Ultralow-Energy Variation-Aware Design: Adder Architecture Study

Hamed Dorosti, Ali Teymouri, Sied Mehdi Fakhraie, and Mostafa E. Salehi

***Abstract*— Power consumption of digital systems is an important issue in nanoscale technologies and growth of process variation makes the problem more challenging. In this brief, we have analyzed the latency, energy consumption, and effects of process variation on different structures with respect to the design structure and logic depth to propose architectures with higher throughput, lower energy consumption, and smaller performance loss caused by process variation in application-specific integrated circuit design. We have exploited adders as different implementations of a processing unit, and propose architectural guidelines for finer technologies in subthreshold which are applicable to any other architecture. The results show that smaller computing building blocks have better energy efficiency and less performance degradation because of variation effects. In contrast, their computation throughput will be mid or less unless proper solutions, such as pipelined or parallel structures, are used. Therefore, our proposed solution to improve the throughput loss while reducing sensitivity to process variations is using simpler elements in deep pipelined designs or massively parallel structures.**

***Index Terms*— Adder structures, architecture, deep pipeline, massive parallel, statistical static timing analysis (SSTA), ultra low energy, variation-aware.**

I. Introduction

As technology advances, the density of integrated circuits grows and power consumption becomes more and more serious [1]. This problem affects the performance of design and causes heating and power supply shortage problems. One major solution is using near/subthreshold computing to reduce power consumption over the complex systems-on-chip [2]. Near and subthreshold computing is attractive in energy-constrained applications, such as sensor networks, to increase lifetime and provide energy harvesting capability for some emerging applications.

In subthreshold region, both static and dynamic ingredients of power consumption are severely reduced because of lower supply voltage. However, circuit delay grows exponentially by descending voltage level and hence, the static energy consumption is increased. In minimum energy point of energy-voltage curve, this increase in static energy dominates the dynamic energy consumption, and scaling supply voltage to lower levels means more delay and more total energy consumption [2], [3].

Because of feature size scaling, the impact of process variations becomes significant and near/subthreshold design intensifies the effects of variations and severely degrades the performance parameters [4]–[6]. In order to control process variation effects, we need to do careful timing analysis and employ statistical approaches rather than the classic worst case analysis.

Static timing analysis (STA) was previously implemented in commercial tools [7] and worst case conditions were considered for each cell timing. Then, cell parameters were used to calculate delays of paths in a complex design by adding up delays of gates in series (n = number of gates)

$$\text{Delay}_{\text{Critical}-\text{path}} = \sum_{i=1}^{n} (\mu_i + 3 \times \delta_i) \tag{1}$$

where μ_i and δ_i represent mean and standard deviation of delay for each gate, respectively. In new technologies, variation has grown and using STA yields losing much of the speed performance, unnecessarily. However, statistical STA (SSTA) is another way to analyze the timing specifications of critical paths of a design for getting more realistic results. Variation of each cell is assumed as a normal (Gaussian) variable [5], [8] (2) and (3)[1] [9]

$$\mu_{\text{Critical}-\text{path}} = \sum_{i=1}^{n} \mu_i, \quad \delta^2_{\text{Critical}-\text{path}} = \sum_{i=1}^{n} \delta_i^2 \tag{2}$$

$$\text{Delay}_{\text{Critical}-\text{path}} = \mu_{\text{Critical}-\text{path}} + 3 \times \sigma_{\text{Critical}-\text{path}}. \tag{3}$$

The SSTA is an accepted method based on statistical manner of variations and supported by recent commercial tools [7], [10]. In this method, σ/μ [3], [5], [9] is an important ratio to compare the severity of variations in cells to have better standard cell design in deep subthreshold region. Verma *et al.* [11] extracted logic chains for Kogge–Stone adder (KSA) to measure delay variability in both 0.3 and 1.2 V voltages. σ/μ ratio contours have been drawn based on delay variability histogram, logic depth, and gate width, and variability mitigation is performed by gate up-sizing. Newer technologies such as dual gate silicon on insulator [12] have lower variability in comparison with bulk CMOS to design robust subthreshold logic cells in 32-nm CMOS.

