Fine-Grained Voltage Boosting for Improving Yield in Near-Threshold Many-Core Processors

Joonho Kong
School of EE
Kyungpook National University
joonho.kong@knu.ac.kr

Arslan Munir
Dept. of CSE
University of Nevada, Reno
arslan@unr.edu

Farinaz Koushanfar
Dept. of ECE
Rice University
farinaz@rice.edu

ABSTRACT

Process variation is a major impediment in optimizing yield, energy, and performance in near-threshold many-core processors. In this paper, we present a comprehensive analysis on yield losses in near-threshold many-core processors. Based on our analysis, we propose energy-efficient yield improvement techniques for near-threshold many-core processors: SRAM cell arrays and Wordline driver voltage Boosting (SWBoost) and Cache voltage Boosting (CBoost). Results reveal that SWBoost and CBoost improve a chip yield by up to 66% and 83%, respectively. Furthermore, runtime energy overheads of SWBoost and CBoost are only 0.46% and 0.54%, respectively, which are much lower than conventional voltage boosting techniques.

Categories and Subject Descriptors

C.1.2 [Computer Systems Organization]: Multiple Data Stream Architectures (Multiprocessors)—Multiple-instruction-stream, multiple-data-stream processors (MIMD)

General Terms

Design, Performance, Reliability

Keywords

Near-threshold computing; process variations; yield; voltage boosting

1. INTRODUCTION

An effective approach to sustain Moore’s law and alleviate the dark silicon problem [5] in advanced process technologies is to lower the processor’s supply voltage to near the transistor’s threshold voltage ($V_{th}$): a computing paradigm known as near-threshold computing (NTC). Decreasing supply voltage from the nominal operating point (nominal $V_{dd}$ operation is also known as super-threshold computing (STC)) decreases operating frequency and hence performance linearly, leakage power exponentially, and active energy per operation quadratically.

Although NTC enables sustaining Moore’s law, process variations, which are caused by manufacturing imperfections and are also an issue in STC, exacerbate in the NTC regime. Each submicron technology generation becomes increasingly susceptible to process variations that manifest across the chip as fluctuations in transistor parameters (mainly, threshold voltage $V_{th}$ and effective gate length $L_{eff}$) around the nominal values. The parametric variation at NTV causes substantial delay and power variation in circuits of identical processor cores, which limits the maximum operating frequency of the entire many-core processor [6]. Furthermore, this variation in chips’ delay and power consumption beyond design margins severely hurts processors’ yield. Improving processors’ yield is imperative as it can significantly impact the revenue of a semiconductor industry.

In this paper, we conduct a comprehensive yield analysis in the near-threshold regime and propose efficient yield improvement techniques that selectively boost $V_{dd}$ in a fine-grained manner for many-core processors. Compared to conventional coarse-grained voltage boosting techniques [12][13][15][16], our fine-grained voltage boosting techniques considerably improve processors’ yield. Moreover, energy efficiency of our proposed techniques reduces leakage-induced yield losses. Our main contributions are summarized as follows:

- We conduct the first comprehensive component-level analysis of yield losses in a near-threshold tiled many-core processor;
- We identify that SRAM-based structures are most vulnerable to yield losses at NTV and justify the necessity of fine-grained voltage boosting mechanisms;
- We propose fine-grained and selective voltage boosting techniques for tiled many-core processors: SRAM cell arrays and Wordline driver voltage Boosting (SWBoost) and Cache voltage Boosting (CBoost);
- We consider leakage-induced yield losses and reveal ineffectiveness of the previous coarse-grained voltage boosting techniques. We also quantify yield improvement by our proposed fine-grained voltage boosting techniques over the existing techniques.

2. YIELD ANALYSIS FOR NTC

Operation in near-threshold regime significantly impacts yield of manufactured chips due to enhanced effects of process variation. This section classifies and analyzes yield losses for different process variation severities. We classify failures that cause yield losses into eight different categories as summarized in Table 1.

<table>
<thead>
<tr>
<th>Metric</th>
<th>Element</th>
<th>Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timing</td>
<td>Logic</td>
<td>Core logic timing failure</td>
</tr>
<tr>
<td></td>
<td>SRAM &amp; logic</td>
<td>L1-D timing failure</td>
</tr>
<tr>
<td></td>
<td>SRAM &amp; logic</td>
<td>L1-D timing failure</td>
</tr>
<tr>
<td>Stability</td>
<td>SRAM</td>
<td>L1-I stability failure</td>
</tr>
<tr>
<td></td>
<td>SRAM</td>
<td>L1-D stability failure</td>
</tr>
<tr>
<td></td>
<td>SRAM</td>
<td>L2 stability failure</td>
</tr>
<tr>
<td>Power</td>
<td>Chip</td>
<td>Excessive leakage failure</td>
</tr>
</tbody>
</table>

2.1 Reference NTV many-core architecture

Our reference architecture is a tiled many-core processor consisting of 64 tiles similar to Tilera’s TILEPro64 processor [14]. The processor features an 8×8 grid of 64 tiles (processor cores) implemented in 11nm process technology. The nominal $V_{dd}$ and $V_{th}$ are 0.55V and 0.33V [7], respectively. The architectural parameters for cache memories follow Tilera’s TILEPro64 specifications as close as possible [14]. Our NTV many-core processor’s cache memories are composed of 8T SRAM cells which are more robust to process variation than 6T SRAM cells [6].

