Microelectronics Reliability 91 (2018) 86–97

Contents lists available at ScienceDirect

Microelectronics Reliability

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

Lifetime-aware scheduling in high level synthesis

Siavash Es'haghi, Mohammad Eshghi*

Shahid Behesti University, Iran

ARTICLE INFO

Keywords:
High level synthesis
Scheduling
Reliability
Aging
Lifetime

ABSTRACT

Among various reliability challenges, accelerated aging is of particular importance. The flexibility of high-level synthesis (HLS) can be employed at an affordable cost to increase the reliability, and in particular lifetime, of the systems. In this paper, we extract the aging characteristics of the functional units (FUs), then, a scheduling method based on integer linear programming (ILP) is proposed to increase the lifetime, using the aging characteristics of the FUs. The proposed method extends the lifetime beyond the minimum expected value, with the minimum latency overhead, by applying the operation chaining and multicycling techniques in an aging-aware approach. The constraint matrix of the proposed ILP formulation is totally unimodular. This technique is suitable for both data-flow intensive and control-flow intensive designs and is applicable for large circuits. Experimental results show that the proposed approach increases the lifetime by 2.33×, and increases the latency by an average of 19.8%.

1. Introduction

Designers of modern electronic systems are consistently facing demands for higher performance, more integrated features, and greater power and cost efficiency. These demands drive the need for a continuous downscaling of very large scale integration (VLSI) technology. As a result, new challenges have emerged in designing electronic systems, especially in the era of nanoscale technologies. One of the most important challenges is to ensure reliability, and in particular, the lifetime of the digital systems.

System reliability is affected by design complexity, process variability, thermal density, noise and soft errors. Moreover, nanoscale transistors suffer from accelerated aging, which degrades transistor characteristics, such as threshold voltage (V_{th}) and channel current, over time [1]. As a result, the error rate increases as the circuit approaches the end of its lifetime, as illustrated in Fig. 1 [2]. Aging severity is increasing in new CMOS technologies, whilst new components such as FinFETs also suffer from the aging phenomenon [3,4]. Therefore, it is necessary to address the lifetime of a circuit as a significant design parameter, along with delay, power, and area.

Various aging compensation and mitigation techniques have been proposed in literature [5–12]. These techniques are broadly classified into design-time and run-time approaches. Worst-case guard-banding, gate sizing [5–7], logic restructuring [8] and pin reordering [9] are some design-time solutions. Run-time techniques such as dynamic voltage and frequency scaling (DVFS) [10], adaptive body biasing (ABB) [11], and power gating [12], are activated based on the aging severity of the running circuit.

VLSI design flow is evolutionary. It involves a number of abstraction levels, from system to physical, in order to conform to design specifications [13]. Reliability optimization can be performed at different abstraction levels. A lot of work has been done at low levels of abstraction to model [2], analyze and manage aging effects. As the level of abstraction increases, the complexity of the circuit decreases, and the degrees of freedom with which to optimize reliability increase. Kumar et al. proposed an aging-aware logic synthesis method [14], where higher levels of abstraction provide greater flexibility to explore the design space. Hence, these levels are more attractive for optimizing the design, and in particular, the lifetime of the circuit. This feature is essential in nano-CMOS circuit design. To date, limited research has been conducted to investigate the lifetime potentials of design space exploration.

High-level synthesis (HLS), also known as behavioral synthesis, is the process of translating a behavioral description into a hardware implementation at register-transfer level (RTL) and is performed at the early stages of the design flow. The design specification is normally written as a behavioral description, in languages such as C. The behavioral description is first compiled into an internal representation, such as a control and data flow graph (CDFG), which is then mapped to the functional units (FUs) that are selected from the resource library to meet the design goals. The most important next steps are scheduling and binding [15]. Scheduling assigns each behavioral operation (such

* Corresponding author.
E-mail address: m-eshghi@sbu.ac.ir (M. Eshghi).

https://doi.org/10.1016/j.microrel.2018.06.016
Received 27 January 2018; Received in revised form 5 May 2018; Accepted 15 June 2018

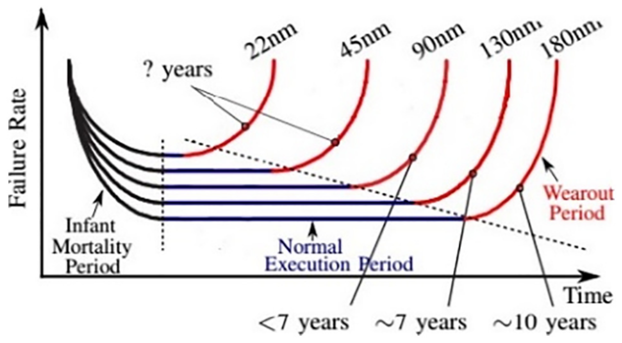

Fig. 1. Failure rate projections considering transistor aging [2].

as add and multiply) in a CDFG to one or more clock cycles (also called control states). By determining the concurrent operations, scheduling plays an important role in exploring the design space. Also, decisions regarding chaining and multicycling are taken at this stage. Binding maps behavioral operations and variables to hardware resources [15].

In order to achieve the expected system reliability and lifetime, it is necessary to consider reliability issues in early design decisions [16]. By highlighting the importance of decisions in early VLSI design stages, a framework to assess soft error reliability parameters in the high-level behavioral description is proposed by Shazli et al. [17]. The flexibility of HLS in design space exploration can be fully leveraged to enhance the system lifetime in a cost-efficient way. In the context of the reliability-aware HLS, previous research has been conducted to exploit the potential of scheduling and binding in reliability enhancement [18–25].

