Microprocessors and Microsystems 38 (2014) 707–716

Contents lists available at ScienceDirect

Microprocessors and Microsystems

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

Low-cost scratchpad memory organizations using heterogeneous cell sizes for low-voltage operations

Syed Gilani [a], Taejoon Park [b,*,1], Nam Sung Kim [a,1]

[a] Department of Electrical and Computer Engineering, University of Wisconsin–Madison, Madison, WI, USA
[b] Department of Information and Communication Engineering, Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu, Republic of Korea

ARTICLE INFO

Article history:
Available online 24 June 2014

Keywords:
Scratch pad memory
SRAM
Process variations
Dynamic voltage and frequency scaling

ABSTRACT

Modern digital signal processors (DSPs) execute diverse applications ranging from digital filters to video decoding. These applications have drastically different arithmetic precision and scratch pad memory (SPM) size requirements. To minimize power consumption, DSPs often support aggressive dynamic voltage/frequency scaling (DVFS) techniques, requiring on-chip memory, such as SPM, to operate at low voltages. However, increasing process variations with aggressive technology scaling have significantly increased the failure rate of on-chip memory designed with small transistors operating at low voltages. Consequently, designs must use either larger and/or more transistors to have memory cells satisfy a target minimum operating voltage (V_{MIN}) under a failure rate constraint. Yet using larger and/or more transistors for the SPM, which consumes a large fraction of the chip area, is costly. In this paper, we first propose SPM designs that exploit (i) the characteristics of applications and (ii) the tradeoffs between memory cell size and V_{MIN}. Our approach can reduce the SPMs chip area by up to 17% and V_{MIN} by up to 52.5 mV. Second, we exploit the error-tolerant characteristics of some applications. Our proposed SPM can support lower V_{MIN} with less *mean square error* than a conventional SPM with shortened word width. For error-sensitive applications that require high precision, we can lower V_{MIN} at the cost of reduced memory capacity. This approach may negatively impact the performance of applications with large memory footprints. However, we demonstrate that such applications are typically constrained by their execution latency requirements and are likely to operate at higher voltages/frequencies than applications with smaller memory footprints to satisfy their real-time execution constraints.

1. Introduction

Many applications in the signal processing domain must satisfy both real-time and power constraints. DSPs are often required to provide deterministic execution latencies for these applications while minimizing data transfers between the main memory and the DSP. Consequently, most DSPs prefer software-controlled SPMs, where the software explicitly specifies data transfers and placement. Since the data required by DSP applications during computations usually can be determined beforehand, using SPMs can significantly improve both the performance and power efficiency of the DSPs by minimizing unnecessary data transfers, enabling memory accesses to be overlapped with computation, and avoiding the tag array accesses and comparisons used in data caches.

Unlike applications executed on general-purpose processors (i.e., CPUs), DSP applications usually exhibit a higher error tolerance [1], allowing many DSPs to reduce arithmetic precision to decrease power consumption. In designing on-chip memory such as SPMs, we can exploit this error tolerance to reduce the chip area for the SPM by reducing its precision (i.e., word width) [2]. However, decreasing the word width of the SPM introduces a hard constraint on the precision of most DSP applications. Moreover, applications that require higher precision may need to perform additional memory accesses to pack and unpack data [3]. Thus, reducing the word width of the SPM is not suitable for DSPs intended to execute a wide range of applications.

Modern DSPs are often utilized in system-on-chips (SoCs) where several different accelerators, processors, and input/output (IO) units are implemented on the same chip. These SoCs typically require extremely low-power consumption and thus support independent voltage/frequency (V/F) for these major components control through multiple power domains [4]. On the other hand, DSP applications are typically statically scheduled and extensively

[*] Corresponding author. Tel.: +82 53 785 6313.
E-mail address: tjpark@dgist.ac.kr (T. Park).
[1] These authors contributed equally to this work.

http://dx.doi.org/10.1016/j.micpro.2014.06.002
0141-9331/© 2014 Elsevier B.V. All rights reserved.

profiled to determine their performance in advance and minimize power consumption [5,6]. This allows DSPs to reduce power consumption considerably using aggressive DVFS, but the reduction of power consumption using DVFS is often limited by a V_{MIN}. This is because the amount of random uncorrelated process variations such as random dopant fluctuation (RDF), which affects the threshold voltage of transistors (V_{TH}), increases with each new technology node, as illustrated in Fig. 1. Such random uncorrelated variations increase mismatches between the transistors in static random access memory (SRAM) cells used to build on-chip memory, and they make the cells unstable at low voltages. This in turn increases the failure probability of memory cells and thus limits voltage scaling of DSPs to a V_{MIN}. Due to the increasing failure probability of individual memory cells at low voltages, memory structures with many cells, such as on-chip caches or SPMs, can limit the V_{MIN} of the entire processor; it becomes more difficult to make many cells operate correctly at such low voltages.

The degree of transistor mismatches (i.e., the standard deviations of transistor threshold voltage (σV_{TH})) is inversely proportional to transistor size, as described in [7]: $\sigma V_{TH} \propto 1/\sqrt{W \times L}$ where W and L denote the channel width and length of transistors, respectively. The easiest approach to lower V_{MIN} is to increase the size of transistors in memory cells, thereby decreasing the degree of device mismatches. However, this is not practical because on-chip memories constitute a large fraction of total chip area in many processors (e.g., 75% for the TI 64× processor [8]); a slight increase in individual memory cell size considerably impacts the overall chip area. Although many techniques (e.g., [9,10]) have been proposed to reduce V_{MIN} of memory cells without notably increasing cell size, the overall failure rate and thus V_{MIN} of on-chip memory is fundamentally dominated by their cell size for a given yield target.

