Exploiting Read/Write Asymmetry to Achieve Opportunistic SRAM Voltage Switching in Dual-Supply Near-Threshold Processors

Yunfei Gu [1,*]**, Dengxue Yan** [2] **, Vaibhav Verma** [1] **, Pai Wang** [1] **, Mircea R. Stan** [1] **and Xuan Zhang** [2]

[1] Charles L. Brown Dept. of Electrical and Computer Engineering, University of Virginia, Charlottesville, VA 22904, USA; vv8dn@virginia.edu (V.M.); pw6eb@virginia.edu (P.W.); mrs8n@virginia.edu (M.S.)

[2] The Preson M. Green Dept. of Electrical and System Engineering, Washington University in St. Louis, St. Louis, MO, 63130, USA; dengxue.yan@wustl.edu (D.Y.); xuan.zhang@wustl.edu (X.Z.)

* Correspondence: yg3eb@virginia.edu

Abstract: Energy-efficient microprocessors are essential for a wide range of applications. While near-threshold computing is a promising technique to improve energy efficiency, optimal supply demands from logic core and on-chip memory are conflicting. In this paper, we perform static reliability analysis of 6T SRAM and discover the variance among different sizing configuration and asymmetric minimum voltage requirements between read and write operations. We leverage this asymmetric property in near-threshold processors equipped with voltage boosting capability by proposing an opportunistic dual-supply switching scheme with a write aggregation buffer. Our results show that proposed technique improves energy efficiency by more than 21.45% with approximate 10.19% performance speed-up.

Keywords: Near Threshold Computing (NTC); Dual-Supply; Static Random Access Memory (SRAM); Reliability; Write Aggregation Buffer

1. Introduction

Improving energy efficiency has become one of the primary objectives of current microprocessor design. Applications ranging from low-emission green data centers to ubiquitous low-power Internet-of-Things (IoT) devices rely on energy- efficient microprocessors. NTC emerges as a promising technique to improve energy efficiency by lowering the supply voltage to near-threshold levels [1]. However, due to the key differences in their circuit structures and activity levels, the logic core and the memory blocks in a microprocessor often demand distinctive supply voltages to optimize their energy efficiency and guarantee the robust reliability respectively. For systems with a single digital supply voltage, compromise has to be made between the different demands of the core and the memory system, resulting in sub-optimal energy efficiency at the full system level. In reality, for NTC systems with only a single digital supply (V_{DD}), the supply voltage level is often dictated by the minimum voltage (V_{min}) to ensure reliable operation of its memory blocks. This requires the system to operate at a higher supply voltage than the optimal supply for the logic core, and results in sub-optimal system-level energy efficiency. On the other hand, to address a major limitation of NTC—its severely-degraded single-thread performance due to the increased critical path delay at lower supply voltages, dual-supply architecture has been proposed to allow temporary voltage boosting during the execution of critical difficult-to-parallelize sequential sections in single core or heterogeneous multi-core processor design. In this paper, we propose a novel opportunistic method to further exploit energy-efficiency in the context of a dual-supply NTC system.

Our in-depth analysis of memory reliability based on circuit-level simulation of typical six-transistor static random access memory (6T-SRAM) cells in 7nm and 16nm FinFET Technology processes reveals that read and write operations exhibit asymmetric behaviors at near-threshold voltages. This discovery suggests that it is possible to operate memory read and write at different

voltage levels without incurring significant performance and reliability penalty. Leveraging this asymmetric reliability behavior, we are able to achieve better energy efficiency by lowering the memory supply during read operations and switching it back to the higher nominal voltage during write operations. In order to reduce the overall performance overhead of our proposed opportunistic supply switching, a write aggregation scheme [2] is developed to augment conventional multi-level cache architecture. Finally, we evaluated this work using diverse benchmark suites, including SPEC2006 [3] and PARSEC [4] benchmarks. our method show 21.45% improvement compared to the baseline, where a single fixed near-threshold voltage is used as the supply.

2. Background and Motivation

Our proposed method stems from several important properties of near-threshold processors and is motivated to address one of its fundamental limitations. In this section, we provide an overview and background discussions on energy efficiency, memory reliability, and dual-supply architecture in near-threshold processors.

2.1. Energy Efficiency of NTC

The goal of NTC is to find the supply voltage that can deliver peak energy efficiency in a computing system. Such energy efficiency improvement is desirable for a variety of applications, such as battery-powered smart phones and embedded systems, as well as data centers that pay hefty electricity bills to power their servers. Since energy is measured by the product of power consumption and execution time, both of which are a function of the supply voltage, there should exist an optimal voltage that minimizes processor energy. Generally speaking, power consumption decreases monotonically with lower supply voltage, whereas the minimum critical path delay increases monotonically.

Studies have found that the optimal supply voltage resides just above the threshold voltage of the transistor [5], and it has since been experimentally proven by many silicon prototypes [6,7]. To gain a more intuitive understanding of NTC, it helps to breakdown the entire power consumption into the dynamic power and the static power. The former is due to the charge and discharge of the logic gates from switching and can be captured on the first order by a quadratic function as $\alpha C_{eff} V_{DD}^2 f_{clk}$, where α is the switching activity factor, C_{eff} represents the effective intrinsic capacitance of the logic circuits and f_{clk} the clock frequency. The latter consists of mostly leakage power that equals $I_{lkg} V_{DD}$, where I_{lkg} represents the static leakage current.

The dynamic part of the energy always scales down quadratically with lower supply voltage and is independent of the operating frequency,whereas the static part can increase sharply as the circuit delay rises at lower supply voltage. That is why the optimal supply is reached slightly above the threshold voltage (V_{th}): when the supply is too far above V_{th}, dynamic energy dominates and energy efficiency worsens; when the supply is below V_{th}, static energy dominates due to long critical path delay, and again energy efficiency suffers.

Besides, since the optimal supply voltage (V_{opt}) is determined by the balance between dynamic and static energy components, computational units with different dynamic/static breakdown could have distinctive V_{opt}. Zhai et al. [8] showed that SRAMs, commonly used for caches, have a higher V_{opt} than processors, by approximately $100mV$. It is caused by the relatively high leakage component of cache energy, a trade-off associated with their large size and high density. Our normalized energy simulation in Figure 1 illustrates this effect using device parameters from 16nm FinFET technology process [9] and activity parameters from Gem5/McPAT simulator [10,11]. If we normalize the energy consumption of the logic core and the memory blocks at nominal supply voltage ($850mV$ in the 16nm FinFET process), we can obtain the energy scaling trend as a function of the supply voltage [12]. Apparently, the core energy minimizes at V_{opt}^{core} around $450mV$, whereas memory energy minimizes at V_{opt}^{mem} around $600mV$. This result also indicates that in a NTC system where only a single supply

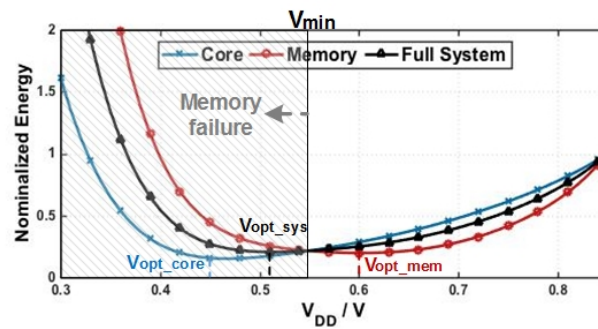



Figure 1. Normalized energy of the logic core, the memory, and the full processor system as a function of the supply voltage (V_{DD}) in a 16nm process.

voltage is available, both the core and the memory have to compromise to yield a unique optimal voltage (V_{opt}^{sys}) for the full system as indicated in Figure 1.

