Functional Design Errors in Digital Circuits

Lecture Notes in Electrical Engineering
Volume 32

For other titles published in this series, go to
www.springer.com/series/7818

Kai-hui Chang · Igor L. Markov · Valeria Bertacco

Functional Design Errors in Digital Circuits

Diagnosis, Correction and Repair

Springer

Dr. Kai-hui Chang
University of Michigan
Dept. Electrical Engineering &
Computer Science
Ann Arbor MI 48109-2122
USA
changkh@umich.edu

Dr. Valeria Bertacco
University of Michigan
Dept. Electrical Engineering &
Computer Science
Ann Arbor MI 48109-2122
USA
valeria@umich.edu

Dr. Igor L. Markov
University of Michigan
Dept. Electrical Engineering &
Computer Science
Ann Arbor MI 48109-2122
USA
imarkov@umich.edu

ISBN: 978-1-4020-9364-7 e-ISBN: 978-1-4020-9365-4

DOI 10.1007/978-1-4020-9365-4

Library of Congress Control Number: 2008937577

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

To the synergy between
science and engineering

Contents

List of Figures

List of Tables

Preface

The dramatic increase in design complexity of modern circuits challenges our ability to verify their functional correctness. Therefore, circuits are often taped-out with functional errors, which may cause critical system failures and huge financial loss. While improvements in verification allow engineers to find more errors, fixing these errors remains a manual and challenging task, consuming valuable engineering resources that could have otherwise been used to improve verification and design quality. In this book we solve this problem by proposing innovative methods to automate the debugging process throughout the design flow. We first observe that existing verification tools often focus exclusively on error detection, without considering the effort required by error repair. Therefore, they tend to generate tremendously long bug traces, making the debugging process extremely challenging. Hence, our first innovation is a bug trace minimizer that can remove most redundant information from a trace, thus facilitating debugging. To automate the error-repair process itself, we develop a novel framework that uses simulation to abstract the functionality of the circuit, and then rely on bug traces to guide the refinement of the abstraction. To strengthen the framework, we also propose a compact abstraction encoding using simulated values. This innovation not only integrates verification and debugging but also scales much further than existing solutions. We apply this framework to fix bugs both in gate-level and register-transfer-level circuits. However, we note that this solution is not directly applicable to post-silicon debugging because of the highly-restrictive physical constraints at this design stage which allow only minimal perturbations of the silicon die. To address this challenge, we propose a set of comprehensive physically-aware algorithms to generate a range of viable netlist and layout transformations. We then select the most promising transformations according to the physical constraints. Finally, we integrate all these scalable error-repair techniques into a framework called *FogClear*. Our empirical evaluation shows that FogClear can repair errors in a broad range of designs, demonstrating its ability to greatly reduce

debugging effort, enhance design quality, and ultimately enable the design and manufacture of more reliable electronic devices.

This book is divided into three parts. In Part I we provide necessary background to understand this book and illustrate prior art. In Part II we present our FogClear methodologies and describe theoretical advances in error repair, including a counterexample-guided error-repair framework and signature-based resynthesis techniques. In Part III we explain different components used in the FogClear flow in detail, including bug trace minimization, functional error diagnosis and correction, an incremental verification system for physical synthesis, post-silicon debugging and layout repair, as well as methodologies for spare-cell insertion. Finally, we conclude this book and summarize our key techniques in the last chapter.