The work in [18,19] explored the expensive duplication or triple modular redundancy of FUs. In [20–22], for each behavioral operation, multiple FUs with different characteristics in terms of delay, area and reliability are implemented. Multicycling is then investigated based on FU features and design objectives. The authors of [23] have focused on chaining the operations aimed at increasing the reliability of the circuit. These papers have mainly focused on the soft errors. A methodology to consider process variability at the behavioral level synthesis is proposed in [24]. The method proposed in [25], minimizes leakage power during high-level synthesis of circuits with bounded delay degradation, using multi-V_{th} resource libraries. Only resource binding is investigated in [25]. Also, the result of this method may be suboptimal.

The lifetime of modern digital circuits is one of the most important reliability challenges. However, process variation and soft errors are considered in most scheduling and binding techniques to optimize reliability [18–24]. Although HLS decisions have significant benefits in terms of reliability, few researchers have considered aging-aware HLS. Therefore, extending the lifetime of circuits using HLS features is still a problem. In particular, the potential of scheduling to optimally improve circuit lifetime has not been explored in previous studies.

In this paper, a new scheduling method is presented to ensure the expected lifetime of a circuit at the design-time. First, the aging characteristics of all FUs are extracted, then the first ILP-based scheduling is performed to determine the minimum latency based on the aging characteristics of FUs. The second ILP-based scheduling is performed under the latency constraint aimed to extend the lifetime, beyond the minimum expected value. Decisions regarding operation chaining and multicycling are considered in this scheduling in such a way that the timing constraints are not violated during the expected lifetime. Using chaining and multicycling techniques, this scheduling is suitable for both data-flow intensive and control-flow intensive designs. The proposed ILP formulation gives the optimal solution in polynomial time and is applied to large circuits. This is the first attempt that performs aging-aware scheduling to extend the circuit lifetime. Experimental results show that this proposed approach ensures the expected minimum lifetime and leads to a longer lifetime compared to aging-unaware scheduling schemes.

The rest of the paper is organized as follows. Background and related work are reviewed in Section 2. Section 3 provides motivational examples for the proposed method of this paper, explained in Section 4. Experimental results are presented in Section 5. Section 6 concludes the paper.

2. Background and related work on aging and HLS

The aging of transistors is one of the major reliability concerns of nanoscale VLSI circuits. Aging increases the delays of FUs over time [1], until these delays cause the timing constraints to violate. As a result, the failure rate increases. The lifetime of a circuit is defined as the timespan that a circuit works without a timing violation, whilst producing exact results. Various aging compensation and mitigation techniques have been proposed at different abstraction levels of the design flow [6,14,25]. HLS is one of the most attractive levels to deal with aging effects, due to its large degrees of freedom, allowing design space exploration to be performed easily. Therefore, inferior results are ignored at early design phases, in a cost-efficient way. In this section, the aging-induced reliability issues and HLS are briefly explained. Related studies are introduced as well.

2.1. Aging and lifetime

As technology scales, reliability becomes a serious challenge in designing VLSI circuits. Reliability of nanoscale VLSI circuits is affected by a variety of factors such as increasing design complexity, process variability, noise, radiation-induced soft errors and aging. Among various reliability challenges, accelerated transistor aging is of particular importance. Negative Bias Temperature Instability (NBTI) and Hot Carrier Injection (HCI) are the dominant sources of the transistor aging [1]. Both NBTI and HCI degrade the threshold voltage and the drive current of the transistor, where the degradation rate depends on various aspects such as workload, temperature, supply voltage and frequency. As a result, devices become slower and delays of FUs increase over time. Due to these increasing delays, timing failures are very likely to occur. Consequently, considerable reduction in the overall lifetime of the digital system is observed, Fig. 1 [2].

The next technology nodes will make transistor aging worse [2]. Further, research indicates that next generation CMOS structures such as FinFETs, suffer from transistor accelerated aging [4,26,27]. Hence, traditional design parameters are no longer sufficient to make a high quality design. Reliability and in particular, lifetime, should be addressed as important design constraints.

In order to study and improve circuit reliability under aging, numerous research has been conducted at various levels of the design flow [5–12,14,25]. At a very low level of abstraction, major efforts have focused on studying the physical origin of the aging phenomenon and developing appropriate models [2]. These models are used to predict aging at higher levels of abstraction. A model for NBTI degradation and recovery is developed in [28], based on the trapping/detrapping mechanism of aging. To consider workload effects, the input signals are characterized with two parameters: signal probability (SP) and transition density (TD). SP is defined as the average number of clock cycles in which the logic value of a signal is 1; and TD is the average number of signal transitions during one clock cycle. An aging-analysis flow on gate level and module level that considers both NBTI and HCI is presented in [29]. Their model is a workload-aware model for predicting the aging severity of a circuit, based on SP and TD information of input vectors of a circuit.

Aging compensation and mitigation techniques have been proposed at higher levels of abstraction. Transistor and gate sizing techniques modify the device area to decrease aging severity [5–7]. Based on detecting functional symmetries and transistor stacking effects, logic restructuring and pin reordering approaches have been proposed in [8,9]. During circuit standby mode, input vector control, internal node control