Thakur *et al.* [13] analyzed the effects of variations in gate oxide thickness, supply voltage, and temperature in four adders and they tried to rank the variation effect of each parameter on delay. As a new design method in [14], SSTA is used to sieve a standard cell library with different variation constraints during synthesis of arithmetic circuits. They have verified the results by Monte Carlo simulations. Islam *et al.* [15] have designed a robust (lower σ/μ ratio) subthreshold full adder considering power-delay product. Arthurs and Di [16] evaluate the variations of both Schmitt-trigger and NULL convention logic 1-bit adders by four-gate libraries characterized at different supply voltages for better static noise margin.

In this brief, we use SSTA method to analyze adder structures considering process variations and extract effective architectural level design guidelines to improve speed performance and energy efficiency. The rest of this brief is organized as follows. In Section II, we will introduce advantages and disadvantages of some popular adder structures as basic blocks of arithmetic units. In Section III, we will describe our method, synthesize our candidate circuits, and extract important parameters of each adder. Then, we will analyze the results and introduce some key guidelines. Finally, the conclusion is drawn in Section IV.

[1]Note that the conventional approach would give: $\sigma^2_{\text{Critical}-\text{path}} = (\sum_{i=1}^{n} \sigma_i)^2$, yielding a much wider distribution, that is not consistent with reality.

Manuscript received October 24, 2014; revised February 10, 2015; accepted March 25, 2015.

The authors are with the Nano-Electronics Center of Excellence, School of Electrical and Computer Engineering, University of Tehran, Tehran 14395-515, Iran (e-mail: hdorosti@ut.ac.ir; a.teymouri@ut.ac.ir; fakhraie@ut.ac.ir; mersali@ut.ac.ir).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2015.2426113

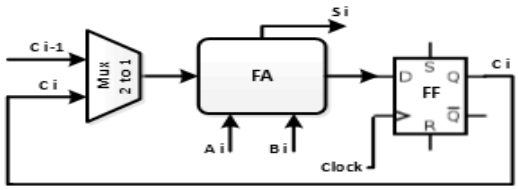

Fig. 1. Single-bit full adder in combination with a flip-flop to do n-bit addition sequentially at different clock cycles.

II. Comparative Study of Some Adder Structures

We choose adder as the key building block of arithmetic units in every processor ranging from general purpose to application specific, because it can be used to implement more complex operations such as multiplication and division or even more complex units, such as fast Fourier transform and finite-impulse response filters. We have selected six different 16-bit adder structures [17], [18] to study in subthreshold region.

Ripple-carry adder (RCA) has simple architecture and linearly extensible for wider computations with respect to area. However, this adder has limited performance because of long carry propagation path from LSB to MSB. Because of long critical path delays in RCA, designers have tried to look ahead carry bit for each higher bit independent of lower neighboring carry bits using a logarithmic-delay tree structure, and each tree optimization strategy implies a new prefix adder.

The first candidate prefix adder discussed is Brent–Kung adder (BKA). This structure has balanced area and timing overheads with shortening the long carry chains $[((2\times\log_2 N)-2)$ logic stages] which is a proper technique to co-optimize area and performance of design. In KSA, addition is performed with higher speed because of parallel computations in shorter paths with only $\log_2 N$ logic stages besides higher area overhead. Han–Carlson adder (HCA) is a combination of BKA and KSA to reduce the complexity and make a tradeoff between area and delay with $\log_2 N+1$ logic stages. Another prefix adder which has minimum logic depth ($\log_2 N$) is known as Lander–Fisher adder (LFA). In this architecture, some nodes have very high fan-outs (up to $N/2$) to reduce the area and this may degrade the performance.

Serial full adder (SFA) is a basic full adder which is combined with a flip-flop to utilize the adder unit at different clock cycles in time-serialized ripple-carry manner (Fig. 1) and the number of clock cycles that it takes is equal to the number of bits.

