CAS-FEST 2010: Mitigating Variability in Near-Threshold Computing

Mingoo Seok, Gregory Chen, Scott Hanson, Michael Wieckowski, David Blaauw, and Dennis Sylvester

***Abstract*—Near threshold computing has recently gained significant interest due to its potential to address the prohibitive increase of power consumption in a wide spectrum of modern VLSI circuits. This tutorial paper starts by reviewing the benefits and challenges of near threshold computing. We focus on the challenge of variability and discuss circuit and architecture solutions tailored to three different circuit fabrics: logic, memory, and clock distribution. Soft-edge clocking, body-biasing, mismatch-tolerant memories, asynchronous operation and low-skew clock networks are presented to mitigate variability in the near threshold $V_{\rm DD}$ regime.**

***Index Terms*—Low voltage, near threshold computing, variability.**

I. Introduction

POWER consumption is an increasingly critical issue in modern integrated circuit (IC) design since power density increases sharply with process technology scaling, and the total number of devices per design continually grows. Constant-field technology scaling theoretically reduces the energy per logic device cubically, as shown in (1). However, these ideal gains are not realized in modern designs. Firstly, this assumption is contingent on the devices consuming solely switch energy. However, the rapid increase of subthreshold and gate leakage components in modern MOSFETs make these static components a significant contribution to total energy consumption. Subthreshold and gate leakage currents increase as device $V_{\rm TH}$ and gate oxide thicknesses decrease. Secondly, constant-field scaling relies on $V_{\rm DD}$ scaling proportionally with minimum features size. However, technology scaling trends exhibit slow $V_{\rm DD}$ scaling, such that switching energy only improves linearly (as shown in (2)). Fig. 1 shows typical $V_{\rm DD}$'s in recent and future technology nodes, with $V_{\rm DD}$ scaling stalling near 1 V.

$$\mathrm{E}_{\text{switch,constant-field}} = \mathrm{C}\cdot {V_{\rm DD}}^2 \propto \frac{1}{S}\left(\frac{1}{S}\right)^2 = \left(\frac{1}{S}\right)^3 \quad (1)$$

$$\mathrm{E}_{\text{switch,no-vdd-scaling}} = \mathrm{C}\cdot {V_{\rm DD}}^2 \propto \frac{1}{S}(1)^2 = \frac{1}{S} \quad (2)$$

$$\mathrm{E}_{\text{switch,density}} = \mathrm{C}\cdot {V_{\rm DD}}^2 \cdot \frac{1}{A} \propto \frac{1}{S}(1)^2 S^2 = S. \quad (3)$$

Manuscript received October 19, 2010; revised January 08, 2011; accepted February 05, 2011. Date of publication May 02, 2011; date of current version May 25, 2011.

The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: dennis@eecs.umich.edu).

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/JETCAS.2011.2135550

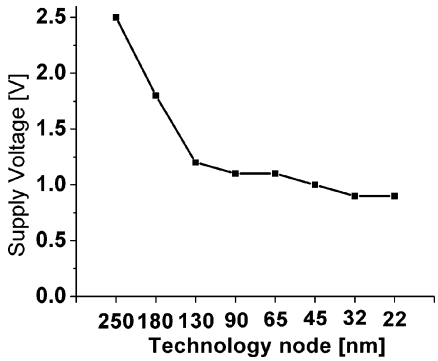

Fig. 1. Supply voltage trends with technology scaling.

Taken together, slow $V_{\rm DD}$ scaling and increased leakage powers have led to poor improvements in computational energy efficiency. Moreover, energy density increases with minimum feature size, as shown in (3), which causes heat removal problems in integrated circuits and packages. At the same time, we are entering a "more than Moore" world where computing is used for a multitude of applications including high-end servers, personal computing devices, and pervasive sensor motes. In all these applications, we are seeing a greater number of devices per system, making energy efficiency critical. Higher energy efficiency would lower the cost of cooling infrastructures in data centers. It also enables personal computing devices with higher performance and longer battery lifetime. Finally, it would create possibilities for new sensing applications which require ultra-low power and energy-autonomy. Therefore, today's IC designers face a challenging problem to improve circuit performance and functionality while minimizing energy consumption at the same time. Improving energy efficiency is not a problem associated with a specific computing application, but rather is a requirement for all applications.

Scaling $V_{\rm DD}$ serves as the strongest knob for reducing energy consumption and energy density by reducing switching energy quadratically. However, it can degrade circuit performance. Therefore for peak computational energy efficiency and reasonable performance, it is proposed to scale $V_{\rm DD}$ down to near $V_{\rm TH}$ [20]. This operation is called near-threshold computing (NTC). The same factors that limit $V_{\rm DD}$ reductions with process scaling create design challenges in NTC circuits. Performance degrades as gate overdrives and currents decrease. Moreover, performance variation increases between nominally identical circuits with different placements in a chip, locations on a wafer, or lots during processing. NTC circuits exhibit more variability since the higher sensitivity of MOSFET current to

V_{TH} variations, both local (uncorrelated) and global (correlated) process variations. Among local variations, non-uniform doping called random dopant fluctuation (RDF) causes V_{TH} variation. Also, imprecise lithography called line edge roughness (LER) causes length variation. Global variation may also stem from doping or geometric differences. In addition to increasing performance variability, these sources of variation may also cause functional yield issues, particularly for state-holding elements, such as static random-access memories (SRAM).

In this tutorial paper, we review near-threshold computing as a powerful option for improving energy efficiency. We examine the benefits and challenges of NTC computing in Section II. In Section III, we focus on design solutions for mitigating variability in three NTC circuit fabrics: logic, memory, and clock network. Proposed techniques include averaging uncorrelated variations, designing mismatch-tolerant circuits, and adaptively correcting for non-idealities. Finally, we conclude in Section IV.