2.2 Analysis of Yield Losses

**Effect of Target Clock Frequency & Process Variation:** Chip yield depends on target clock frequency and process variation severity. Process variation severity is expressed as...
Our yield loss analysis, which reveals that there are more tiles containing one failing component among L1-I, L1-D, L2, and processor core) as classified in Table 1 excluding leakage power failure. We count the number of tiles containing timing or stability failures in microarchitectural components for a sample of 100 chips where the maximum number of failure occurrences is 64 tiles \times 100 \text{ chips} = 6400. Fig. 3 depicts composition of yield losses for \((\sigma/\mu)_{V_{th}} = 0.15\).

Results reveal that the timing failures in 8T SRAM-based components (i.e., caches) are dominating factors in yield losses. The core logic timing failures are negligible and only appear when the target frequency \(\geq 225\) MHz. The stability failures are also negligible in our many-core processor operating at 0.55 V. Results further indicate that L2 caches are most susceptible to failures due to process variation as compared to other 8T SRAM-based components. The large size of L2 caches as compared to L1 caches result in greater timing failure rate in L2 caches since large cache size implies large number of 8T SRAM cells and a large number of parallel independent delay paths.

3. VOLTAGE BOOSTING TECHNIQUES FOR NTV MANY-CORE PROCESSOR

In this section, we propose fine-grained microarchitectural component-level voltage boosting techniques: SWBoost and CBoost. Our proposed techniques are based on our comprehensive yield analysis, which reveals that SRAM-based components (L1-I, L1-D, and L2 caches) are more susceptible to the effects of process variations than the processor core logic (Section 2.2) in the NTC regime. This section also discusses coarse-grained voltage-boosting techniques proposed in prior work [12][13][16]: tile-level boost (TBoost) and voltage margining (VM). We also provide implementation guidelines of our proposed techniques in this section.

3.1 Proposed Voltage Boosting Techniques

**SWBoost:** SWBoost supplies the boosted \(V_{dd}\) only to the wordline drivers and SRAM cell arrays. The range of boosted \(V_{dd}\) is 0.57V–0.65V. SWBoost can provide yield improvements as most failures in cache memories occur due to timing and stability failure in 8T SRAM cells. SWBoost can provide energy savings as compared to boosting the whole tile or whole cache memories. Fig. 4 depicts our proposed boosting techniques. In the case of SWBoost, we only boost wordline and SRAM cell supply voltage. SWBoost is based on dual-voltage rail (DVR); however, SWBoost operates in a finer-grained manner as compared to the conventional DVR [12][13][15][17]. SWBoost selectively applies the boosted \(V_{dd}\) to faulty cache components to improve yield and energy efficiency. Two power gating P-type metal-oxide-semiconductor (PMOS) transistors are required for a component, which are employed to select either nominal \(V_{dd}\) or boosted \(V_{dd}\) for each component. Since there are three cache memory components (L1-I, L1-D, and L2) in each tile, a total of six PMOS transistors are needed for each tile. To support SWBoost for all of the 64 tiles in the processor, 384 power gating PMOS transistors are required.

To determine whether to supply nominal or boosted \(V_{dd}\) to a cache component, SWBoost uses three fault indicator...
VDD

Figure 4: Our proposed voltage boosting technique.

(F) bits (stored in a non-volatile memory) for each tile: D-cache F-bit, I-cache F-bit, and L2-cache F-bit. D-cache F-bit, I-cache F-bit, and L2-cache F-bit determine if a timing- or stability-related failure exists in L1-D, L1-I, and L2 cache, respectively. A total of 192 bit non-volatile storage is required for storing these F-bits for our many-core processor core. We use the same boosting granularity as presented in [12][13], although prior work does not boost L2 caches.

3.2 Existing Voltage Boosting Techniques

TBoost: TBoost is a coarse-grained voltage boosting technique that supplies boosted voltage at the tile-level in case of failure in any of the tile’s component (processor core, L1-I, L1-D, and L2 cache). In our TBoost implementation, we use the same boosting granularity as presented in [12][13], although prior work does not boost L2 caches.

Voltage Margining (VM): VM [16] is a coarse-grained voltage boosting technique that increases the chip-wide VDD to save the faulty chips (or tiles) and improve performance. VM consumes more energy than SWBoost. As in the case of SWBoost, CBoost also needs 192 F-bits.

