter quickly becomes lower at low voltages, causing the output capacitance to charge (or discharge) slower. At a certain point the supply voltage becomes too low such that the output capacitance is not charged before the sample moment, resulting in abrupt decrease in probability of correctness.
Simulations with various operating frequencies also demonstrate the delay of the circuit. Interestingly, the model used

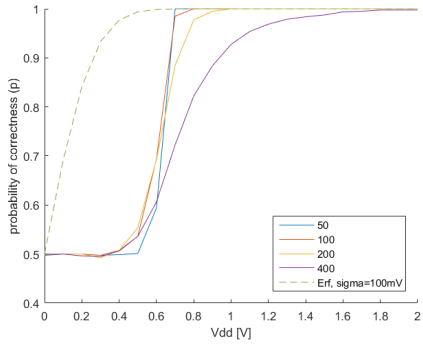


Fig. 1. CMOS inverter simulations for various noise scales

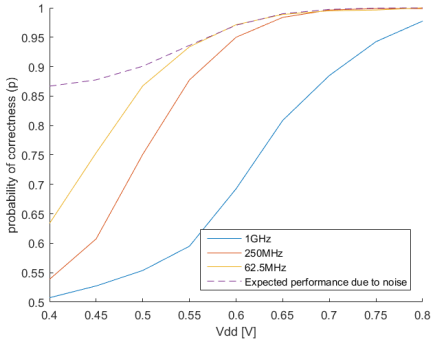


Fig. 2. CMOS inverter simulations for various operating frequencies

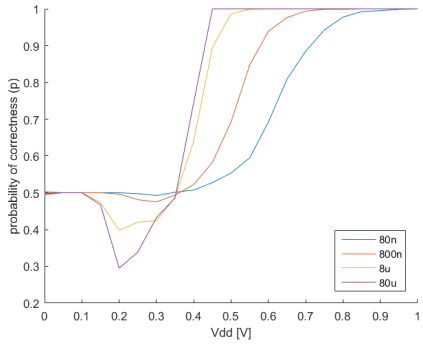
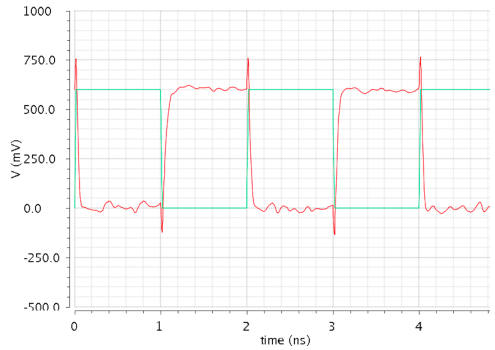


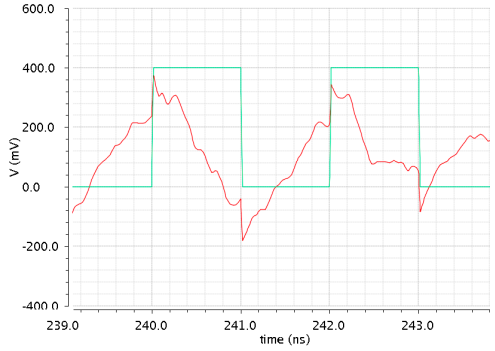
Fig. 3. CMOS inverter simulations for various gate widths

in Eq. 1 proves to be the theoretical maximum for the energy probability tradeoff, shown as dotted line in Fig. 2. To plot the dotted line, the standard deviation (σ) of the samples has been calculated for each supply voltage setting, which is assumed to be caused completely by the noise in the system. Whereas, we assumed it to be fixed at 100mV in Fig. 1. The theoretical maximum performance of a system in the presence of the measured amount of noise is calculated by filling in the supply voltage and noise standard deviation in the model specified by Eq. 1.