II. Near-Threshold Computing (NTC)

A. Improved Energy Efficiency

Scaling V_{DD} to the NTC region reduces energy and energy density quadratically, while simultaneously reducing subthreshold and gate leakage in a super-linear to exponential manner by mitigating short-channel effects and reducing gate oxide electric field. Therefore, voltage scaling has been popular to improve energy efficiency in both industry and academic designs. However, V_{DD} scaling is not free. It degrades performance and noise margins. So V_{DD} must be intelligently scaled to balance these design tradeoffs. Several commercial products only use voltages scaled down to about 70% of the nominal voltage due to the constraints on performance[2]–[4].

V_{DD} can be scaled to near or below threshold voltage of devices to achieve maximum energy savings when performance is less constrained. As V_{DD} is continually scaled, circuits will eventually approach the theoretical limit for functional CMOS operation. This V_{DD} is given by $8kT/q$ (k: Boltzmann constant, T: temperature, q: electron charge) by Meindl *et al.* [1]. However, the lowest functional V_{DD} is not the energy optimal point. Zhai *et al.* [5] and Calhoun *et al.* [19] pointed out that there is an intermediate energy-optimal supply voltage for CMOS circuits. As the supply voltage scales from nominal V_{DD}, the energy initially reduces in a quadratic fashion due to savings in switching energy, which accounts for the majority of the total energy in the region. However, leakage energy exponentially increases as V_{DD} scales, eventually offsetting the dynamic energy savings. Therefore as shown in Fig. 2, the total energy is minimum at an intermediate voltage, defined as the energy-optimal V_{DD} or V_{min}. V_{min} is commonly subthreshold, and at V_{min} we expect about 20 × energy savings over nominal operation.

The energy savings from operation at the energy-optimal V_{DD} come with 500–1000× performance degradation relative to nominal operation. Interestingly, most of the energy savings are generated from scaling V_{DD} to a voltage just above the threshold voltage of devices, as shown in Fig. 2. This yields a 10× energy decrease, while additional V_{DD} scaling to V_{min} results in only an additional 2× benefit. Energy benefits are not as great below NTC since leakage energy consumption becomes more significant. Also, performance degrades gracefully from nominal V_{DD} to the near threshold regime while rapid degradation occurs between NTC in subthreshold operation. Therefore, NTC provides a better energy-delay tradeoff.

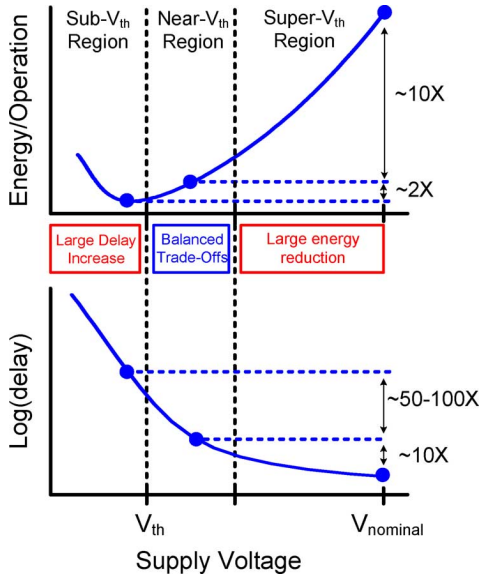

Fig. 2. Energy and delay in different V_{DD} operating regions [20].

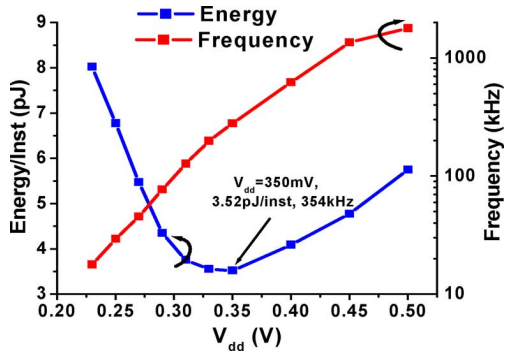

Fig. 3. Subliminal processor frequency and energy breakdowns at various supply voltages [6].

One demonstrated microcontroller design for low voltage operation clearly shows the benefit of NTC over subthreshold computations. As shown in Fig. 3, the energy optimal supply voltage for this design is 350 mV [6]. The processor consumes 3.52 pJ/inst while operating at 354 kHz, which is 9 × energy savings and 57× performance degradation, compared to the design operating at nominal supply voltage. However, at near threshold voltages, the performance degrades by only 11.4× while energy consumption is still 6.6× lower. This is a favorable energy-delay tradeoff for both energy and performance constrained systems, while subthreshold computing is often limited to niche applications with relaxed performance requirements (e.g., sensor applications).

B. Design Challenges

Although NTC provides a better energy-performance tradeoff, it leads to several challenges that must be overcome.

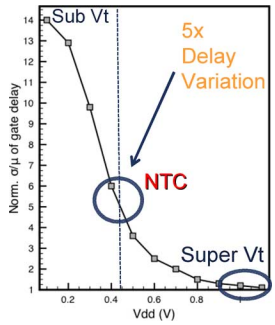

Fig. 4. Impact of voltage scaling on gate delay variability [20].

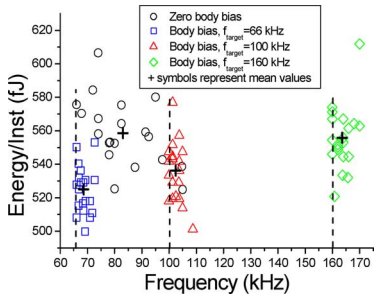

Fig. 5. Body biasing techniques for three target frequencies [6].

