Microelectronics Reliability 55 (2015) 990–1004

Contents lists available at ScienceDirect

Microelectronics Reliability

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

Swarm intelligence driven design space exploration of optimal *k*-cycle transient fault secured datapath during high level synthesis based on user power–delay budget

Anirban Sengupta [a], Reza Sedaghat [b,*]

[a] Computer Science & Engineering, Indian Institute of Technology, Indore, India
[b] Electrical & Computer Engineering, Ryerson University, Toronto, Canada

ARTICLE INFO

Article history:
Received 19 September 2014
Received in revised form 15 March 2015
Accepted 15 March 2015
Available online 30 March 2015

Keywords:
High Level Synthesis
Multi-objective design space exploration
DSE, Secured DSE
Control Data Flow Graph (CDFG)
k-cycle Fault
Single Event Upsets (SEU)

ABSTRACT

Fault security indicates ability to provide error detection or fetch correct output. Fault security assures possibility of using either hardware redundancy or time redundancy to optimize the overheads associated with fault security. However, generation (design space exploration (DSE)) of an optimal fault secured datapath structure based on user power–delay budget during high level synthesis (HLS) in the context *k*-cycle transient fault is considered an intractable problem. This is due to the fact that for every type of candidate design solution produced during exploration, a feasible *k*-cycle fault secured datapath may not exist which satisfies the conflicting user constraints/budget. Secondly, insertion of inapt cut (resulting in an additional checkpoint) to optimize delay overhead associated with fault security in most cases may not result in optimal solutions in the context of user constraints/budgets. The solutions to the above problems have not been addressed in the literature so far. The paper therefore presents the following novelties: (a) an algorithm for fault secured DSE process (b) handling *k*-cycle transient faults during DSE (c) schemes for selecting appropriate edges for inserting cuts that selects available locations in the scheduled Control Data Flow Graph (CDFG) which minimizes delay overhead associated with fault security (d) swarm intelligence (particle swarm optimization) driven DSE process that adaptively/intelligently computes the candidate design solutions for generating an optimal fault secured datapath.

Results of the proposed approach when tested on standard benchmarks yielded optimal results in most cases as evident from the data obtained for generational distance (GD), spacing (S), spreading (Δ) and weighted metric (W_m). Further, results of comparison with a recent approaches indicated significant reduction of final cost (better quality) for the proposed approach.

1. Introduction

DSE in high level synthesis includes searching an optimal datapath from a set of assorted design alternatives of equivalent functionality which offer higher performance and lower power expenditure with complete fault reliability. The aim of the exploration approach is to reduce the large and complex design space into a set of feasible design solutions meeting multiple designs objectives and functionality. This DSE process in HLS aims at optimizing conflicting issues like area, power and performance along with certain orthogonal issues like runtime and quality of results (QoR) [1–11,38–40].

However, optimizing only area, power and performance remains no longer sufficient now. This is applicable specifically for current generation of systems which demand designs (especially for space applications where radiation induced faults are highly possible) that require ability to detect errors occurring due to transient fault (such as single event upset). Transient faults are radiation induced faults which are non-permanent in nature. These nonrecurring faults can be caused by energized particles, environmental noise or electromagnetic interference. The duration of such faults is in order of a few picoseconds [12,13]. The occurrence of transient faults has increased due to recent advancements in technology where the packing of millions of transistors on a single chip have become feasible. The increase in density per unit area is negatively impacting the device and overall systems reliability by making it susceptible to transient fault or the Single Event Upset (SEU) [14] especially in space applications. Therefore, to

* Corresponding author.
E-mail addresses: asengupt@iiti.ac.in (A. Sengupta), rsedagha@ee.ryerson.ca (R. Sedaghat).

http://dx.doi.org/10.1016/j.microrel.2015.03.010
0026-2714/

achieve complete reliability of the system, multi cycle transient fault security [14–18] needs to be considered as design metric (or constraint) during multi-objective DSE in HLS [17].

The objectives of the presented work are as follows:

(a) Proposes algorithm to design a *k*-cycle transient fault secured SDFG.
(b) Proposes the first fault secured multi-objective design space exploration approach to identify optimal single/multi-cycle (*k*-cycle) transient fault detectable/secured solutions of the design space based on upper limit of conflicting user budgets.
(c) Proposes schemes for selecting appropriate edges for inserting cuts that selects available locations in the scheduled CDFG which minimizes delay overhead associated with fault security.