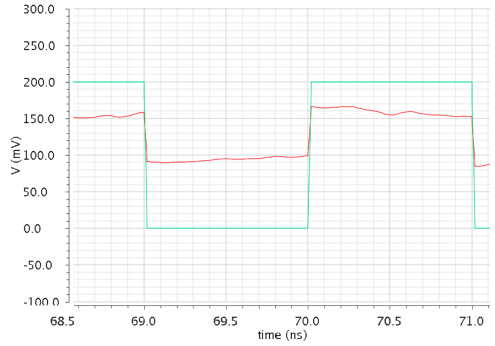
We have also investigated the influence of various gate widths at a constant noise factor of 200. Fig. 3 shows the simulation results for below 1V supply voltages to focus on the NTV region. As can be expected, the performance for a wider MOSFET is similar to that of a lower noise level. However, in case of wide MOSFETS at 0.35V or lower supplies, the probability of correctness drops below 0.5. This should not occur in theory, since noise should not make performance worse than guessing a bit. Going below 0.5 means that the device is actively malfunctioning. Closer inspection reveals that this is indeed the case. Parts of the transients of the 80u wide MOSFET are shown in Fig. 4 that depicts correct operation at 0.6V, a mostly correct behavior (p=0.8) at 0.4V and a mostly wrong behavior (p=0.3) at 0.2V. When the input signal changes from 0V to V_{dd}, the output signal which was at V_{dd} is momentarily pushed above V_{dd} due to the capacitive coupling between input and output. This is an extra push in the wrong direction, which has to be compensated by the inverter. In the 0.4V plot, the compensation for the extra push already takes up roughly one third of the available time. In the 0.2V plot the inverter is not even halfway compensating the extra push at the sampling point. This changes the behavior from inverting the signal to passing it, which gives probabilities of correct behavior below 0.5. Only a low frequency that pushes the entire signal to 0V or V_{dd} increases the odds again.



(a) Correct behavior at V_{dd}=0.6V



(b) Mostly correct behavior at V_{dd}=0.4V



(c) Mostly wrong behavior at V_{dd}=0.2V

Fig. 4. Transients of the 80u wide MOSFET at various operating voltages

IV. Delay Propagation in PCMOS Systems

We further investigated the influences that connected probabilistic building blocks have on each other. We simulated the 4-bit ripple carry adder comprised of 4 full adders in Cadence IC with the same assumptions as that of the inverter. Fig. 5 and Fig. 6 present the simulated E-p curves for the carry and sum outputs respectively for the 4-bit ripple carry adder along with calculated ones according to Eq. 2. Theoretically, the outputs of stages 2, 3 and 4 are expected to be almost equal to that of stage 1, but in our simulations they are worse. The theoretical curves are based on the assumption that the propagation of error is only due to probability of correctness metric. However, the delayed correct outputs of stage i can make the probability of correctness worse for stage $i+1$ than calculated by Eq. 2. Therefore, a logical explanation for the theory and simulation not being equal is the delay propagation. At the start of a clock, the calculations are started using whatever value is present at the input from the previous calculation. There is a 50% chance that the next input will be different. However, the new value will not be available immediately. Therefore, the calculation may be underway when a new value settles on an input. The calculation of the output then starts again, but with less time left to complete it before the clock cycle ends. Unfortunately, this problem stacks for additional stages resulting in the most significant sum output to fail first.

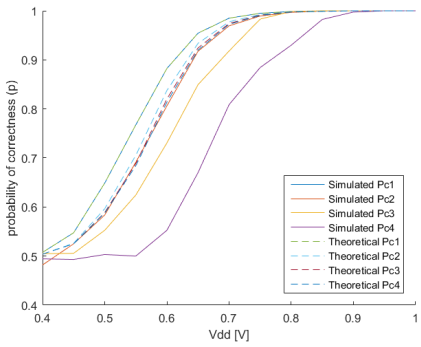


Fig. 5. Simulated and theoretical E-p curves for carry outputs

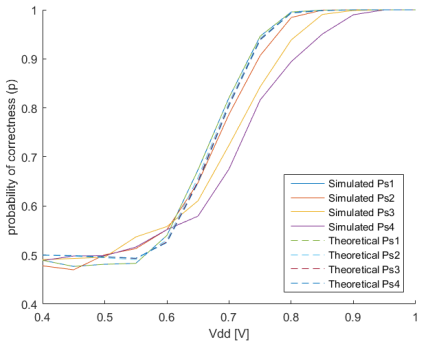


Fig. 6. Simulated and theoretical E-p curves for sum outputs

V. Conclusion

While modern energy efficient low voltage designs focus on near threshold voltage operation for general purpose computing and exploiting an application's intrinsic error resilience by deploying PCMOS circuits for application specific computing, our results emphasize the impact of delay on PCMOS bits at NTV and lower voltage operations. This delay is very important to be considered while modeling PCMOS systems, as it propagates and has crucial effects on the most significant bits of the computations.