III. Experimental Results and Analysis

We have synthesized candidate adders from register-transfer level to gate level net-lists using standard synthesis tools. These net-lists are optimized based on the defined constraints to achieve maximum working frequency while we use similar gates as load capacitance. To initiate the synthesis flow, we have introduced a custom 20-cell 90-nm CMOS technology library which is designed for 0.3 V and have characterized it for different supply voltages from 0.3 to 1 V at 0.1 V steps. These libraries are designed using gate sizing with respect to static noise margin and parameters for local and global variations. The effects of process variations on critical path delays are obtained through Monte Carlo SPICE simulations using similar gates as load capacitance and the resulting histogram is fitted to a normal distribution. Therefore, we compare different structures based on synthesis and simulation results. Monte Carlo method simulates the circuit by sweeping the whole variation parameters, such as gate oxide thickness, threshold voltage, and channel length, and does the measurements for iterations, individually.

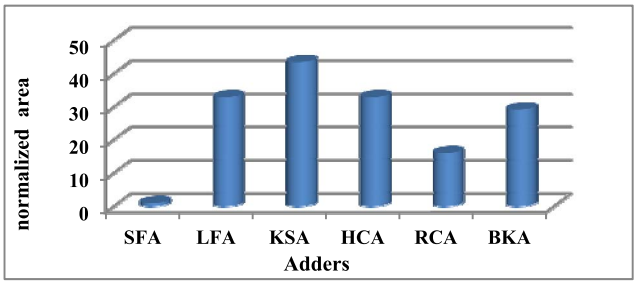

Fig. 2. Area results normalized to SFA in 90-nm CMOS.

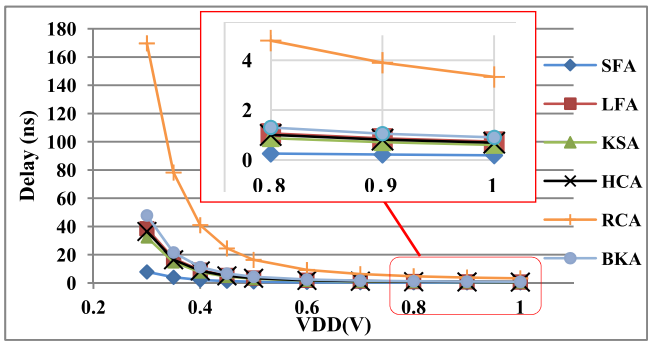

Fig. 3. Critical path delay of different adder structures in 90-nm CMOS.

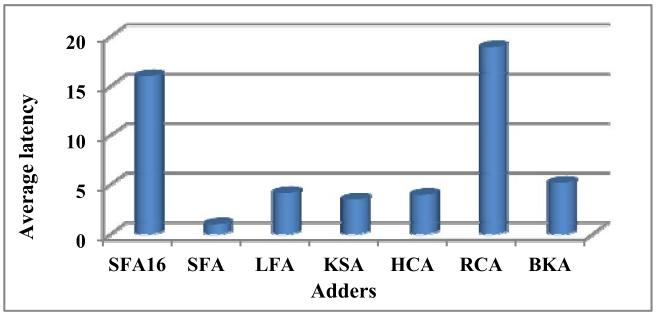

Fig. 4. Average of maximum latency at different voltages to do 16-bit addition for different voltages normalized to SFA.

A. Area

A quick look at Fig. 2 implies that the KSA has the largest area among all adders, and both HCA and LFA have the second place. In addition, the area of RCA structure is the lowest among more complex ones and is almost 16 times bigger than serial single full adder (SFA).

B. Performance and Throughput

The critical path delay as speed performance measure is directly related to the logic depth and driving fan-outs of internal nodes of structures, and every increase in these parameters is translated to more path delay and lower working frequency. Fig. 3 shows the critical path delays of all structures in all expected voltage levels, and confirms our expectation about the fastest (SFA) and slowest (RCA) adders. The second place is for BKA (because of more logic depth) and the third one is for Lander–Fisher due to higher fan-outs (maximum fan-out for $N = 16$ is eight). The comparison between Han–Carlson and Kogge–Stone shows that the logic depth in the first one is 20% more, and the working frequency is almost 10% slower.