These challenges include performance degradation, variability, and circuit robustness. As pointed out in Section II-A, CMOS circuits that operate in the near threshold regime exhibit significantly degraded performance. This is inevitable due to the reduced overdrive in gate voltage. In an industrial 45 nm technology, the fan-out-of-four inverter delay (FO4) at a near threshold voltage of 400 mV is 10 × slower than at a nominal supply voltage of 1.1 V. In the following subsection, we will focus on the variability challenge in NTC designs.

1) Increased Performance Variation: In the near-threshold regime, the driving current of MOSFETs is exponentially sensitive to $V_{\rm TH}$, $V_{\rm DD}$, and temperature variations due to a greater contribution of subthreshold current to driving current. As shown in Fig. 4, the gate delay in near threshold regimes ($\sim$ 450 mV) varies by 5× more than at nominal voltage with only global process variations. Furthermore, assuming supply voltage and temperature fluctuations increase the variability by 2×, respectively, the total delay variability of a gate can be up to 20×. Compared to a typical delay variability of 30%~50% over process, voltage and temperature variations [7], the large variability at near-threshold regimes poses a significant challenge to design NTC circuits. Note that simply adding delay margin is not feasible in NTC circuits since the prohibitive amount of margins to cover the large variability degrade circuit performance and energy efficiency. Local process variations also contribute significantly to delay variation.

2) Increased Functional Failure: The high sensitivity of MOSFETs to process, voltage and temperature in near threshold regimes can not only increase performance variability but also lead to functional failures. In particular, local process variations such as RDF and LER can cause a significant stability problem in the circuits based on contentions. Inverters connected in a positive feedback are used in latches and SRAM. These structures suffer from less read and write stability when the relative strengths among nominally matched MOSFETs deviate largely from their design values. In particular, SRAM arrays often employ the smallest device allowed in the technology, which makes them even more vulnerable to those local variations.

III. Mitigating Variability in Different Circuit Fabrics

In this section, we focus on variability problems in NTC and present the recent advances in circuit and architecture techniques to mitigate variability. Since each circuit fabrics require different measures to be most effective, we discuss mitigating strategies for logic, memory, and clock network, separately. Also we specify the nature of variation source, either random or global, that each technique targets to mitigate. Random variation is roughly defined as one that differently affects individual devices that are closely placed in design. Global variation contrarily affects every device in the same way across dies or wafers.

A. Logic

Traditionally, adding delay margin has been used to cope with logic delay variability. However, this is inadequate in NTC since the larger variability forces prohibitively large margins, substantially hurting the energy and performance benefit of NTC. Hence, in this section we discuss architectural and circuit solutions without adding delay margin for variation tolerance.

1) Body Biasing: Body biasing is a well-known technique for adapting performance and leakage to global variations of process, voltage, and temperature variations. While effective in nominal voltage operations, body biasing is particularly effective in near threshold regimes since device is exponentially sensitive to $V_{\rm TH}$ changes. Thus, body biasing is a strong lever for modulating the performance of NTC circuits, which help to mitigate delay variability from process and environmental variations.

Hanson *et al.* applied the body bias technique to a subthreshold processor [6] for addressing variability from process variations. The body bias is applied globally, i.e., every negative field effect transistor (NFET) shares p-well bias while every positive field effect transistor (PFET) shares n-well bias. As shown in Fig. 5, the test chips initially exhibit a wide spread of performance. However, the body biasing can minimize the spread as well as change a center frequency, 66, 100, and 160 kHz in the experiments, which confirm the effectiveness of the body biasing techniques. Hwang *et al.* also employed global adaptive body biasing in a closed loop to dynamically compensate for NFET versus PFET drive strength differences across a range of operating points. This technique enabled functionality of a digital filter for a wide range of supply voltage from nominal 1.2 to 85 mV [21].

While effective to mitigate the variability from global variations, body biasing technique is often less practical to cope with local variations since a finer-grained body biasing technique can cause significant control and area overheads.

2) Soft Edge Clocking: One of the most straightforward methods to mitigate the variability from local variations is to use deep pipeline stages, i.e., longer delay paths per stage. Since each gate delay increases or decreases in a random manner, path

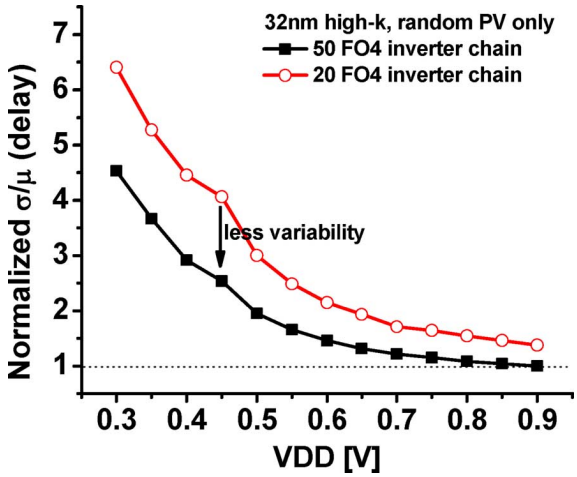

Fig. 6. Delay variability with long and short FO4 inverter chain.

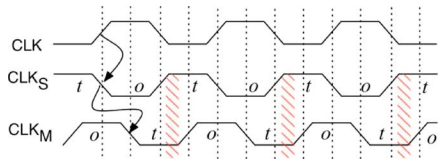

Fig. 7. Delaying the master clock creates a window of transparency [9].

delays consisting of several gates often exhibit significantly reduced variability by so-called averaging effect. Fig. 6 shows that delay variability from random process variations can be mitigated in a long pipeline stage, based on SPICE simulations with 32 nm industrial CMOS technology. However, deep pipeline stage degrades performance which could have been improved by more aggressive pipelining. Therefore, this technique is mostly used for the designs with relaxed performance constraints such as sensing systems [6], [8].