3.3 Implementation Issues

Our proposed fine-grained voltage boosting designs (SWBoost and CBoost) require power gating PMOS transistors and non-volatile F-bits, and hence requires some area overhead. A single power gating PMOS transistor requires huge area compared to regular transistors (~ 6K transistors) [12], however, the overall area overhead is negligible compared to the total processor area as the state-of-the-art processors integrate more than a billion transistors. A conservative estimate of area overhead of all the power gating PMOS transistors in our many-core processor is 0.23%. The area overhead of non-volatile storage for 192 F-bits is negligible.

Although our designs use two different voltage levels (nominal and boosted), our designs do not need voltage level converters because of small difference between these voltage levels [11]. In our designs, devices with non-zero threshold voltages can be used in place of voltage level converters.

4. EVALUATION

This section presents evaluation results focusing on yield improvement and energy overhead of our proposed fine-grained voltage boosting techniques (SWBoost and CBoost) and coarse-grained voltage boosting techniques proposed in prior work: TBoost and VM. We also present yield results for the spare tiles (ST) technique for yield improvement: STx represents the case when ‘x’ number of spare tiles are employed for mitigating the effects of process variation. We compare our voltage boosting techniques with a baseline technique in which the L2 cache is excluded from the tile-wide boosting similar to the technique introduced in [12]. Results indicate that SWBoost is most energy-efficient because of fine-grained boosting of only the SRAM arrays and wordline drivers of faulty cache memory components. SWBoost and CBoost show a runtime energy overhead of 0.46% and 0.54% on average, respectively, as compared to the baseline.

4.1 Evaluation setup

For yield estimation, we use VARIUS-NTV [6] process variation model, which is specialized for NTC. We specify Vth and Lref variation severities whereas other parameters are set to their default values in VARIUS-NTV. For workload-dependent energy consumption of our many-core processor, we use Snipersim [1] to extract the access counts of each functional unit, which are then given as an input to McPAT [10] scaled for 11nm technology node. We use 15 multi-threaded benchmarks from SPLASH-2 (barnes, cholesky, lu, ocean, radiosity, radix, and raytrace) and PARSEC (blackholes, bodytrack, dedup, fluidanimate, freqmine, raytrace, streamcluster, and swaptions) for our evaluations.

4.2 Yield

Table 2 summarizes yield results for various boosting techniques with boosted VDD of 0.57V, 0.61V, and 0.65V and spare tiles for various target clock frequencies (TCF) and process variation severities. Results indicate that ST techniques impart best yield when (σ/µ)Vth = 0.1 and target clock frequency is 200 MHz. Our CBoost technique delivers best yield for relatively high target clock frequencies.

For instance, CBoost enables 54% yield, which is highest across all considered techniques, when target clock frequency is 300 MHz and boosted VDD is 0.65V. CBoost enables 15% yield improvement over TBoost when target clock frequency is 300 MHz and boosted VDD is 0.65V. Better yield of CBoost than TBoost is due to the coarser-grained boosting (tile-wide) of TBoost that results in more power/energy consumption. This additional energy consumption in TBoost causes additional leakage-induced yield losses, which limits the yield attainable from TBoost. SWBoost can obtain comparable yield to CBoost and TBoost when the target frequency is low (≤ 200 MHz), however, the yield attainable from SWBoost decreases sharply as the target frequency increases. VM provides the worst yield among all the considered techniques due to low leakage power efficiency that results in high leakage-induced yield losses.

Results indicate that attainable yield from all voltage boosting techniques deteriorates for (σ/µ)Vth = 0.12 and (σ/µ)Vth = 0.15 due to increased process variations. However, overall trend of yield results across various techniques is similar to the case of (σ/µ)Vth = 0.1.

4.3 Energy

Fig. 5 shows the geometric mean of normalized energy results for various multi-threaded workloads for (σ/µ)Vth = 0.15. The ‘TBoost+L2’ in Fig. 5 denotes the TBoost technique in which the L2 cache is excluded from the tile-level VDD boosting similar to the technique introduced in [12]. Results indicate that SWBoost is most energy-efficient because of fine-grained boosting of only the SRAM arrays and wordline drivers of faulty cache memory components. SWBoost and CBoost show a runtime energy overhead of 0.46% and 0.54% on average, respectively, as compared to the baseline.
that fine-grained selective voltage boosting techniques not only reduce both timing- and leakage-induced yield losses but also improve runtime energy efficiency. Based on our yield analysis, we propose architectural component-level Vdd boosting techniques: SWBoost and CBBoost. Results reveal that SWBoost and CBBoost improve a chip yield by up to 66% and 83%, respectively. Results also verify that energy overhead of our proposed techniques is significantly less than the conventional techniques. SWBoost is most energy-efficient among all the evaluated techniques with a maximum energy overhead of only 0.46% as compared to the baseline. In our future work, we plan to incorporate a process variation model for interconnect and optimize our design to take into account both interconnects and tiles under process variations in the NTV regime.

Acknowledgements. This work was supported in part by NSF Career grants (No. 0646829 and 1116858), ARO YIP grant (R17450), ONR R17460, and the University of Nevada, Reno.

7. REFERENCES