Microprocessors and Microsystems 38 (2014) 776–787

Contents lists available at ScienceDirect

Microprocessors and Microsystems

journal homepage: www.elsevier.com/locate/micpro

An ultra-low power resilient multi-core architecture with static and dynamic tolerance to ambient temperature-induced variability

Daniele Bortolotti *, Andrea Bartolini, Luca Benini

Department of Electrical, Electronic and Information Engineering (DEI), University of Bologna, Viale Risorgimento 2, 40136 Bologna, Italy

ARTICLE INFO

Article history:
Received 6 December 2013
Revised 16 May 2014
Accepted 11 June 2014
Available online 21 June 2014

Keywords:
Near-threshold-computing
Ultra-low power
Ambient temperature
Variability
Static/dynamic reconfiguration

ABSTRACT

Near-threshold operation is today a key research area in Ultra-Low Power (ULP) computing, as it promises a major boost in energy efficiency compared to super-threshold computing and it mitigates thermal bottlenecks. Unfortunately near-threshold operation is plagued by greatly increased sensitivity to threshold voltage variations, such as those caused by ambient temperature fluctuation. In this paper we focus on a tightly-coupled ULP processor cluster architecture where a low latency, high-bandwidth processor-to-L1-memory interconnection network plays a key role. We propose an architectural scheme to tolerate ambient temperature-induced variations capable of statically (off-line) and dynamically (on-line) adapting the processor-to-L1-memory latency without compromising execution correctness. We extensively tested our solution in different scenarios and we evaluated the different design trade-offs, showing the cost, performance and reliability gain compared to state-of-the-art static solutions. The dynamic solution, thanks to its lightweight runtime overhead, outperforms the static solution and is able to reach a performance gain up to 25% in a typical use case scenario with a very low (<4%) area overhead.

1. Introduction

Classical CMOS scaling, which drove the semiconductor growth during the past several decades, has recently slowed down. Moore's law is nowadays delivering reduced energy gains [1–3]. In deep sub-micron technological nodes the supply voltage has remained essentially constant and improvements on dynamic energy efficiency have dramatically stagnated. On the other hand leakage contribution continues to increase and highly impacts the power budget of modern embedded systems. In this "Moore's law twilight era", further energy gain can be achieved by moving to the near-threshold computing (NTC) domain [1–5]. Device operation in near-threshold is a very promising design approach and in the last decade several designs showed the feasibility and the benefits of NTC [6,7]. By reducing the supply voltage from the nominal value to the level of the threshold voltage ($V_{dd} \approx V_t$), the energy per operation decreases by a factor of 10× [1,5]. Performance penalties are in the same order of magnitude [1], thus representing the main drawback of this design paradigm. Moreover, reducing further the supply voltage in the sub-threshold region ($V_{dd} < V_t$) is less attractive. Moving from near-threshold to sub-threshold the performance will drop by an additional 50–100× while the associated energy improvement is shown to be only doubled [8].

Despite providing excellent energy-frequency trade-offs for many application scenarios, NTC faces three key challenges that must be overcome for a widespread adoption: *performance variation*, *performance loss* and *functional failure*.

Systematic and random variations are already significant issues in advanced nanometric technology and operating at very-low voltages exacerbates the effect of both. Performance uncertainty in the near-threshold region, due to the global process variation alone (mainly due to random dopant fluctuations and lithographic process), increases of five times from 30% at nominal supply voltage [1,9]. Moreover, operating at low voltage also increase the sensitivity to temperature variations and supply ripple. Both contribute with another 2× factor to the performance variation leading to a total performance uncertainty increase of 20×. The classic solution of worst case design corner cannot tackle such enormous amount of variability. Taking conservative margins with over-design result in systems running at only a small fraction of their potential performance. In addition, guardbanding techniques are wasteful not only in terms of performance but even for energy consumption due to the high impact of leakage currents.

Another main issue with low-voltage operation is the potential performance loss. Operating in near-threshold can seriously limit the degree of usage of voltage-scaling for a given processing

[*] Corresponding author. Tel.: +39 0512092759; fax: +39 0512093785.
E-mail address: daniele.bortolotti@unibo.it (D. Bortolotti).

http://dx.doi.org/10.1016/j.micpro.2014.06.004
0141-9331/© 2014 Elsevier B.V. All rights reserved.

requirement. Parallel computing with multi-core architectures can alleviate this issue, provided that the algorithms to be executed are eligible for a parallel implementation. The work presented in [10] explores the trade-offs in terms of performance and power for single-core or multi-core solutions at near-threshold. The application domain comprise various biomedical signal processing requirements, where often embarrassingly parallel execution exist. Authors estimate more than 30% of energy loss for the single-core design, when compared to the multi-core, under high workload requirements [10]. As a matter of fact, the single-core needs to operate with a supply voltage twice higher than the multi-core solution to achieve the same throughput. In [11] authors show that exploiting NTC, combined with a multi-core architecture design, enables ultra-low power wearable health monitoring systems achieving up to 40% power savings with respect to the state-of-the-art.