As leakage increases with respect to switching energy, it becomes more efficient to run faster, hence V_{min} is shifted higher. Usually, SRAM cache can run with optimal energy efficiency at a higher speed than its surrounding logic [13]. The minimum operating energy is achieved at the point where the switching and leakage components are balanced. At voltages higher than the NTC operating point, switching energy dominates. Below that, leakage energy dominates. Thus the optimum operating point is usually slightly above the threshold voltage [14].

2.2. Memory Reliability in NTC

Another critical issue that limits NTC from achieving the optimal energy efficiency is its degraded reliability. Compared to the super-threshold operation at the nominal supply voltage, near-threshold operation is more susceptible to process variation[15,16], supply noise [12], and temperature fluctuations [17] (PVT), especially for the memory blocks in NTC, due to their minimum-sized devices and large array numbers. As indicated in Figure 1, the minimum voltage that allows reliable memory operation (V_{min}) could exceed V_{opt}^{sys} and often dictates the actual supply voltage [15,18], pushing the NTC system further away from its optimal efficiency point.

For NTC operation, 8T SRAM has been proposed as an alternative more reliable memory cell structure than 6T SRAM, but it involves trade-offs between power, area and performance. Although 6T SRAM cell is indeed more vulnerable due to mismatches between the transistors in the cross-coupled inverters [19], it can be designed to overcome its vulnerability and work reliably in NTC with voltage boosting topology. 6T SRAM cell also leaks at least 4.5% less, and a single cell is over 30% smaller in area than 8T SRAM cell, which indicates that 6T SRAM cell remains an appealing choice for a NTC with voltage boosting SoC [20].

2.3. Dual-Supply Architecture for Voltage Boosting

The underlying premise of NTC is that workloads can be effectively parallelized so that operating large number of energy-efficient NTC cores can deliver higher throughput at the same power budget, thus overcoming the curse of "Dark Silicon" [21]. However, single thread performance is needed in some cases to overcome "Amdahl bottlenecks", such as inherently serial code regions, lock contention, communication overheads, and long-tail latency distribution in servers. To mitigate the severe negative impact of single-thread performance in NTC, the implementation of dual supply rails have been proposed to temporarily boost the supply and the operating frequency of NTC processors in 10's of clock cycles [16] **.**

Several key facts have emerged from earlier development of NTC techniques so far: (1) energy efficiency compromise exists in single-supply NTC systems due to different V_{opt} for core and memory; (2) memory reliability limits further lowering of the supply voltage to achieve energy optimality; (3)

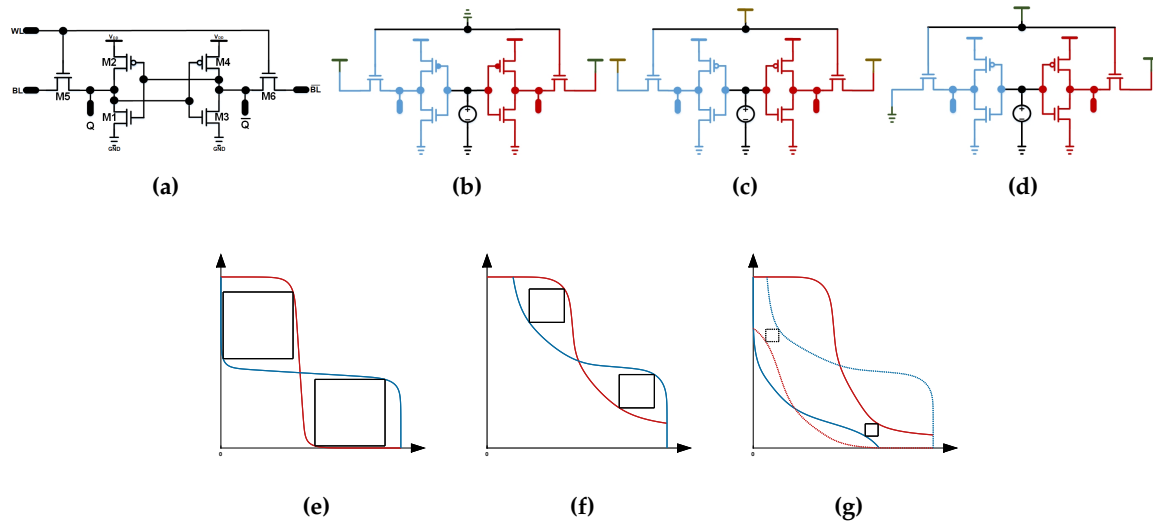



Figure 2. (a) Circuit schematic of a 6T SRAM cell; (b)-(d) test circuits to measure HSNM, RSNM, and WSNM-Write 0; (e)-(g) conceptual butterfly curves for HSNM, RSNM, and WSNM.

dual-supply architecture addresses a fundamental shortcoming of NTC's single-thread performance through voltage boosting.

In our paper, we aim to achieve better energy efficiency in NTC processors by opportunistically tapping the wasted energy due to the single-supply and memory V_{min} constraints. Our topology takes advantage of the dual-supply architecture originally proposed for voltage boosting and extends it to gain additional energy savings.

3. Asymmetric Memory Reliability

The critical role that memory reliability plays in NTC systems demands an in-depth examination of the underlying mechanisms contributing to memory failures at near-threshold voltages.

The most common memory structure used in on-chip caches is a 6T SRAM cell which is presented in Figure 2(a). It consists of a cross-coupled inverter pair and two access transistors. The inverter pair that contains pull-down (PD) and pull-up (PU) transistor is used to store bit 0 or 1 through positive feedback, and the access transistor are controlled by the wordline (WL) to connect the internally-store bit value to the bitline (BL). Typical read and write operations of the SRAM array follow different well-defined control signal sequences [22]. During a read access, BL and $\overline{BL}$ are first pre-charged to V$_{DD}$. Then WL goes high, so that Q and $\overline{Q}$ drive data through the access transistors (PG) M5 and M6 to split up the BL and $\overline{BL}$, and the voltage difference between BL and $\overline{BL}$ is sensed by a sense amplifier (SA) to resolve the read data. During a write access, BL and $\overline{BL}$ are driven to the complementary input data by a write driver, and WL turns on PG to drive the data into the internal storage nodes. Due to the read and write coupling, the sizing of PG, PD and PU is sensitive for reliability property.