One potential solution to address delay variability while minimizing performance overhead is to use a type of soft-edge flip-flop (SFF) as a sequential element for pipelining. A SFF is identical to a D-flip-flop (DFF) except the fact that the SFF has a short transparent window or softness between clock signals of master and slave latches. This softness can provide an extra clock time if proceeding paths cannot finish switching within a nominal clock period. Hence, soft edge clocking can mitigate the delay variability from random, uncorrelated variations, which are significant in NTC. In essence, it is similar to a time borrowing ability of two-phase latches. However SFFs still synchronize at a clock edge, causing less design complexity. The amount of softness can be configured post-silicon using tunable delay elements for preventing hold time violations. Fig. 7 shows two clock signals for master and slave latches and transparent windows (shaded) created in a SFF.

Wieckowski *et al.* [9] exploited this technique in designing a finite-impulse response (FIR) filter to show that small amounts of softness achieved 11.7% performance improvement over a standard DFF design and improvement of 9.2% compared to a DFF with useful skew. These gains, shown in Fig. 8, demonstrate a greater tolerance to local variation that becomes even more important in the NTC operating region.

B. Memories

It is well-known that conventional 6T SRAM exhibits more variability due to positive feedback loops than logic at low V_{DD}. In addition, SRAM is more susceptible to V_{TH} variation caused by random dopant fluctuation (RDF) since SRAM often employs smallest devices for higher density. At NTC supply voltages, RDF is the dominant form of process variation and the foremost reason for variability in SRAM. In this respect, it is critical and challenging to variability manifested in SRAM.

1) Near-Threshold SRAM Bitcell Design: One of the most straightforward methods to improve variability in 6T SRAM is upsizing of transistors in a bitcell. This serves to average out non-uniformities in channel doping, resulting in more uniform device V_{TH}'s [14] at the cost of larger area and energy consumption.

Alternatively, supply or word line modulation can be used to improve read and write variability while still using a 6T structure [15], [23]. One work by Zhai *et al.* uses a bitcell optimized for single-ended read stability. A supply modulation technique is then used on a per column basis to improve write-ability. Thus, the read and write operations are effectively decoupled by adding complexity in the periphery of the core array. As another example, authors in [23], [28] propose to use differential 6T SRAM bitcells with assist circuitry or 2-cycle writes for improving variability. Error correction codes are another effective measure to mitigate variability as in [31], demonstrated using 6T bitcells.

Instead of relying on the 6T bitcell structure, some authors propose the use of alternative bitcell structures [16], [24], [29], [30]. The most common alternative cell is described in [16], namely an 8T SRAM bitcell based on the premise of decoupling the read and write operations by adding an isolated read-out buffer (Fig. 9). This effectively allows for separate optimization of write and read operations, resulting in higher design robustness to variability. Starting from the basic 8T SRAM array, [29] further addresses variability by utilizing reverse short channel effects and marginal bit line leakage compensation techniques. Additionally, incorporating sense amplifiers in 8T SRAM array is investigated in [24] in an effort to compensate for variability using redundancy. Chen *et al.* [18] show that the alternative 8T bitcell actually requires the least area overhead to mitigate variability when selecting among upsized 6T, 6T with a write assist, and upsized 8T using a systematic analysis. This result is shown in Fig. 10.

Along with 8T bitcells, 10T bitcells have also been proposed. Chang and Kulkarni use 10T bitcells, which help to mitigate variability through differential read schemes [25], [26]. On the other hand, Kim employs a single-ended read with data dependent bit line leakage compensation scheme to mitigate process and environmental variations [30].

2) Crosshairs SRAM: To mitigate variability in SRAM, [17] proposes the Crosshairs method to detect and adaptively correct parametric failures. While using the same area as a traditional 6T SRAM bitcell, Crosshairs tunes the SRAMs power and ground supply networks to mitigate excessive variation. Fig. 11 schematically shows that the strength of pull-up and pull-down devices can be tuned by connecting V_{DD} or $\mathrm{G_{ND}}$ of a bitcell to

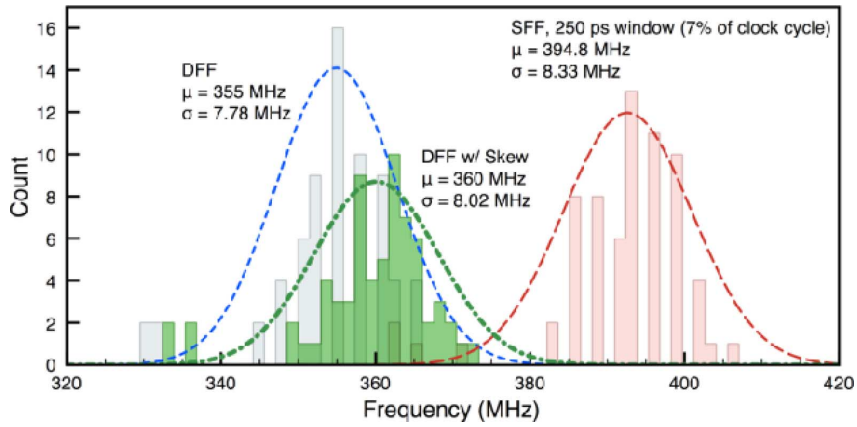

Fig. 8. FIR filter with soft edge clocking compared to standard flip-flops (SFF); presented with and without useful skew [9].

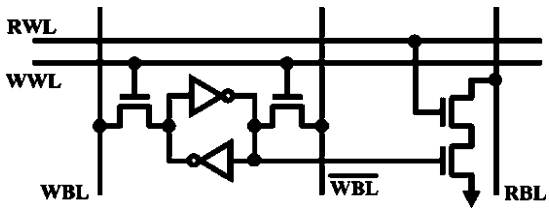

Fig. 9. Alternative 8T SRAM Cell, decoupling read and write.