Calculation of computational throughput is based on addition latency for the same size inputs. Fig. 4 shows the average of maximum latency of different adders at different voltages to perform full-length addition of 16-bit operands as a measure of

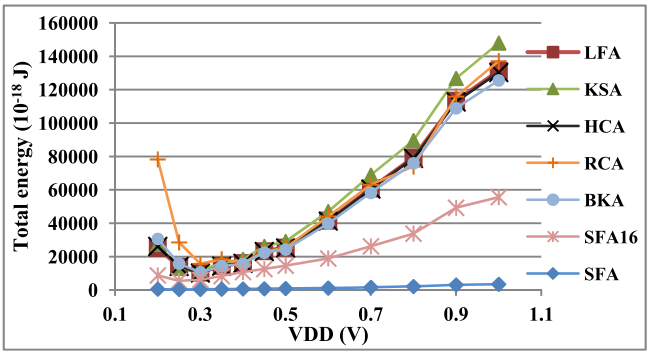

Fig. 5. Total energy consumption of different structures.

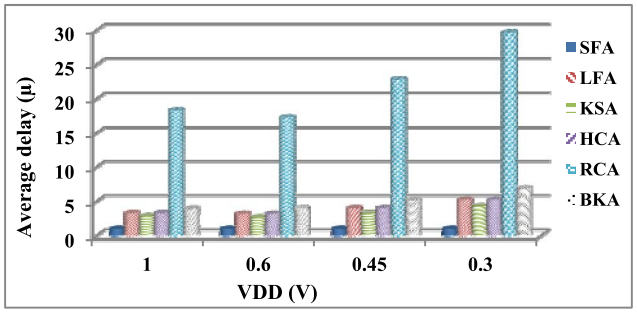

Fig. 6. Average delay (μ) parameter of critical paths normalized to SFA.

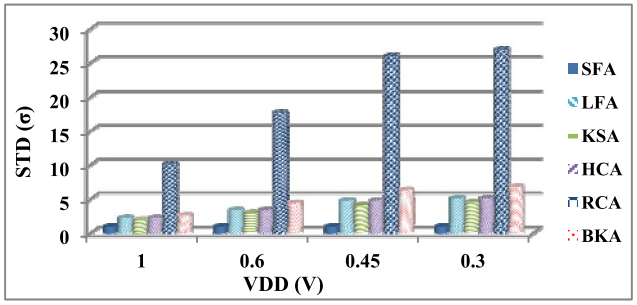

Fig. 7. STD (σ) parameter of critical paths normalized to SFA.

computation throughput. Obviously, statistical Monte Carlo analysis has been used for throughput measurement. As shown, SFA has higher throughput than RCA adder due to accumulated delay variation at worst case design corners for RCA elements, whether the addition algorithm is the same. The Kogge–Stone has the best throughput among all candidates.

C. Energy Consumption or Power-Delay Product

Energy consumption or power-delay product is an important metric in energy-constrained systems besides speed. The dynamic energy is reduced at lower supply voltages according to (4) and it follows the dynamic power trend-line. While performing computation in RCA compared with SFA, there are more active nodes with unnecessary transitions, and the difference in dynamic energy is obvious from

$$E_{\text{Dynamic}} \sim \frac{1}{2} C_L V_{\text{DD}}^2. \tag{4}$$

Static energy consumption is determined by area and duration of operation that is decided by critical path delay, and this may change the rank of a structure compared with static power. RCA and BKAs have the highest static energy because of longer critical path and larger area, respectively. The SFA has the lowest static energy due to smaller area and shorter critical path delay and the other candidates have closer energy consumptions because of closer areas and path delays.

Fig. 5 shows the total energy consumption of adders in the range of 0.2–1 V power supply voltages. According to this figure, almost minimum energy point is reached at 0.3 V power supply, and it is apparent that SFA performing 16-bit operation has the lowest energy consumption in comparison with the others due to lower static and dynamic energy ingredients.

At voltages above the minimum energy point, because of dominant dynamic part, the Kogge–Stone consumes more energy and in ranges below the minimum point due to dominant static part, RCA has the first place for its higher delay. In other words, above the minimum energy point the complexity determines the amount of energy consumption, and below this point the effect of delay prevails to the extent that dominates the effect of complexity.

D. Variations

Because of higher process variations in newer technologies, more specifically in subthreshold region, the timing specifications of design widely varies in different chips. Using SSTA to analyze timing specifications, (2) and (3) imply that any increase in gate variation or logic depth (longer critical path) will worsen the variation fluctuations of critical paths. In this brief, due to similarity in variations of different gates existing in our developed library the logic depth has more importance in final variation status.