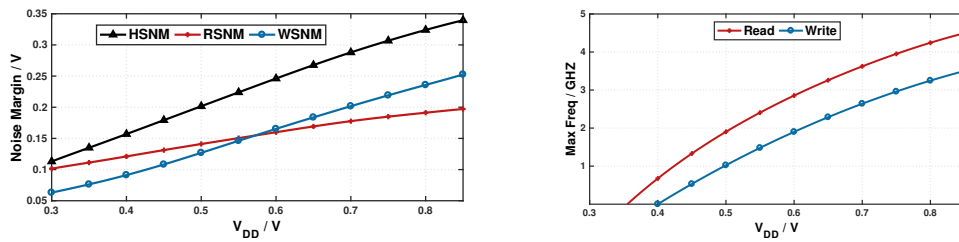
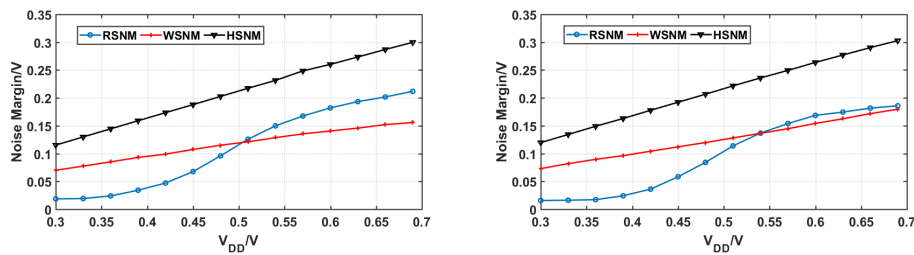
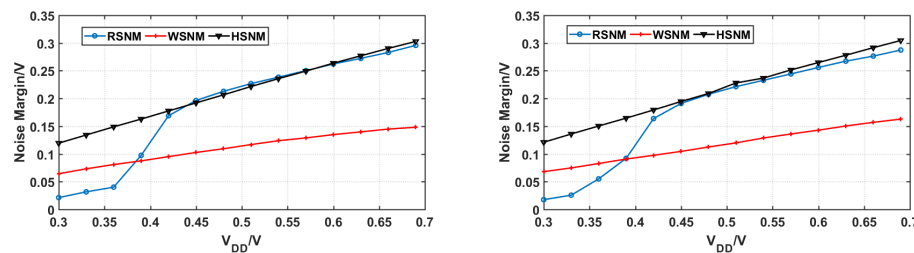
In order to evaluate the reliability property of 6T SRAM cell under low supply voltages, we first look at its static noise margin (SNM) as a function of supply. The SNM criteria are widely used to characterize the static stability of SRAM cells. The static noise margins for hold (HSNM), read (RSNM), and write (WSNM) can be obtained by overlapping the voltage transfer curves (VTC) of the cross-coupled inverters within a cell, often called the butterfly curve [23]. The test circuits to measure these SNM metrics are shown in Figure 2(b)-(d), where the cross-coupled feedback loop is intentionally broken to simulate different biasing conditions during hold, read, and write operations. The resulting butterfly curves are conceptually illustrated in Figure 2(e)-(g). The noise margins are extracted from the butterfly curve as the diagonal distance of the biggest square that can fit inside the butterfly curve. We can analyze the shapes of the butterfly curves. For example, during the read operation, BL and $\overline{BL}$ are pre-charged to V_{DD}, which pull the VTC higher and cause it to move slowly to the bit-0 low voltage

(a) HSNM, RSNM, and WSNM vs. supply voltage. **(b)** Max frequency for read and write vs. supply voltage.

Figure 3. 6T SRAM characteristics in 16nm technology.

(a) HSNM, RSNM, and WSNM vs. supply voltage for 1:1:1 SRAM cells.

(b) HSNM, RSNM, and WSNM vs. supply voltage for 1:1:2 SRAM cells.

(c) HSNM, RSNM, and WSNM vs. supply voltage for 1:2:2 SRAM cells.

(d) HSNM, RSNM, and WSNM vs. supply voltage for 1:2:3 SRAM cells.

Figure 4. 6T SRAM SNM characteristics in ASAP 7nm PDK.

level as in Figure 2(f) for read, compared to the curves in Figure 2(e) for hold. Similarly, during write, one side of the bitlines is driven to ground, the other to V_{DD}. As a result, one of the VTC corresponding to the ground side switches to low voltage level sharply as shown in Figure 2(g).

To obtain a quantitative measurement of the SNMs across a wide supply range, we sweep the supply voltage from $0.3V$ to $0.85V$, the nominal supply voltage of the 16nm predictive technology model(PTM) using customized sizing configuration. Figure 3(a) presents the different noise margins as a function of the supply voltage. Interestingly, read stability and write stability show opposite behaviors at different end of the voltage spectrum: when the supply voltage is high, WSNM is higher than the RSNM; while at lower supply voltage, the roles are flipped with RSNM being higher than WSNM. Our results corroborate with previous work on write margin analysis [24], and hence we hypothesize that write operation fails earlier than read when we lower the supply voltage. Figure 3(b) shows that read operation indeed outlasts write, and it is possible to reliably read from the SRAM cell at near-threshold voltages where write would have already failed. Hence if we could switch power supply to a lower voltage for SRAM read operation, a significant power will be saved while maintaining reliable operation. Similar results for different voltages for read and write SRAM reliability have also been observed experimentally in 14nm devices [25], where read failure rate is reported to be lower and so is its minimum error-free voltage.

Additionally, we validate the PG, PU, PD transistor sizing impact on static reliability characteristic for SRAM cells in deeply scaled technology, 7nm ASAP PDK [26]. We sweep the supply voltage from $0.3V$ to $0.7V$ with several popular standardized SRAM cell ratios. The SRAMs implemented in our test have the fin-ratios (PU:PG:PD) of 1:1:1, 1:1:2, 1:2:2 and 1:2:3. With the variance of fin-ratios, the static noise margins change as shown in Figure 4(a)-(d). The RSNM and WSNM will always flip while in the different voltage domain. Interestingly, as the PG ratio increasing, the intersection between RSNM and WSNM moves to sub-threshold voltage domain and the upper side of it approaches to RSNM. Associated with above discussion, the 1:1:1 and 1:1:2 SRAM cells in 7nm process may retain the similar static reliability property and supply voltage demands for read and write separately. In conclusion, we believe this asymmetric reliability property could open new opportunities to improve energy efficiency of NTC processors even in high-end deeply scaled technology process.

4. Opportunistic Dual-Supply Switching

Our circuit-level analysis of the SRAM reliability in Section 3 reveals new energy saving opportunity by leveraging the asymmetric read/write behaviors —if we can separate the read operations from the write operations and set the memory block to lower supply voltages during read-only transactions, the processor could potentially more closely approach its optimal supply voltage that minimizes full-system energy efficiency, especially in a NTC processor as illustrated in Figure 1. Moreover, it may be possible to take advantage of the dual-supply architecture that has already been proposed for boosting NTC's degraded single-thread performance [16] by connecting the memory blocks to dual supply rails and alternating their supply voltages based on transitions between read and write instruction sequences. This forms the basic idea behind our proposed opportunistic dual-supply switching scheme for the memory systems in NTC processors. In this section, we analyzed cache behavior characterization of a usual general purpose computational-intensive workload SPEC2006 and data-intensive workload PARSEC, in order to leverage the opportunistic improvement potential upon the read/write intensity in full system simulation. We then discuss the design considerations and the implementation details of our method.