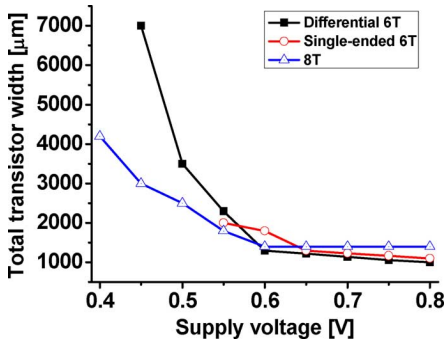

Fig. 10. Total transistor width for two-cycle delay at iso-robustness.

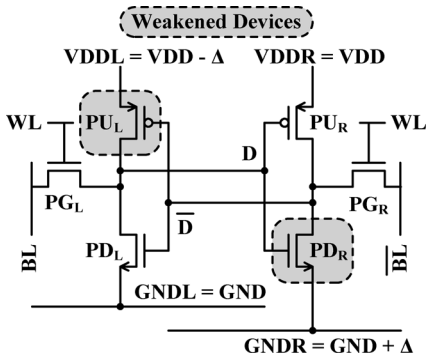

Fig. 11. V_{DD} and G_{ND} configuration in Crosshairs [17].

different potential. It improves yield with respect to timing and stability constraints.

For accommodating this scheme with less overhead, each bitcell has connections to left and right vertical power rails (VDDL and VDDR) and horizontal ground rails (GNDL and GNDR).

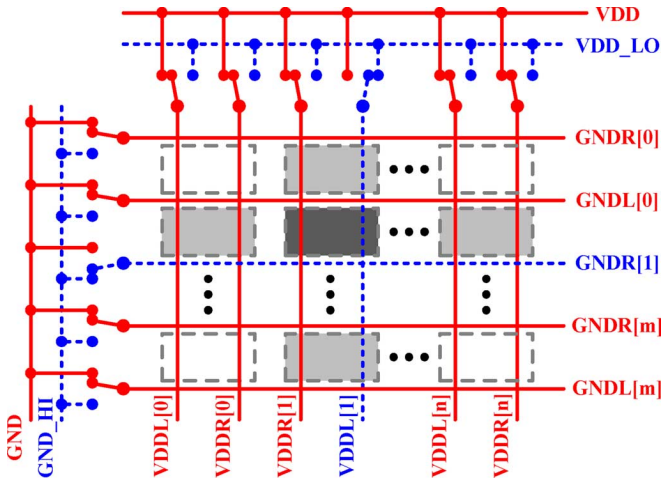

Fig. 12. V_{DD} and G_{ND} potentials controlled by header and footer cells increasing array efficiency[17].

The bitcell is identical to a commercial differential 6T design except that the vertical V_{DD} rail is split into VDDL and VDDR, as shown Fig. 12.Crosshairs eliminates variability by adjusting V_{DD} in the column and Gnd in the row where a parametric failure occurs, as shown in Fig. 12. Thus, the orthogonal tuned supply rails target parametric failures at their intersection

By targeting slow bitcells due to local process variation creates, which dictate the overall performance of an SRAM array, Crosshairs mitigates variability and increases array performance by 13% at a tuning voltage of 20 mV from the measurement from 70 chips with 128 ×256 32 kb Crosshairs SRAM banks in a 45 nm CMOS process (Fig. 13).

3) Asynchronous Write Operation: Asynchronous operation is an attractive method to mitigate variability since it can eliminate delay margin which can cause performance and energy overhead. While often causing a large overhead in gate level, NTC-friendly bitcells like 8T (Fig. 9) enable employing a detector of asynchronous write completion with little overhead in memory peripherals. Seok *et al.* [8] suggested asynchronous write detection where the memory signals the processor when a write operation completes by XOR-ing its read and write ports, as shown in Fig. 14. Delaying the completion signal is needed to ensure the completion of writing operation at both nodes of a

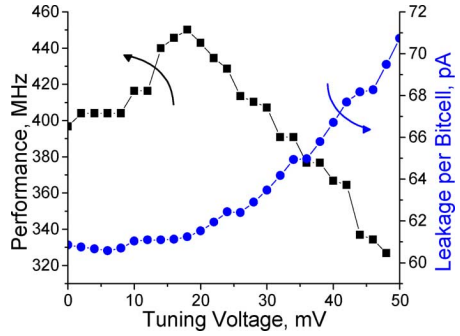

Fig. 13. Crosshairs improve array performance by 13% and have a modest leakage overhead [17].

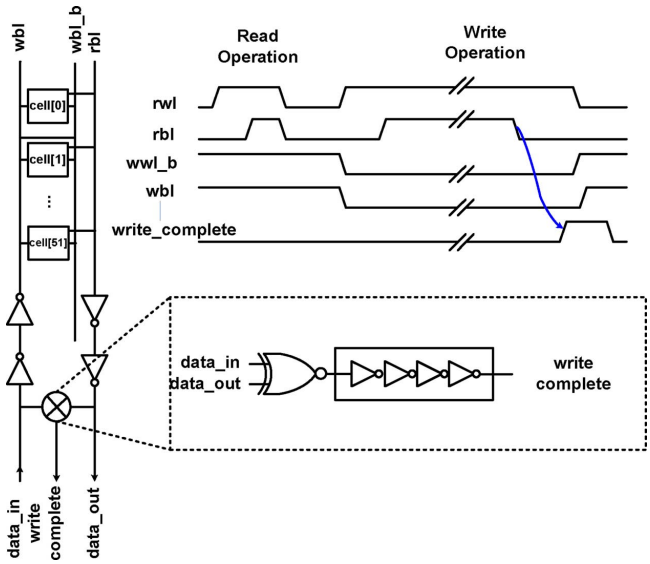

Fig. 14. Asynchronous write completion detection.

bitcell since read-out senses only one node of back to back inverters in a SRAM bitcell.