The third of the fundamental barriers for NTC is functional failure. More than the logic cells, embedded memories suffer from variations with the high risk of causing severe functional failures. Unfortunately, the failure probability of the conventional 6-transistors SRAM cell increases considerably as the supply voltage is scaled down. Read failure, due to the lack of Static Noise Margin (SNM), is one of the major failure factors, limiting the efficiency of voltage scaling. An SRAM cell in 65 nm technology has a failure probability of 10^{-7} at nominal voltage, while in NTC the failure rate increases by five orders of magnitude to approximately 4% [9]. The usage of more reliable SRAM bit-cells, such as 8-transistors or 10-transistors cells, allows to operate at lower voltage, however, such solutions incur in large area penalties. A typical solution to this problem is to have separate voltage islands for logic and cores. By keeping the power supply of SRAM cells higher will reduce the error rate and, moreover, it enables faster memories in the order of few cycles of latency.

Variability constraints when operating in NTC push the architecture toward a topology in which several processing elements communicate with each other through a shared memory system. An emerging paradigm, among the several many-core architectures recently proposed, consists of leveraging tightly-coupled clusters as building blocks [11–14]. In a shared memory paradigm, these designs try to overcome the scalability problems encountered when increasing the number of Processing Elements (PEs) that share a unique interconnection and memory system. To overcome this problem, such architectures deploy a hierarchical design where PEs are clustered into small-medium sized subsystems. The small number of PEs makes it possible to design a low-latency interconnect between processors and L1 (in-cluster) memories, while scaling to larger system sizes is enabled by cluster replication and a scalable interconnection medium like a Network-on-Chip.

Putting variability in the picture, in such chip multiprocessor architectures the interconnect clearly becomes a single point of failure and therefore a crucial element for system reliability. As a matter of fact, in [15] authors introduced a resilient single-cycle interconnection network, based on configurable pipeline stages, that can statically (boot-time) tolerate delay variations due to static process variations or transistor aging.

Since ultra-low power (ULP) devices operating at near-threshold voltage, due to the low power dissipated are safe from self-heating effects, die temperature is hot-spot free and mainly follows ambient temperature [1,16–18] which can greatly vary for daily/seasonal fluctuations or indoor/outdoor transitions. As a consequence of this, performance variability cannot be effectively addressed only by adopting static solutions, requiring lightweight runtime solutions reactive to variations that can lead to functional failures when ambient temperature significantly changes.

1.1. Contribution

This work tackles this issue further extending a previously presented architectural scheme to achieve resiliency to critical path variations induced by ambient temperature fluctuations [19]. This is done by exploiting a resilient logarithmic interconnect and integrating it with a set of new HW modules capable of sensing the current ambient temperature, recognize possible hazards, checking memory and link consistency and react by reconfiguring, through a SW procedure, the interconnect delays. The dynamic solution proposed in [19] is here coupled with the static variation tolerant scheme. As a result the two architectures are presented with its separate HW modules, control policy and timing error detections routines. The proposed *static* solution thanks to its adaptive nature (selective per-link delays insertion), it shows to a reduced slowdown compared to the static frequency scaling technique and reduced area ($\approx 2.6\%$) and power overheads. On the other hand, the *dynamic* variation tolerant scheme has a slightly higher area ($\approx 3.8\%$) and power overhead but leads to a performance gain up to 25%, when compared to its static counterpart, in a typical use case scenario.

The rest of the paper is organized as follows. In Section 2 the baseline target architecture is introduced. Section 3 discusses in detail the proposed solutions (both static and dynamic approaches) with details on the building blocks of the schemes as well as their working principle. Next, in Section 4 we describe the experimental setup and the simulation framework used to compare the proposed schemes with state-of-the art static solutions. Finally, the conclusions of this work are presented in Section 5.

2. Baseline architecture

The recent shift towards many-core architectures brings new architectural paradigms: today several academic and commercial many-cores architectures deploy a hierarchical design where processing elements are organized into small-medium sized tightly-coupled clusters. We chose as a target cluster architecture one similar to [12,13,11]. Our shared memory cluster, shown in Fig. 1, features 16 Processing Elements (PEs) each one with a private Instruction Cache.

The PEs do not have private data caches or memories, therefore avoiding memory coherency overhead. They all share a first level (L1) multi-banked tightly coupled data memory (TCDM) acting as a shared data scratchpad memory, not as a data cache. Intra-cluster communication is based on a low-latency high bandwidth *Logarithmic Interconnect* (LIC). It consists of a Mesh-of-Trees (MoT) interconnection network (Fig. 2) able to support single-cycle communication between processors and memories, resembling the hardware module initially proposed in [20]. As shown in Fig. 2, the MoT network connects $N = 2n$ PEs and $M = 2m$ Memory Banks (MBs). It contains $\text{Log}_2(M)$ levels of routing primitives and $\text{Log}_2(N)$ levels of arbitration primitives.

The interconnect operates word-level address interleaving on the memory banks to reduce banking conflicts in case of multiple accesses to logically contiguous data structures. The LSBs of the address field determine the routing path to the destination. In case of multiple conflicting requests, for fair access to memory banks, a round-robin scheduler arbitrates the access and a higher number of cycles is needed depending on the number of pending conflicting requests. In case of no banking conflicts data routing is done in parallel for each PE, thus enabling a sustainable full bandwidth for PEs-memories communication. The TCDM has a number of memory ports equal to the number of banks to have concurrent access to different memory locations. Once a read or write requests is brought to the memory interface, the data is available on the