4.1. Memory Behavior Characterization of Workload SPEC2006

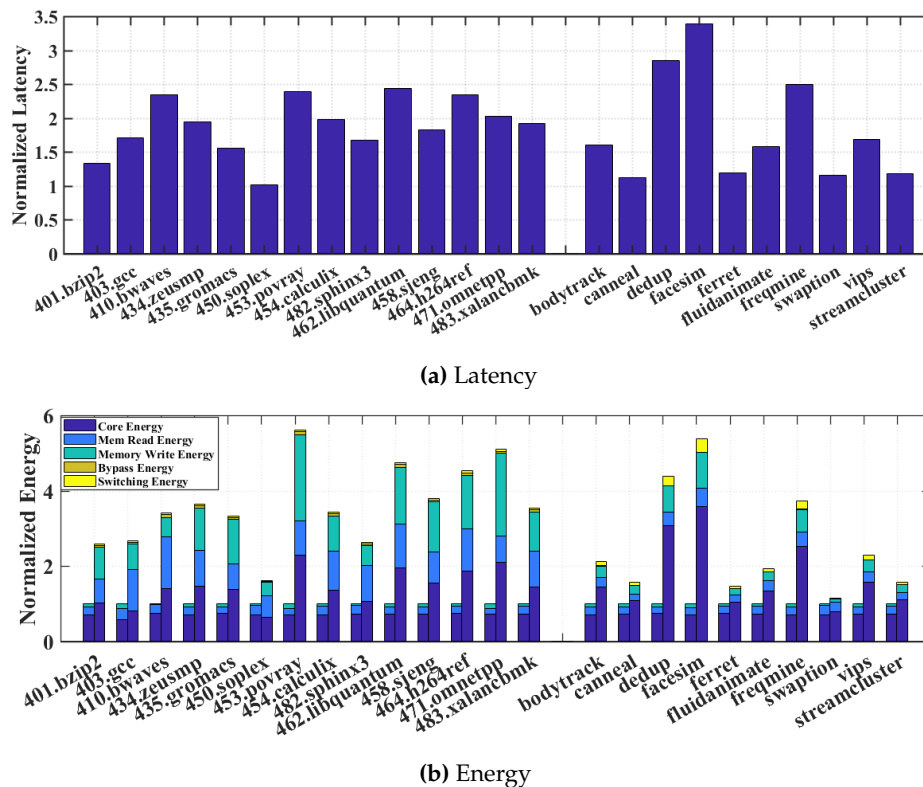
According to the dual-supply switching scheme discussed above, the read and write instruction sequences are crucial, because the ratio of read operations and write operations and their orders would influence the switching behavior and energy reduction opportunity. The larger percentage of read operations over total memory behaviors means that there will be more chances to lower the voltage supply to V_L to decrease the energy consumption. To investigate such memory behavior characteristics, we selected diverse general-purpose benchmark SPEC2006 and PARSEC, which are computation-intensive and data-intensive respectively. In our study, we used Gem5 cache trace probe to analyze the data request between processor and cache. Several selected representative workloads

Figure 5. Read and write operations ratio distribution among cache traces of different SPEC2006 and PARSEC benchmarks.

from the SPEC2006 suites are evaluated on X86 system with a three-level cache hierarchy: 64kB L1 cache, 256kB L2 cache and 2MB last level cache. Figure 5 shows the ratio distribution between total read and write operations. Interestingly, the read operations of most workloads among both SPEC2006 and PARSEC take more than 70% of all memory operations, which illustrates that it has a promising potential to let our switching architecture dominates the voltage supply with V_L instead of V_H.

4.2. Naive Greedy Switching

In a system with dual supply rails, each component can be connected to either rail via power switches that are controlled by enabling/disabling signals. However, switching between these supply rails incurs non-negligible latency and energy costs. For example, according to one such dual-rail application switching between $400mV$ and $600mV$ in 32nm process [16], the rise time is approximately 9ns and the fall time 7ns. Therefore, overheads have to be accounted for when evaluating the potential benefits of dual-supply switching for the memory blocks in NTC.

Let us first evaluate a naive approach of greedy switching—the memory supply is set to $V_L = 450mV$ whenever there is a read transaction and to $V_H = 850mV$ whenever there is a write transaction. We pick these voltage levels because V_L is close to V_{opt}^{core} in this technology process, and V_H is the nominal supply voltage. We use the NTC configuration with a single supply voltage at $V_{opt}^{sys} = 510mV$ as the baseline case for comparison. We use the transition delay numbers from previously published dual-supply architecture for our estimation [16]. Figure 6(a) reveals that execution latency can rise significantly due to the additional transition time of the supply switching. The latency increase can be as much as 60%-70% for most SPEC2006 and PARSEC benchmarks. In addition to the latency penalty, Figure 6(b) indicates that naive greedy switching results in significant energy overheads on top of the normal energy consumption broken down for core, memory read, and memory write.

(a) Latency

(b) Energy

Figure 6. The normalized total energy and latency breakdown of the naive greedy switching scheme across benchmarks in SPEC2006 and PARSEC.

Several sources contribute to these overheads: the core and memory energy overheads are due to increased leakage while the memory waits the rise time from V_L to V_H; and the power switch overhead refers to the additional energy lost due to the heat dissipation on the parasitic resistance of the power switch. It is worth pointing out that in the naive greedy switching scheme, the reason that memory read and write energies increase compared to the baseline is because we account for the fact that memory read now happens at $500MHz@450mV$ and write at $500MHz@850mV$, compared to the NTC baseline where both operate at $500MHz@510mV$. This also explains why the core energy decreases since in the naive greedy switching, it operates at V_{opt}^{core}.

The overwhelming latency and energy penalty suggest that the naive greedy approach of opportunistically switching the memory supply to V_L during read transactions and to V_H during write transactions is prohibitively costly, thus defeating the original purpose of improving system energy efficiency. To find a more practical approach, we analyze the detailed penalty distribution by comparing different benchmarks. We observe that in benchmarks that exhibit smaller latency and energy penalty, such as 401.bzip, 450.soplex, and 482.sphinx3, there are much fewer supply switching activities. Since every switching incurs a fixed one-time latency and energy cost due to transition delay, ideally a long consecutive read sequence is preferred to amortize the cost and reap more energy benefits from staying in V_L that minimizes core energy per operation.

4.3. Long Consecutive Read Opportunity

Based on our study of the naive greedy switching overhead, it is clear that to maximize energy saving opportunity using dual-supply switching, we need long consecutive read sequences which are not interrupted by write operation in the memory transaction trace. In order to visualize and identify such opportunity from the memory transaction traces, we propose a quantitative measure using the cumulative distribution function (CDF) of consecutive read sequences to represent the long consecutive read sequence potential (LCRSP) in different benchmarks, as defined by Equation 1.

$$F_S(s) = P(S \le s) = \frac{\sum_n s \times t}{r} \times 100\% \qquad (1)$$

where $F_S(s)$ is the cumulative probability function of consecutive read sequence the represents the entire read operations released from CPU or higher level cache when the benchmark is executed; n represents the set number of read sequence groups with unique consecutive read sizes; s equals to $log_{10}(unique\ consecutive\ read\ size)$; t is the number of occurrence of sequences with each unique consecutive read size; and r is the total number of read operations.