Compared to read operation, multi-cycle asynchronous writes have little impact on performance in microcontrollers since it can be performed in background unless a processor requests several consecutive write operations.

C. Clock Network

Clock network design presents a unique challenge for mitigating variability in NTC. With the highest switching activity, the clock network consumes up to $\sim$ 40% of total dynamic power in many designs [11]. With similar trends in NTC, clock networks have a large impact on total energy consumption and require additional design consideration. Along with the low power requirements, the clock network should be designed for low variability as well. As shown in (4), skew must be well-defined against process and environment variations, otherwise the design can have short paths and functional failures [12] that designers are unaware of in design time. Additionally, slew needs to be well-controlled since it degrades the setup and hold time of registers.

Authors in [27] tackled the problem by proposing the use of charge-pump based clock buffers. Due to the exponential

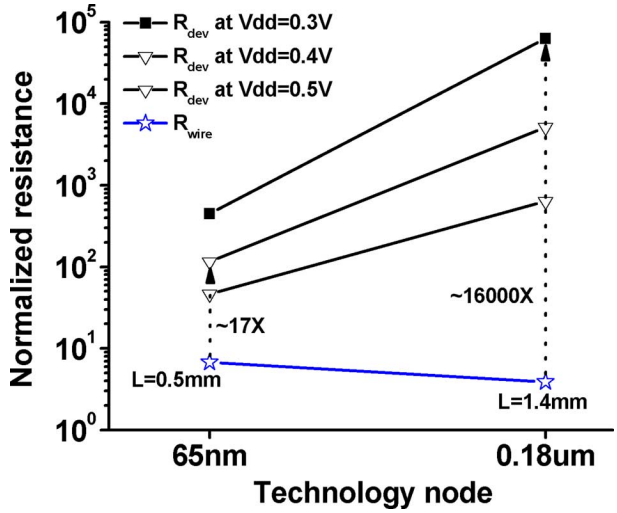

Fig. 15. Device and interconnect resistance trends [10].

sensitivity of subthreshold current to gate-source voltage, these custom buffers can significantly improve performance and mitigate variability. However, it incurs energy overhead, which calls for more energy-efficient techniques.

One interesting observation is that interconnect resistance is relatively negligible, compared to the on-resistance of transistors at near-threshold or subthreshold regimes. As shown in Fig. 15, relatively long wires (0.5~1.4 mm) have orders of magnitude lower resistance than the on-resistance of MOSFETs. More scaled technologies have smaller gaps in resistance, however it remains approximately 20× between a 0.5 mm long wire and a moderately sized MOSFET at $V_{\mathrm{DD}} = 0.5$ V.

$$\mathrm{T_{cq,reg}(T_{clk,slew})} + \mathrm{T_{min,logic}} \geq \mathrm{T_{hold}(T_{clk\ slew})} + \mathrm{T_{clk,uncertainty}}. \quad (4)$$

Due to the negligible interconnect resistance, Seok *et al.* [10] found that clock network design methodology in the NTC regime can be radically different from traditional designs. Typically in super-threshold regimes, designers add buffers to improve slew and balance the buffers to minimize skew. However it becomes disadvantageous in NTC regimes since buffer delay varies up to 20× (as shown in Section II-B) with process, temperature, and supply voltage variations and thus degrades skew and slew variability, while reducing already negligible skew and slew contributions of interconnects. Therefore, they propose a method using no or very few buffers inside clock networks for minimizing skew and slew variability.

As a case study, seven clock networks shown in Fig. 16 are implemented for a 16 b MSP430-like microprocessor [13] and simulated in SPICE. Results confirm that an optimally selected clock network, signal route or un-buffered H-Tree, greatly outperforms other typical clock networks in skew and slew variability with no energy overhead, as seen in Fig. 17.

IV. Conclusion

NTC is one of the most attractive approaches to mitigate the prohibitive power consumption exhibited in modern integrated circuits. Along with its benefits, however, NTC brings its own

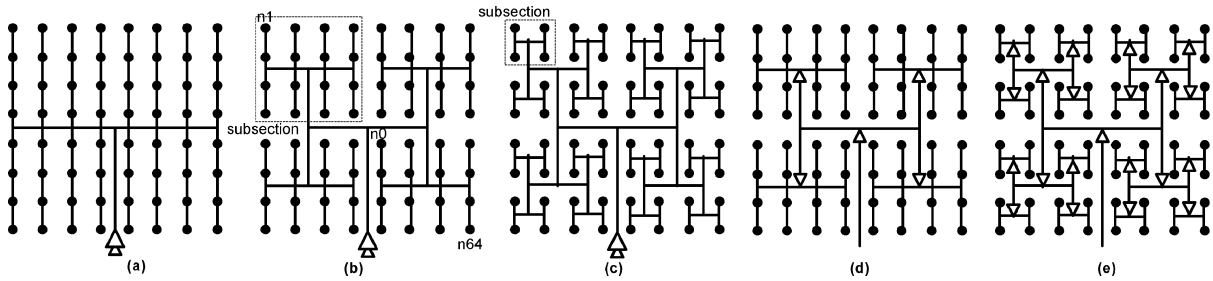

Fig. 16. Schematics of different clock networks used in benchmarks [10]. (a) Signal-route. (b) Unbuffered H-Tree (1 LV). (c) Unbuffered H-Tree (2 LV). (d) Buffered H-Tree (1 LV). (e) Buffered H-Tree (2 LV).