2. Related work

So far in the literature, many attempts have been made to solve the DSE problem in HLS. The approaches developed so far are aims at exploring the design space along with balancing multi-conflicting issues during generation of the optimal/near-optimal design alternative (or Pareto front). Over the years the process of DSE has evolved where the requirements specified by the user have also become more convoluted, ranging from simple area–delay tradeoff in initial years to complex power-delay-reliability tradeoff in recent years. For example, HLS approaches such as [12,20–22,42] just included the aspect of single cycle fault security with hardware redundancy, but without any focus on evolving/exploring an optimal multi-cycle fault secured design based on user power-delayconstraints. This turns insufficient especially in the context of multi-cycle fault secured design with strict demands of power budget and delay where just generating a fault secured design is non-optimal (as it suffers from user constraint violation). Further, approaches [12,20–22] have chosen to insert cuts to optimize the resource overhead due to fault security. However, insertion of inapt cut point in the design to optimize overhead associated with fault security in most cases may fail to yield optimal solutions in the context of user constraints/budgets and multi cycle transient fault. Additionally, all previous approaches on fault security besides ignoring apposite selection of cut, have also not dealt with generating exploring optimal multi cycle transient faults (*k*-cycle) secured designs in context of user constraints. The contributions of Refs. [12,20–22,42] are discussed below.

For example, in [12] authors duplicate the Control Data Flow Graph (CDFG) and try to map the second unit onto the same hardware as the first, adding FUs as needed. The technique uses the algebraic properties of associativity, distributivity, and commutativity to aid mobility in scheduling the duplicate CDFG and thus take better advantage of idle resources. The approach in [20] involves partitioning of the CDFG into regions or sub graphs. The authors presented a hardware redundancy based CED approach which breaks the data dependences between the nodes. This is done to improve the sharing between normal and duplicate computations. The original and the duplicate computations which are represented by a region are performed on distinct hardware. This is done so that, every regions output can be compared to identify the faults within the regions. For this, voting/comparing on the results of the regions is done. In [21], a concurrent error detection (CED) scheme is employed to detect and isolate the faults within a system (circuit) while it is in use. In [22] authors investigated a method for exploring the tradeoff between the area and latency of the CED design in HLS. The approach sometimes used hardware redundancy or time redundancy or a combination of both to produce fault secure designs. Designs were made secure on the basis of checkpointing introduced in the system. Further, authors in [42] introduced a high level online test synthesis tool which is robust and capable of providing online testability within the iterative process of high level synthesis. More specifically, the authors have presented a high level CAD tool that provides self-checking designs with minimum possible latency, with design space exploration focusing on the area–delay driven cost function. The approach presented in [42] is different than the proposed work in the following aspects: (a) the proposed work considers during DSE, power–delay cost function (b) the proposed work during DSE, considers multi-cycle transient faults (c) proposed work considers during DSE, insertion of apt cuts to optimize delay overhead associated with fault security.

Besides above, [1,23–28] are worth mentioning which are efficient DSE approaches/tools however without any consideration on transient fault aspect during DSE. For example, authors in [23] have proposed simulated annealing (SA) DSE method called 'SALSA' for optimizing delay which uses many probabilistic search operators to enhance the performance of SA-based technique. Authors in [24] have applied GA to the binding and allocation phase. A specific crossover technique has been introduced which is based on force directed algorithm. The limitations of approaches [23,24] besides being unable to handle transient fault is that it does not consider power factor during design trade-off. In addition, Authors in [25] introduced a tool called AutoPilot for HLS. It performs C/C++/systemC-to-RTL synthesis. Although this tool performs area–performance–power tradeoff during DSE, however, the datapath architecture yielded by the tool does not have ability to detect transient fault. Further, authors, in approach [26] also used simulated annealing to generate optimum results; however, the work did not consider transient fault detection during exploration. In [1], authors introduced a tool called SystemCoDesigner that offers rapid design space exploration with rapid prototyping of behavioral systemC models. An automated integrating approach is developed by integrating behavioral synthesis into their design flow. However, the approach besides being unable to handle transient fault, is also limited to area–delay tradeoff. Additionally, authors in [27] proposed an approach based on hierarchical and multiple clock domain HLS to target low power design on FPGA. The authors targeted FPGA unlike the proposed approach. Further, this work besides being unable to handle transient fault also does not consider power constraint as it only considers throughput. In addition, authors in [28] proposed a machine learning method for DSE which introduce a transductive experimental design that can wisely sample micro-architecture choices and use them for training in the learning model. The approach, besides being unable to detect faults, does not consider power during exploration. But the proposed approach presented in this paper besides considering power and execution time (or delay) as design objective during exploration, also considers multi cycle transient faults during optimal datapath generation. Therefore, it ensures an optimal fault secured datapath generation after exploration based on conflicting user constraints. Further, the proposed DSE framework is also driven by an intelligent PSO algorithm which incorporates multiple parameters and conditions to handle efficient exploration.

3. Background/advantages of PSO

Particle Swarm Optimization (PSO) is a heuristic search methodology that tries to imitate the travels of a flock of birds aiming at finding food [29,34]. PSO is based on a population of particles flying through a multi-dimensional search space. Each particle possesses a position and a velocity; both variables are changed to emulate the social psychological tendency to impersonate the