The parameters of delay distributions are derived from Monte Carlo simulations considering both of global and local variations at different supply voltages and are normalized to the SFA adder. Fig. 6 shows the ratios of averages of critical path delays at four different supply voltages for different adders normalized to SFA. Fig. 7 shows the normalized ratios for delays standard deviations. According to these figures, longer path (RCA) in a design means more variability and more deviations in timing specifications and shorter paths (SFA) mean greater certainty in working frequency of the design.

According to experimental results, using single bit full adder in multiple clock cycles improves area and energy consumption by factors of 40× and 3×, respectively. In addition, serial structure suffers less by factor of 2 from process variation in comparison with KSA as the fastest design, because of breaking the logic depth. These improvements are achieved by degrading the computation throughput by a factor of 3. This structure is attractive for energy-constrained applications with low or mid speed requirements. In many scientific applications with regular processing algorithms, such as dense matrix multiplication and image processing [19], [20], high-performance low-cost architecture is required. To compensate the degradation of computation throughput, using three serial adders in parallel structure, we can achieve the same speed performance as KSA with the similar amount of energy consumption, while the area difference is so far.

Using higher levels of parallelization or deeper pipeline structures causes the throughput to rise significantly. In addition, shorter logic paths mean that the effect of process variations on working frequency is less and higher throughput with more certainty can be achieved besides lower energy consumption. However, bigger designs with longer critical paths dissipate more static energy and have increased uncertainty in working frequency and achievable computational throughput. Table I presents the results of parallel and pipelined structures in comparison with baseline design for RCA and confirms the previous analysis.

The analysis clarify that using serial single bit computing unit in consequent cycles to do the operation in more than one clock cycle causes to leak less during computation cycle in newer technologies

TABLE I
PERFORMANCE MEASUREMENT OF DIFFERENT CONFIGURATIONS AT 0.3 V

Configuration	RCA	SFA16	Pipelined RCA (16stages)	Parallel SFA16 (16units)
Area (um^2)	179	11.2	179	179
Energy(10^{-18}J)	15727.3	6128.6	98057.6	98057.6
Throughput(MOPS)	4	7.1	114	114
Frequency(MHz)	4	114	114	114
Latency(ns)	246.3	140.5	140.5	140.5
Energy/Throughput	3932	863	860	860

specifically in subthreshold region, and the effect of process variations on working frequency is reduced. However, the mid performance is achieved between candidate structures. Therefore, smaller computation width in hardware and spreading the computation complexity over the time will improve area and energy efficiency significantly, and reduces throughput loss due to process variation. Utilizing parallel and pipelined structures will improve the throughput besides keeping the previous achievements.

IV. CONCLUSION

In this brief, we have analyzed the latency, energy consumption, and effects of process variation on different adder structures as different implementations of a popular processing unit with respect to the design structure and logic depth to propose architectural guidelines. These guidelines are applicable to any other architecture without any dependence to functionality of the design to achieve higher throughput, lower energy consumption, and smaller performance loss caused by process variation in application-specific integrated circuit design.

Simulation results and analysis confirm that, SFA has smaller area, less timing fluctuations, and the highest working frequency, and its throughput is similar to RCA. Utilizing SFA in parallel architecture or pipelined version of RCA improves the throughput besides the energy efficiency and variation resistance. Therefore, in order to decrease the variation effects and to increase the throughput/performance of design, we need to use deeper pipelines such as systolic arrays or massively parallel designs such as graphics processing unit structures with simpler building blocks. Increasing the pipeline depth in a design causes to break the paths into shorter sections to increase the throughput and decrease variations. Simpler computational building blocks consume lower energy and observe lower performance variations too. Finally, we conclude that utilizing such blocks in a massively parallel architecture is another way to compensate the process variation effects and lower the frequency uncertainty plus lowering timing fluctuations due to process variations.

REFERENCES

[1] M. B. Taylor, "A landscape of the new dark silicon design regime," *IEEE Micro*, vol. 33, no. 5, pp. 8–19, Sep./Oct. 2013.