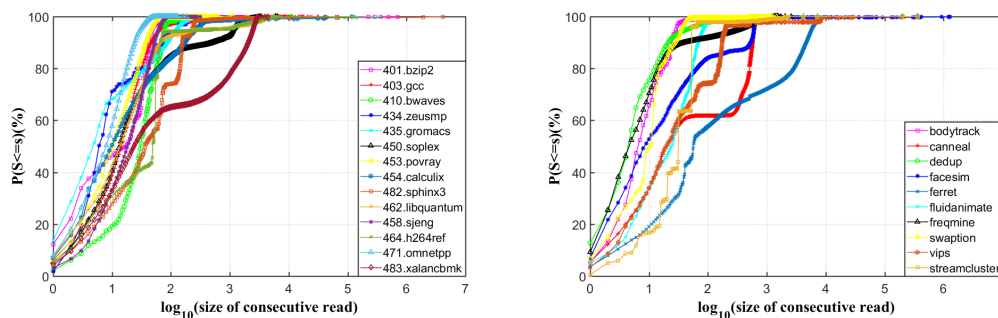
Using the definition in Equation 1, we can plot the CDFs of consecutive read sequences generated by executing different SPEC2006 and PARSEC benchmarks, as shown in Figure 7(a)-(b). If we look at three characteristic points where the CDF crosses 20%, 50%, and 80% probability, we find that

(a) Selective SPEC2006 benchmarks Emulation   **(b)** Selective PARSEC benchmarks Emulation

Figure 7. CDF graphs represent distribution of consecutive read sequences among cache traces of different SPEC2006 and PARSEC benchmarks.

the greater the value of the horizontal axis corresponding to these percentages, the higher LCRSP the benchmark has according to its cache trace. For example, in 483.xalancbmk shows, these three CDF characteristic points are approximately $F(s = 10^{0.602}) = 18.718\%$, $F(s = 10^{1.38}) = 50.353\%$ and $F(s = 10^{3.062}) = 79.99\%$, which means 79.99% of the trace has a consecutive read size of $10^{3.062}$ or smaller. It suggests that 483.xalancbmk has many long consecutive read sequences, therefore is a good candidate for dual-supply switching. Conversely, similar measure would suggest that the cache trace of 435.gromacs's has a poor portion of long reads, aggressively switch the dual-supply power will bring unexpected cost. Figure 7(a)-(b) indicate SPEC2006 and PARSEC benchmarks are diverse and the opportunity to save energy is unevenly distributed. Interestingly, we note that PARSEC benchmarks generally show higher potential thanks to the longer consecutive read sequences in PARSEC's data-intensive workloads.

4.4. Write Aggregation Buffer (WAB)

Since the read-to-write transition delay is the culprit that dominantly contributes to the latency and energy overheads shown in Figure 7(a) and Figure 7(b), we need to improve the LCRSP of different benchmarks by reducing the number of read-to-write transitions. This is equivalent to reducing the number of write operations interrupting the long consecutive read sequence. Therefore, we propose a micro-architecture unit called write aggregation buffer (WAB) and Write back aggregation buffer (WBAB) to accumulate write operations and stop releasing them frequently from disturbing consecutive reads.

The micro-architecture structure of our proposed WAB and WBAB is based on write buffer design used in cache architecture of conventional CPU systems like Intel Xeon and AMD64 . In conventional processor, a write buffer can hold the data temporarily and stop it being written from the higher-level cache (for write backward) or to the next-level cache or the main memory (for write forward). It allows the lower-level cache to service the subsequent read requests without waiting for long write transactions to complete at slow main memory. We enhance the conventional write buffer design with aggregation functionality to accumulate write transactions together to a specific length and release multiple write requests to the global cache access sequence all at once. During the WAB releasing period, read transactions still have higher priority. When a read request is issued to the global access sequence, the write release operation at the WAB would halt and wait for reads to be completed. Actually, this policy increases the duration of write-level power switch-on, but reduces the supply switching activities in disguise.

The system diagram in Figure 8 illustrates how our proposed WAB structures are inserted at the L1 and L2 cache hierarchy in a NTC processor with dual supply rails. The components highlighted

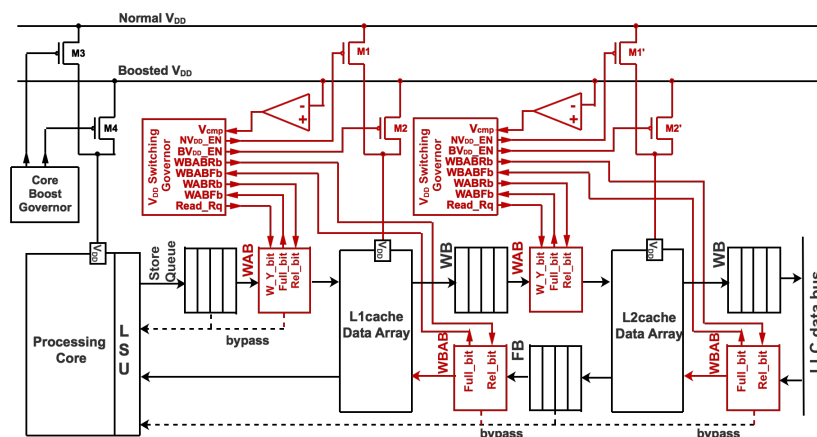



Figure 8. System Block Diagram

Algorithm 1 V_{DD} Switching Governor Algorithm

Input: *WABFb*, *WBABFb* and *V_{cmp}*
Ouput: *NVDD_EN*, *NVDD_EN*, *WBABRb*, *Read_Rq* and *WABRb*
 1: *WABFb = WAB_Full_bit*
 2: *WBABFb = WBAB_Full_bit*
 3: *WABRb = WAB_Rel_bit*
 4: *WBABFb = WBAB_Rel_bit*
 5: *W_Y_bit = Read_Rq* // Read request bit
 6: **if** Request read **then**
 7:     *Read_Rq* ← 1 // set *W_Y_bit* to 1
 8:     *WAB Write halt and wait for reads*
 9: **else**
10:     *Read_Rq* ← 1 // set *W_Y_bit* to 0
11:     **if** *WAB_Full_bit* == 1 | *WBAB_Full_bit* == 1 **then**
12:         *NVDD_EN* ← 1 // set *NVDD_EN* to 1
13:         *BVDD_EN* ← 0 // reset *BVDD_EN* to 0
14:     **else**
15:         *NVDD_EN* ← 0 // set *NVDD_EN* to 0
16:         *BVDD_EN* ← 1 // reset *BVDD_EN* to 1
17:     **end if**
18:     **if** *V_{cmp}* == 1 **then**
19:         **if** *WAB_Full_bit* == 1 **then**
20:             *WAB_Rel_bit* ← 1 // WAB release aggregated write data chuck by once
21:         **end if**
22:         **if** *WBAB_Full_bit* == 1 **then**
23:             *WBAB_Rel_bit* ← 1 // WBAB release aggregated write data chuck by once
24:         **end if**
25:     **else**
26:         *WAB_Rel_bit* ← 0
27:         *WBAB_Rel_bit* ← 0
28:         // Hold aggregated write data chuck in WAB or WBAB
29:     **end if**
30: **end if**

---

in red are the WABs and their associated auxiliary circuits to enable the opportunistic dual-supply switching. Inserted between Write Buffer (WB) and L1 D-cache is one WAB that aggregates the write request from the processor core side. Another WAB, labeled as Write Back Aggregation Buffer (WBAB), is inserted between Write Back Buffer and Fill Buffer (FB), and it accumulates the write back requests from lower level cache side. However, WBAB does not have to yield to read transactions because of the CPU scheduling priority. Accompanied these buffers are auxiliary circuitry including power switches, voltage comparators, and dual-supply switching controllers.