Acknowledgment

This work was conducted in the context of the ASTRON and IBM joint project, DOME, funded by the Netherlands Organization for Scientific Research (NWO), the Dutch Ministry of EL&I, and the Province of Drenthe. Furthermore, we are thankful to dr.ir. A.J. Annema, ICD group, University of Twente, for simulation support.

References

[1] Moore, Gordon E. "No exponential is forever: but "Forever" can be delayed! [semiconductor industry]." Solid-State Circuits Conference, 2003. Digest of Technical Papers. ISSCC. 2003 IEEE International.

[2] Paul Alexander et al, "SDP Element Concept", SDP-PROP-DR-001-1, Release 1, 2013-06-6.

[3] Jeon, Dongsuk, et al. "A super-pipelined energy efficient subthreshold 240 MS/s FFT core in 65 nm CMOS." Solid-State Circuits, IEEE Journal of 47.1 (2012): 23-34.

[4] Dreslinski, Ronald G., et al. "Near-threshold computing: Reclaiming moore's law through energy efficient integrated circuits." Proceedings of the IEEE 98.2 (2010): 253-266.

[5] Semeraro, Greg, et al. "Energy-efficient processor design using multiple clock domains with dynamic voltage and frequency scaling." High-Performance Computer Architecture, 2002. Proceedings. Eighth International Symposium on. IEEE, 2002.

[6] Greenhalgh, Peter. "Big little processing with arm cortex-a15 and a7-cortex" ARM White Paper (2011): 1-8.

[7] Pinckney, Nathaniel, et al. "Assessing the performance limits of parallelized near-threshold computing." Proceedings of the 49th Annual Design Automation Conference. ACM, 2012.

[8] Pinckney, Nathaniel, David Blaauw, and Dennis Sylvester. "Low-Power Near-Threshold Design: Techniques to Improve Energy Efficiency." Solid-State Circuits Magazine, IEEE 7.2 (2015): 49-57.

[9] Krause, Philipp Klaus, and Ilia Polian. "Adaptive voltage over-scaling for resilient applications." Design, Automation & Test in Europe Conference & Exhibition (DATE), 2011. IEEE, 2011.

[10] Chippa, Vinay K., et al. "Analysis and characterization of inherent application resilience for approximate computing." Proceedings of the 50th Annual Design Automation Conference. ACM, 2013.

[11] KV Palem. "Energy aware computing through probabilistic switching." Technical Report GIT-CC-03-16, Georgia Institute of Technology, 2003.

[12] Palem, Krishna V. "Energy aware computing through probabilistic switching: A study of limits." Computers, IEEE Transactions on 54.9 (2005): 1123-1137.

[13] Korkmaz, Pinar, Bilge ES Akgul, and Krishna V. Palem. "Energy, performance, and probability tradeoffs for energy-efficient probabilistic CMOS circuits." Circuits and Systems I: Regular Papers, IEEE Transactions on 55.8 (2008): 2249-2262.

[14] Cheemalavagu, Suresh, et al. "A probabilistic CMOS switch and its realization by exploiting noise." IFIP International Conference on VLSI. 2005.

[15] Jaeyoon Kim and Sandip Tiwari. "Inexact computing using probabilistic circuits: Ultra low-power digital processing." ACM J. Emerg. Technol. Comput. Syst. 10, 2, Article 16, February 2014.

[16] Korkmaz, Pinar, Bilge ES Akgul, and Krishna V. Palem. "Ultra-low energy computing with noise: Energy performance probability." Emerging VLSI Technologies and Architectures, 2006. IEEE Computer Society Annual Symposium on. IEEE, 2006.

[17] Lau, Mark SK, et al. "Modeling of probabilistic ripple-carry adders." Electronic Design, Test and Application, 2010. DELTA'10. Fifth IEEE International Symposium on. IEEE, 2010.