In [11], we exploited varying memory footprint size and precision requirements of key DSP kernels and the tradeoffs between memory cell sizes and V_{MIN} to reduce both chip area and V_{MIN} for the SPM, which was the key novelty of our proposed SPM organizations, compared to prior approaches just exploiting error tolerance of DSP applications. In this paper, we extend our prior study [11] to address various impacts of our proposed approaches by providing more comprehensive analyses. More specifically, the techniques presented in [11] allow a greater V_{MIN} reduction for applications with smaller memory footprints, but increase V_{MIN} for applications with larger memory footprints. In this paper, we analyze the relationship between SPM footprint of real-time embedded applications and their compute intensity (computations per second required to meet real-time deadline) and demonstrate that for such applications a larger SPM footprint is associated with a higher compute-intensity. We argue that applications with larger memory footprints (and thus higher compute intensity) cannot employ aggressive DVFS and thus cannot take advantage of V_{MIN} reduction. This is because these applications are constrained by their execution latencies to meet real-time deadlines. We examine key signal and media processing kernels and applications that require a variety of memory footprint sizes and evaluate their execution latencies to demonstrate that most applications with large memory footprints cannot benefit from low V_{MIN}. Consequently, our proposed approaches do not negatively impact the power reduction of applications with large memory footprints. Finally, we also discuss the area and access time impact and scalability of our proposed SPM design to larger SPM sizes.

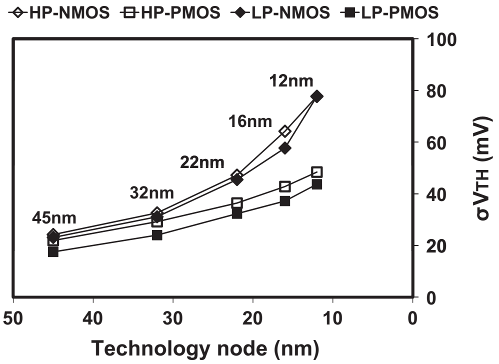

Fig. 1. The standard deviations of threshold voltage (V_{TH}) of high-performance (HP) and low-power (LP) NMOS/PMOS transistors based on ITRS 2009 projections [13].

The remainder of this paper is organized as follows. Section 2 examines the tradeoff between failure probability and the size of 6-transistor SRAM cells. Sections 3–5 present the proposed SPM organizations to exploit varying memory footprint size and precision requirements of key DSP kernels. Section 6 analyzes potential area, V_{MIN}, and power reductions provided by the proposed SPM organizations. Section 7 discusses various impacts of our proposed SPM organizations. Section 8 discusses prior related studies. Section 9 concludes this study.

2. SRAM V_{MIN} versus cell size

We estimate the failure probability of 6-transistor SRAM cells in this study using the method described in [12]. The standard deviation of each transistors V_{TH} is given by [7]:

$$\sigma V_{TH} = \sqrt{\frac{q}{3\varepsilon_{ox}}} \times \sqrt{\frac{T_{ox} \times (V_{TH0} - V_{FB} - 2) \times \phi_B}{W \times L}} \tag{1}$$

where q is the charge amount of an electron, ε_{ox} is the permittivity of SiO_2, T_{ox} is the gate-oxide thickness, V_{TH0} is V_{TH} at zero body bias, V_{FB} is the flat-band voltage, and ϕ_B is the Fermi potential. As Eq. (1) shows, the magnitude of the variations in V_{TH} increases as the transistor size ($W \times L$) decreases. The σV_{TH} for a NMOS (PMOS) transistor with W equal to the minimum L in the high-performance 32 nm predictive technology model (PTM) is 24 mV (29 mV) [13]. For each statistical sample, we apply random V_{TH} values based on Eq. (1) to individual transistors in an SRAM cell, and determine read/write failures using SPICE.

Our baseline cell has the minimum width ($W = 3\lambda$) for all 6 transistors. To analyze the relative area of six different SRAM cells, we create cell layouts using TSMC 0.18 μm technology design rules and a Cadence Virtuoso layout editor. This is shown in Fig. 2 where an SRAM cell consists of a pair of pull-down (PD), pass-gate (PG), and pull-up (PU) transistors. The widths of PD, PG, and PU transistors are denoted by W_{PD}, W_{PG}, and W_{PU}, respectively. Since W_{PG} is often less than or equal to W_{PD}, it does not contribute to the cell size increase. Thus, the sum of W_{PD}, W_{PG}, and a fixed width cost (e.g., the minimum spacing between the active to N-well areas) determines the overall area of an SRAM cell. Note that the cell height is usually fixed while the sum of W_{PD} and W_{PU} determines the width of the cell; the minimum size cell has W_{PD} + W_{PU} equal to 6λ plus the fixed width cost. For each cell, we sweep the size of W_{PD}, W_{PG}, and W_{PU} to find the minimum failure probability of each cell using the high-performance 32 nm PTM and most probable failure point (MPFP) method as follows:

Objective:

$$miminize(P_{FAIL}(W_{PU}, W_{PD}, W_{PG})) \tag{2}$$

Constraint:

$$W_{PG} \leqslant W_{PD}, \quad W_{PU} + W_{PD} \leqslant W_{CONST} \tag{3}$$