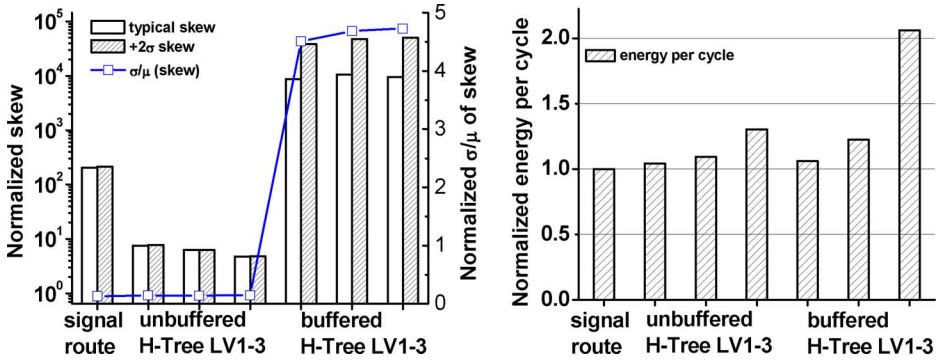

Fig. 17. Comparison of skew variability and energy consumption between clock networks [10].

set of problems. In this work, variability, one of the three major challenges arising in NTC, is reviewed. Several circuit and architecture solutions for addressing variability in logic, memory, and clock network are discussed. Alleviating variability through those techniques will help make NTC a reality, benefiting a wide spectrum of computing needs by solving the critical power problem.

REFERENCES

[1] R. Swanson and J. Meindl, "Ion-implanted complementary MOS transistors in low-voltage circuits," *IEEE J. Solid-State Circuits*, vol. SC–7, no. 2, pp. 146–153, Apr. 1972.

[2] Transmeta, "Transmeta Crusoe," [Online]. Available: http://www.transmeta.com/

[3] Intel, Hillsboro, OR, "Intel XScale," [Online]. Available: http://www.intel.com/design/intelxscale/

[4] IBM, "IBM PowerPC," [Online]. Available: http://www.chips.ibm.com/products/powerpc/

[5] B. Zhai *et al.*, "Theoretical and practical limits of dynamic voltage scaling," in *Proc. 41st Design Automation Conf.* , 2004, pp. 868–873.

[6] S. Hanson *et al.*, "Performance and variability optimization strategies in a sub-200 mV, 3.5 pJ/inst, 11 nW subthreshold processor," in *Symp. on VLSI Circuits*, 2007, pp. 152–153.

[7] S. Borkar *et al.*, "Parameter variations and impact on circuits and microarchitecture," in *ACM/IEEE Design Autom. Conf.*, 2003, pp. 338–343.

[8] M. Seok *et al.*, "The Phoenix processor: A 30 pW platform for sensor applications," in *IEEE Symp. on VLSI Circuits*, 2008, pp. 188–189.

[9] M. Wieckowski *et al.*, "Timing yield enhancement through soft edge flip-flop based design," in *IEEE Custom Integr.Circuts Conf. (CICC)*, Sep. 2008, pp. 543–546.

[10] M. Seok *et al.*, "Clock network design for ultra-low power applications," in *Int. Symp. on Low Power Electron. and Design*, 2010, pp. 271–276.

[11] N. Magen *et al.*, "Interconnect power dissipation in a microprocessor," in *Int. Workshop on SLIP*, 2004, pp. 327–333.

[12] J. Kwong *et al.*, "A 65 nm sub-$\mathrm{V}t$ microcontroller with integrated SRAM and switched-capacitor DC-DC converter," in *ISSCC*, 2008, pp. 318–616.

[13] *OpenCores*, [Online]. Available: http://www.opencores.org

[14] M. J. M. Pelgrom *et al.*, "Transistor matching in analog CMOS applications," in *Int. Technical Dig. Electron Devices Meeting (IEDM '98)*, Dec. 1998, pp. 915–918.

[15] B. Zhai *et al.*, "A sub-200 mV 6T SRAM in 130 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Feb. 2007, pp. 332–606.

[16] L. Chang *et al.*, "A 5.3 GHz 8T-SRAM with operation down to 0.41 V in 65 nm CMOS," in *IEEE Symp. on VLSI Circuits*, 2007, pp. 252–253.

[17] G. Chen *et al.*, "Crosshairs SRAM—An adaptive memory for mitigating parametric failures," in *Eur. Solid-State Circuits Conf.*, 2010, pp. 366–369.

[18] G. Chen *et al.*, "Yield-driven near-threshold SRAM design," in *ACM/IEEE Int. Conf. on Comput.-Aided Design*, 2007, pp. 660–666.

[19] B. Calhoun *et al.*, "Characterizing and modeling minimum energy operation for subthreshold circuits," in *Int. Symp. on Low Power Electron. and Design*, 2004, pp. 90–95.

[20] R. Dreslinski *et al.*, "Near-threshold computing: Reclaiming Moore's law through energy efficient integrated circuits," *Proc. IEEE*, no. 2, pp. 253–266, Feb. 2010.

[21] M.-E. Hwang *et al.*, "ABRM: Adaptive β-ratio modulation for process-tolerant ultra dynamic votlage scaling," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 18, no. 2, pp. ???–???, Feb. 2010.

[22] A. Wang *et al.*, "A 180 mV FFT processor using subthrehold circuit techniques," in *Int. Solid State Circuits Conf.*, Feb. 2004, pp. 292–529.

[23] S. R. Sridhara *et al.*, "Microwatt embedded processor platform for medical system-on-chip applications," in *Symp. on VLSI Circuits*, Jun. 2010, pp. 15–16.

[24] N. Verma *et al.*, "A 256 kb 65 nm 8T subthreshold SRAM employing sense-amplifier redundancy," *IEEE J. Solid-State Circuits*, vol. 43, no. 1, pp. 141–149, Jan. 2008.

[25] I.-J. Chang *et al.*, "A 32 kb 10T sub-threshold SRAM array with bit-interleaving and differential read scheme in 90 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 44, no. 2, pp. 650–658, Feb. 2009.