The above pseudo code algorithm 1 and the illustration of an example R/W access sequence execution re-ordered by WAB in Figure 9 explains how the dual-supply switching controller works with the WAB and WBAB. When WAB/WBAB is not filled, the read requests are released to the cache R/W access sequence directly and the write requests are forwarded to WAB/WBAB. Meanwhile, the bypass structure responses directly to the read request, so direct access to the cache is not needed for a read after write (RAW) data access. Once WAB/WBAB reaches a fixed buffer length of aggregated

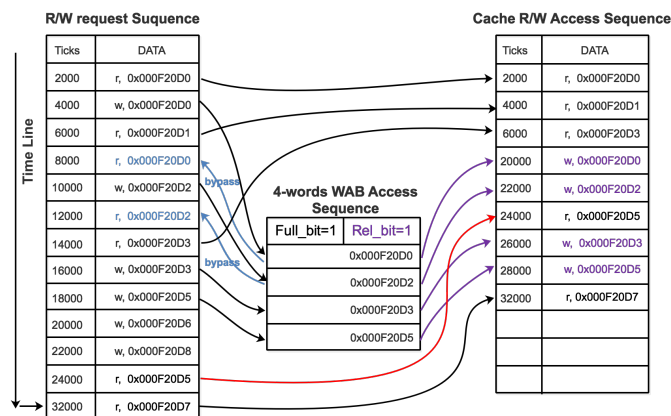



Figure 9. Example R/W access sequence execution re-ordered by WAB

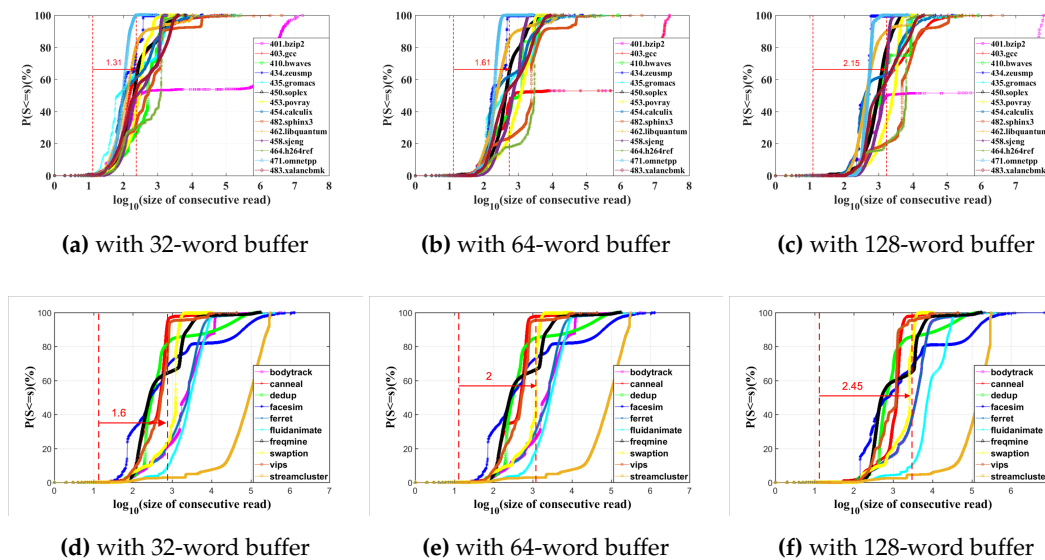



Figure 10. LCRSP improvement by varying buffer sizes using the cache traces of different SPEC2006 and PARSEC benchmarks

write operations, the *WAB_Full_bit* or *WBAB_Full_bit* will be set up to 1, indicating WAB/WBAB are filled. Followed by this, the dual-supply power switches are activated. The transistor M1 will be turned off and M2 turned on. During V_L to V_H transition (from $450mV$ to $850mV$ in our example), WAB/WBAB cannot release aggregated write operations to the global cache access sequence, since reliable write is not guaranteed in that voltage transition interval and corresponding latency may incur. We place a voltage comparator to determine whether V_{DD} reaches a safe level for write. When that condition is met, V_{cmp} will turn to 1 and set the *WAB_Rel_bit* or *WBAB_Rel_bit*, which in turn activates the write release operation at WAB and WBAB. If WAB and WBAB release their aggregated write operations simultaneously, they should follow the CPU tick schedule to manage the order of each single write operation in the aggregated write sequence. Please note that by setting V_H at a high voltage level of $850mV$, our proposed system architecture is fully compatible with voltage boosting for NTC processors, and allows our switching scheme to run concurrently with voltage boosting if needed.

4.5. WAB and WBAB Enabled Switching

To examine the effect of WAB and WBAB to the distribution of read sequences, we compare the CDF curves of different benchmark cache traces with and without WAB and WBAB, as well as with varying WAB and WBAB sizes. Figure 10 presents the LCRSP improvement by implementing various size WAB and WBAB among the cache traces of diverse SPEC2006 and PARSEC benchmarks.

According to the approximate mean value of X axis which means $\log_{10}$ *(unique consecutive read size)*, corresponding to characteristics points 50%, we can see that after implementing WAB and WBAB, all of their CDFs experience a right shift. The abscissa of the averaged characteristic points are increased by one to two roughly. These results suggest that our proposed WAB and WBAB structure is able to universally improve the characteristics of LCRSP across diverse benchmarks that could result in $10\times$ to roughly $1000\times$ longer read sequence for our opportunistic dual-supply switching scheme to exploit further energy savings. when we observe the LCRSP promotions of SPEC2006.401.bzip2, PARSEC.fluidanimate and PARSEC.streamcluster workload by increasing the WAB size, we discover that there should be a saturation border for each workload's LCRSP improvement. Since The LCRSP curves of these three workloads mentioned above almost remain unchanged by implementing variant size WAB.