[2] A. Wang, B. H. Calhoun, and A. P. Chandrakasan, *Sub-Threshold Design for Ultra Low-Power Systems*. New York, NY, USA: Springer-Verlag, 2006.

[3] Z. Bo *et al.*, "Energy-efficient subthreshold processor design," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 17, no. 8, pp. 1127–1137, Aug. 2009.

[4] H. Iwai, "Roadmap for 22 nm and beyond (Invited Paper)," *Microelectron. Eng.*, vol. 86, nos. 7–9, pp. 1520–1528, 2009.

[5] *International Solid State Circuits Conference 2013 Trends*. [Online]. Available: http://isscc.org/doc/2013, accessed 2014.

[6] X. Chen, L. Yang, R. P. Dick, L. Shang, and H. Lekatsas, "C-pack: A high-performance microprocessor cache compression algorithm," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 18, no. 8, pp. 1196–1208, Aug. 2010.

[7] *Synopsys On-Line Documents*. [Online]. Available: http://www.synopsys.com/support/pages/dow.aspx, accessed 2014.

[8] A. Srivastava, D. Sylvester, and D. Blaauw, *Statistical Analysis and Optimization for VLSI: Timing and Power*. New York, NY, USA: Springer-Verlag, 2006.

[9] S. R. Sarangi, B. Greskamp, R. Teodorescu, J. Nakano, A. Tiwari, and J. Torrellas, "VARIUS: A model of process variation and resulting timing errors for microarchitects," *IEEE Trans. Semicond. Manuf.*, vol. 21, no. 1, pp. 3–13, Feb. 2008.

[10] M. Tehranipoor, K. Peng, and K. Chakrabarty, *Test and Diagnosis for Small-Delay Defects*. New York, NY, USA: Springer-Verlag, 2011.

[11] N. Verma, J. Kwong, and A. P. Chandrakasan, "Nanometer MOSFET variation in minimum energy subthreshold circuits," *IEEE Trans. Electron Devices*, vol. 55, no. 1, pp. 163–174, Jan. 2008.

[12] R. Vaddi, S. Dasgupta, and R. P. Agarwal, "Device and circuit co-design robustness studies in the subthreshold logic for ultralow-power applications for 32 nm CMOS," *IEEE Trans. Electron Devices*, vol. 57, no. 3, pp. 654–664, Mar. 2010.

[13] A. Thakur, D. Chilamakuri, and D. Velenis, "Effects of process and environmental variations on adder architectures," in *Proc. 49th IEEE Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2006, pp. 36–40.

[14] J. Crop, R. Pawlowski, N. Moezzi-Madani, J. Jackson, and P. Chaing, "Design automation methodology for improving the variability of synthesized digital circuits operating in the sub/near-threshold regime," in *Proc. Int. Green Comput. Conf. Workshops (IGCC)*, Jul. 2011, pp. 1–6.

[15] A. Islam, A. Imran, and M. Hasan, "Robust subthreshold full adder design technique," in *Proc. Int. Conf. Multimedia, Signal Process. Commun. Technol. (IMPACT)*, Dec. 2011, pp. 99–102.

[16] A. Arthurs and J. Di, "Analysis of ultra-low voltage digital circuits over process variations," in *Proc. IEEE Subthreshold Microelectron. Conf. (SubVT)*, Oct. 2012, pp. 1–3.

[17] M. Talsania and E. John, "A comparative analysis of parallel prefix adders," in *Proc. Int. Conf. Comput. Design*, Las Vegas, NV, USA, Jul. 2013, pp. 29–36.

[18] B. Parhami, *Computer Arithmetic: Algorithms and Hardware Designs*. London, U.K.: Oxford Univ. Press, 2009.

[19] K. T. Johnson, A. R. Hurson, and B. Shirazi, "General-purpose systolic arrays," *IEEE Comput.*, vol. 26, no. 11, pp. 20–31, Nov. 1993.

[20] M. Bekakos, I. Ž. Milovanović, T. I. Tokić, Ć. B. Dolićanin, and E. I. Milovanović, "Selecting mathematical method for systolic processing," *Sci. Pub. State Univ. Novi Pazar A, Appl. Math., Inf. Mech.*, vol. 3, no. 1, pp. 53–58, 2011.