[26] J. P. Kulkarni *et al.*, "A 160 mV, fully differential, robust schmitt trigger based sub-threshold SRAM," in *Int. Symp. on Low Power Electron. and Designs*, Aug. 2007, pp. 171–176.

[27] J. Kil *et al.*, "A high-speed variation-tolerant interconnect technique for sub-threshold circuits using capacitive boosting," *IEEE Trans. Very Large Scale Integration (VLSI) Syst.*, vol. 16, no. 4, pp. 456–465, Apr. 2008.

[28] A. Goel *et al.*, "REad/Access-Preferred (REAP) SRAM -Architecture-aware bit cell design for improved yield and lower VMIN," in *IEEE Custom Integr. Circuits Conf.*, 2009, pp. 503–506.

[29] T. Kim *et al.*, "A voltage scalable 0.26 V, 64 kb 8T SRAM with V_{min} lowering techniques and deep sleep mode," *IEEE J. Solid-State Circuits*, vol. , no. 6, pp. 1785–1795, Jun. 2009.
[30] T. Kim *et al.*, "A 0.2 V 480 kb subthreshold SRAM with 1 k cells per bitline for ultra-low-voltage computing," *IEEE J. Solid-State Circuits*, no. 2, pp. 518–529, Feb. 2008.
[31] H. Qin *et al.*, "Error-tolerant SRAM design for ultra-low power standby operation," in *Int. Symp. on Quality Electron. Design*, Mar. 2008, pp. 30–34.

Mingoo Seok (S'05) received the Bachelors degree (*summa cum laude*) in electrical engineering from Seoul National University, South Korea, in 2005, and the Masters and Ph.D. degrees from University of Michigan in 2007 and 2011, respectively, all in electrical engineering.

He is currently a Member of Technical Staff in the Systems and Applications R&D Center of Texas Instruments Incorprated, Dallas, TX. He has published more than 25 journal and conference papers in his field of research, which includes low power circuit and system design methodologies for improving energy efficiency, performance, and variability both in digital and analog domains. He holds one pending US patent and two invention disclosures.

Dr. Seok received 1998–2001 Excellency Fellowship from Seoul National University, 1999 Distinguished Undergraduate Scholarship from the Korea Foundation for Advanced Studies, 2005 Doctoral Fellowship from the same organization, and 2008 Rackham Pre-Doctoral Fellowship from University of Michigan, Ann Arbor. He also won 2009 AMD/CICC Student Scholarship Award for picowatt voltage reference work and 2009 DAC/ISSCC Student Design Contest for the 35 pW sensor platform design (aka Phoenix Processor).

Gregory Chen (S '06) received the B.S. and M.S. degrees in electrical engineering from the University of Michigan in 2006 and 2009, respectively, where he is currently pursuing the Ph.D. degree in electrical engineering.

His research interests include power management for low-power systems, energy harvesting, and robust SRAM design.

Scott Hanson received the B.S., M.S., and Ph.D. degrees in electrical engineering from the University of Michigan, in 2004, 2006, and 2009, respectively.

He is the CEO and founder at Ambiq Micro. He led the development of Ambiq Micro's core technology at the University of Michigan. He speaks widely on energy-efficient circuits at trade-shows and conferences and has more than 25 publications on related topics.

Michael Wieckowski received the Ph.D. degree in electrical and computer engineering from the University of Rochester, NY, in 2007.

From 2007 until 2010, he worked as a research fellow at the University of Michigan, Ann Arbor. His research is focused on low-power mixed-signal design to enable energy constrained computing platforms and includes variation tolerant low-voltage memory, inductorless power management systems, and dynamically tuned low- voltage pipelines.

David Blaauw worked for Motorola, Inc. (now Freescale) in Austin, TX, for 8 years, where he was the manager of the High Performance Design Technology group. Since August 2001, he has been on the faculty at the University of Michigan where he is a Professor. His work has focussed on VLSI design with particular emphasis on ultra low power design and dynamic adaptivity to variations and reliability.

Dr. Blaauw was the Technical Program Chair and General Chair for the International Symposium on Low Power Electronic and Design. He was also the Technical Program Co-Chair of the ACM/IEEE Design Automation Conference and a member of the ISSCC Technical Program Committee.

Dennis Sylvester (S'95–M'00–SM'04–F'11) received the Ph.D. degree in electrical engineering from the University of California, Berkeley, where his dissertation was recognized with the David J. Sakrison Memorial Prize as the most outstanding research in the UC-Berkeley EECS Department.

He is a Professor of Electrical Engineering and Computer Science at the University of Michigan, Ann Arbor. He previously held research staff positions in the Advanced Technology Group of Synopsys, Mountain View, CA, Hewlett-Packard Laboratories, Palo Alto, CA, and a visiting professorship in Electrical and Computer Engineering at the National University of Singapore. He has published 290 articles along with one book and several book chapters in his field of research, which includes low-power circuit design and design automation techniques, design-for-manufacturability, and interconnect modeling. He holds 6 US patents. He also serves as a consultant and technical advisory board member for electronic design automation and semiconductor firms in these areas. He is co-founder of Ambiq Micro, a fabless semiconductor company developing ultra-low power mixed-signal solutions for compact wireless devices.

Dr. Sylvester received an NSF CAREER award, the Beatrice Winner Award at ISSCC, an IBM Faculty Award, an SRC Inventor Recognition Award, and eight best paper awards and nominations. He is the recipient of the ACM SIGDA Outstanding New Faculty Award and the University of Michigan Henry Russel Award for distinguished scholarship. He has served on the technical program committee of major design automation and circuit design conferences, the executive committee of the ACM/IEEE Design Automation Conference, and the steering committee of the ACM/IEEE International Symposium on Physical Design. He is currently an Associate Editor for IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS and previously served as Associate Editor for IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS. He is a member of ACM and Eta Kappa Nu.