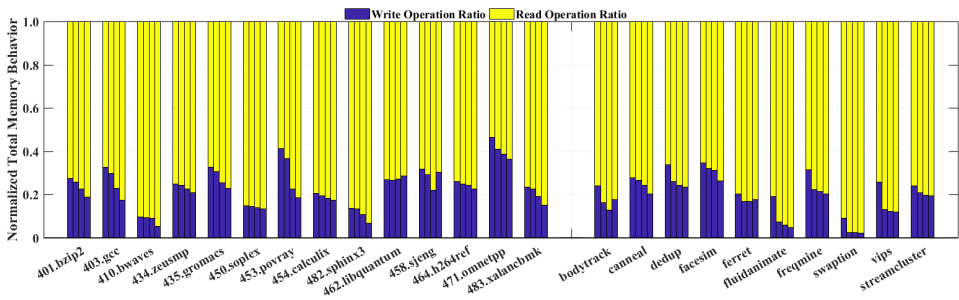



Figure 11. Cache executed read and write operations ratio distribution variance with different size among the cache traces of different SPEC2006 and PARSEC benchmarks

Meanwhile, the read and write operations that are executed by cache will also be decreased by our WAB/WBAB because of the bypass mechanism. The WAB/WBAB plays the crucial role aggregating the write operations together to the consecutive chunk as the releasing candidate and acts as the enlarged write buffer or write back buffer which are already existed in current CPU architecture as well. The immediate reuse data will not go into cache directly, but be executed in WAB/WBAB instead. Though the entire memory behavior number still stays fixed because of the constant instructions, the cache executed read and write ratio that will change and provide more opportunities to switch the voltage supply to V_L. Figure 11 shows the normalized read and write operations ratio variance by implementing 32-word, 64-word and 128-word WAB and WBAB from left to right.

5. Experimental Evaluation

In this section, we set up a simulation framework to evaluate and verify the energy efficiency and performance overhead of our proposed dual-supply switching technique in NTC processors.

5.1. System Configuration

We configure the processor parameters according to Table 1 with 64kB L1-cache (32kB I-cache and 32kB D-cache), 256kB L2cache, and 2MB LLC on a X86 platform. We use Gem5 [10] as the architecture performance simulator to generate the cache traces and McPAT [11] to generate the corresponding power traces for a collection of benchmarks in SPEC2006 and PARSEC. Since McPAT does not scale to the low voltage levels used in our system, we use circuit-level simulation results from Cadence Virtuoso based on a 16nm FinFET device PTM [9] model to scale the frequency, dynamic, and leakage powers. Memory read and write energy are derived from Cacti [27] power and frequency analysis with different voltage scaling techniques using Cadence and Cacti. The power, latency, and area of

Table 1. System Configuration

Platform Architecture	X86
Temperature	380K
Clock Frequency	500MHZ
Feature Size	16 nm
Functional Units	ALU 6, MUL 1, FPU 2
Branch Prediction	2KB BTB, 16-entry RAS
Fetch	16-entry buffer, Min. 8 cycles fetch-dispatch time
L1 D-cache	32kB, 8 assoc, 2 cycle hit latency
L1 I-cache	32kB, 4 assoc, 2 cycle hit latency
L2 cache	Private 256kB, 8 assoc, 6 cycle access latency
L3 cache	shared, 2MB, 16 assoc, 14 cycle access latency
Coherence Protocol	MESI (Modified, Exclusive, Shared, Invalid)
Main Memory	2GB DDR3-1066

combinational logic such as the dual-supply controller is based on synthesized results from Synopsys VCS and Design Compiler.

5.2. Energy Efficiency Improvement

Using the NTC processor of the same configuration with only a single fixed supply voltage at $V_{opt}^{sys} = 510mV$ as our baseline example for comparison, we evaluate the energy efficiency of our proposed dual-supply switching technique. Compared to the NTC baseline, we improve the energy efficiency by implementing the WAB and WBAB with dual-supply in the system. Actually, the WAB and WBAB structures with bypass mechanism have ability to reduce the total number of read and write cache access time and response to the read requests from bypass instead. As bypass could be accessed faster and consume much less energy, it provides us with large energy and latency benefits. From Figure 12(a), we get the nominalized energy for NTC, with 32-word , 64-word, and 128-word buffer from left to right. We could conclude that aggregating write by WAB and WBAB with dual-supply will bring a considerable energy efficiency improvement. Though there are still some energy overhead due the additional combinational logics, buffers and bypass, they are negligibly small compare with the dominant core and memory energy consumption. By averaging the nominalized energy, we get approximately 21.45% energy saving.

Finally, we look at the performance impact of our technique. Figure 12(b) shows that by adding WAB, we are able to significantly reduce the execution latency overhead imposed by the transition delay of the supply switching. If there is no buffer in this system, it results a large latency overhead as presented in Figure 6(b) based on the naive greedy switching scheme. However, the WAB and WABA enabled system eliminates this kind of penalty using V_{cmp} signal to determine the best timing to release write operations instead of halting. This improvement allow our technique to achieve more than 21.45% energy saving as well as 10.19% latency reduction.

(a) Energy

(b) Latency

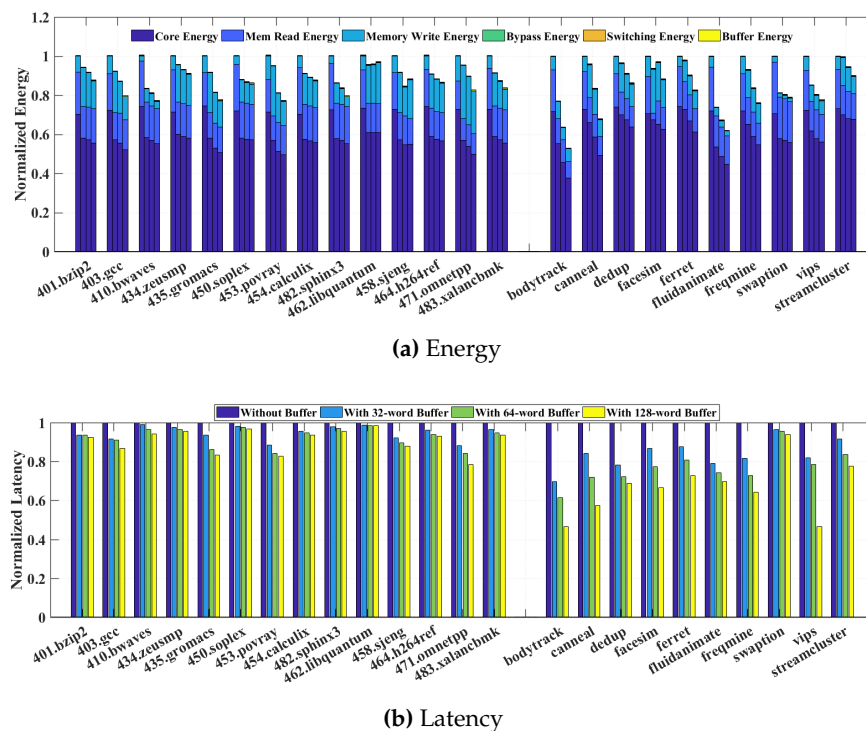



Figure 12. LCRSP improvement with different size buffer among the cache traces of different SPEC2006 and PARSEC benchmarks

6. Conclusion

We present a novel method to improve energy efficiency of near-threshold processors beyond the limit of a single-supply architecture. We perform circuit analysis to distinguish memory read and write reliability at low supply voltages and discover asymmetric behavior where reliable read operations can be achieved at much lower supply voltages than write. Leveraging this asymmetric reliability behavior, we propose an opportunistic dual-supply switching scheme enabled by the additions of write aggregation buffer in the memory hierarchy. Our experimental evaluation results demonstrate that our technique can yield more than 21.45% energy efficiency improvement with 10.19% performance speed-up on average across a variety of diverse benchmarks from SPEC20006 and PARSEC.

Acknowledgments: We thank funding support from NSF Award #1657562.

7. Reference

1. Dreslinski, R.G.; Wieckowski, M.; Blaauw, D.; Sylvester, D.; Mudge, T. Near-threshold computing: Reclaiming moore's law through energy efficient integrated circuits. *Proceedings of the IEEE* **2010**, *98*, 253–266.
2. Gu, Y.; Yan, D.; Vaibhav, V.; Mircea, S.; Zhang, X. SRAM based Opportunistic Energy Efficiency Improvement in Dual-Supply Near-Threshold Processors. Proceedings of the 55th Annual Design Automation Conference 2018. ACM, 2018.
3. Henning, J.L. SPEC CPU2006 benchmark descriptions. *ACM SIGARCH Computer Architecture News* **2006**, *34*, 1–17.
4. Bienia, C.; Kumar, S.; Singh, J.P.; Li, K. The PARSEC Benchmark Suite: Characterization and Architectural Implications. Proceedings of the 17th International Conference on Parallel Architectures and Compilation Techniques, 2008.
5. De, V.; Vangal, S.; Krishnamurthy, R. Near threshold voltage (ntv) computing: Computing in the dark silicon era. *IEEE Design & Test* **2017**, *34*, 24–30.
6. Zou, A.; Leng, J.; Zu, Y.; Tong, T.; Reddi, V.J.; Brooks, D.; Wei, G.Y.; Zhang, X. Ivory: Early-stage design space exploration tool for integrated voltage regulators. Proceedings of the 54th Annual Design Automation Conference 2017. ACM, 2017, p. 1.
7. Ruhl, G.; Dighe, S.; Jain, S.; Khare, S.; Vangal, S.R. IA-32 processor with a wide-voltage-operating range in 32-nm CMOS. *IEEE Micro* **2013**, *33*, 28–36.
8. Zhai, B.; Dreslinski, R.G.; Blaauw, D.; Mudge, T.; Sylvester, D. Energy efficient near-threshold chip multi-processing. Proceedings of the 2007 international symposium on Low power electronics and design. ACM, 2007, pp. 32–37.
9. Zhao, W.; Cao, Y. New generation of predictive technology model for sub-45 nm early design exploration. *IEEE Transactions on Electron Devices* **2006**, *53*, 2816–2823.
10. Nathan, B.; Bradford, B.; others. The gem5 simulator. *ACM SIGARCH Computer Architecture News* **2011**, *39*, 1–7.
11. Li, S.; Ahn, J.H.; Strong, R.D.; Brockman, J.B.; Tullsen, D.M.; Jouppi, N.P. McPAT: an integrated power, area, and timing modeling framework for multicore and manycore architectures. Microarchitecture, 2009. MICRO-42. 42nd Annual IEEE/ACM International Symposium on. IEEE, 2009, pp. 469–480.
12. Zhang, X.; Tong, T.; Kanev, S.; Lee, S.K.; Wei, G.Y.; Brooks, D. Characterizing and evaluating voltage noise in multi-core near-threshold processors. Proceedings of the 2013 International Symposium on Low Power Electronics and Design. IEEE Press, 2013, pp. 82–87.
13. Dreslinski, R.G.; Wieckowski, M.; Blaauw, D.; Sylvester, D.; Mudge, T. Near-threshold computing: Reclaiming moore's law through energy efficient integrated circuits. *Proceedings of the IEEE* **2010**, *98*, 253–266.
14. Bailey et al., B. Near-Threshold Computing. *Semiconductor Engineering* **2017**.
15. Karpuzcu, U.R.; Sinkar, A.; Kim, N.S.; Torrellas, J. Energysmart: Toward energy-efficient manycores for near-threshold computing. High Performance Computer Architecture (HPCA2013), 2013 IEEE 19th International Symposium on. IEEE, 2013, pp. 542–553.

16. Miller, T.N.; Pan, X.; Thomas, R.; Sedaghati, N.; Teodorescu, R. Booster: Reactive core acceleration for mitigating the effects of process variation and application imbalance in low-voltage chips. High Performance Computer Architecture (HPCA), 2012 IEEE 18th International Symposium on. IEEE, 2012, pp. 1–12.

17. Shafique, M.; Garg, S.; Henkel, J.; Marculescu, D. The EDA challenges in the dark silicon era: Temperature, reliability, and variability perspectives. Proceedings of the 51st Annual Design Automation Conference. ACM, 2014, pp. 1–6.

18. Zhang, X.; Tong, T.; Brooks, D.; Wei, G.Y. Evaluating adaptive clocking for supply-noise resilience in battery-powered aerial microrobotic system-on-chip. *IEEE Transactions on Circuits and Systems I: Regular Papers* **2014**, *61*, 2309–2317.

19. Chang, C.H.; Liu, C.Q.; Zhang, L.; Kong, Z.H. Sizing of SRAM cell with voltage biasing techniques for reliability enhancement of memory and PUF functions. *Journal of Low Power Electronics and Applications* **2016**, *6*, 16.

20. Kutila, M.; Paasio, A.; Lehtonen, T. Comparison of 130 nm technology 6T and 8T SRAM cell designs for Near-Threshold operation. Circuits and Systems (MWSCAS), 2014 IEEE 57th International Midwest Symposium on. IEEE, 2014, pp. 925–928.

21. Esmaeilzadeh, H.; Blem, E.; Amant, R.S.; Sankaralingam, K.; Burger, D. Dark silicon and the end of multicore scaling **2011**. pp. 365–376.

22. Rabaey, J.M.; Chandrakasan, A.P.; Nikolic, B. *Digital integrated circuits*; Vol. 2, Prentice hall Englewood Cliffs, 2002.

23. Seevinck, E.; List, F.J.; Lohstroh, J. Static-noise margin analysis of MOS SRAM cells. *IEEE Journal of solid-state circuits* **1987**, *22*, 748–754.

24. Vashishtha, V.; Vangala, M.; Sharma, P.; Clark, L.T. Robust 7-nm SRAM Design on a Predictive PDK. Circuits and Systems (ISCAS), 2017 IEEE International Symposium on. IEEE, 2017, pp. 1–4.

25. Ganapathy, S.; Kalamatianos, J.; Kasprak, K.; Raasch, S. On characterizing near-threshold SRAM failures in FinFET technology. Proceedings of the 54th Annual Design Automation Conference 2017. ACM, 2017, p. 53.

26. Vashishtha, V.; Vangala, M.; Clark, L.T. ASAP7 predictive design kit development and cell design technology co-optimization. Computer-Aided Design (ICCAD), 2017 IEEE/ACM International Conference on. IEEE, 2017, pp. 992–998.

27. Thoziyoor, S.; others. *CACTI 6.5*; HP Laboratories, 2009.