A Property Checking Methodology for Weakly Programmable System-on-Chip IPs

Eine Methode zur Eigenschaftsprüfung von schwach programmierbaren System-on-Chip IPs

Vom Fachbereich Elektrotechnik und Informationstechnik
der Technischen Universität Kaiserslautern
zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften (Dr.-Ing.)
genehmigte Dissertation

von
Dipl.-Ing. Sacha Loitz
geb. in Hamburg, Deutschland

D 386

Dekan:	Prof. Dr.-Ing. Hans D. Schotten
Gutachter:	Prof. Dr.-Ing. Dr. rer. nat. habil. Wolfgang Kunz, Technische Universität Kaiserslautern
2. Gutachter:	Prof. Dr.-Ing. Görschwin Fey, Unversität Bremen
Datum der mündlichen Prüfung:	02. Oktober 2013

Berichte aus der Halbleitertechnik

Sacha Loitz

A Property Checking Methodology for Weakly Programmable System-on-Chip IPs

Eine Methode zur Eigenschaftsprüfung von schwach programmierbaren System-on-Chip IPs

D 386 (Diss. Technische Universität Kaiserslautern)

Shaker Verlag
Aachen 2014

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at http://dnb.d-nb.de.

Zugl.: Kaiserslautern, TU, Diss., 2013

Copyright Shaker Verlag 2014
All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior permission of the publishers.

Printed in Germany.

ISBN 978-3-8440-2468-5
ISSN 0945-0785

Shaker Verlag GmbH • P.O. BOX 101818 • D-52018 Aachen
Phone: 0049/2407/9596-0 • Telefax: 0049/2407/9596-9
Internet: www.shaker.de • e-mail: info@shaker.de

Acknowledgments

The content of this thesis is the result of five years of intensive research conducted in the Electronic Design Automation Group at the University of Kaiserslautern. This work would not have been possible without the support of numerous people whom I would like to thank.

First of all I owe special thanks to my supervisor Prof. Wolfgang Kunz for giving me the opportunity to work in this very interesting research field. His constant support, fruitful advises and permanent motivation were important factors for the success of this dissertation.

For the review of my thesis, the therefore shown interest in my work, and the resulting feedback I thank Prof. Görschwin Fey.

The designs analyzed in the course of this work were developed in the Microelectronic Systems Design Group at the University of Kaiserslautern. I thank Prof. Norbert Wehn for allowing us to verify these designs, his support and the great interest that he has shown in the results of my work. Furthermore, I am very grateful to the designers Timo Vogt and Christian Brehm for introducing me into the design flow, the fruitful collaboration and for being always open and helpful whenever I observed some design behavior that did not fit to my properties.

I express my gratitude to Markus Wedler and Dominik Stoffel for all the discussions and the corrections of papers and this dissertation. From the Electronic Design Automation Group I thank Carmen Vicente-Fess, Binghao Bao, Andreas Christmann, Minh Duc-Nguyen, Roland Hecker, Matthias Legrom, Oliver Marx, Ingmar Neumann, Evgeny Pavlenko, Bernard Schmidt, Hans Serr, Max Thalmaier, Joakim Urdahl and Carlos Villarraga for the great time that we had.

I thank my parents for their support and motivation and last but not least I am deeply indebted to my wife Xuemei and our daughters Nina and Sophie who often missed me while I was working on the finalization of this thesis.

To my wife Xuemei

Abstract

Over the last years we have seen an increasing trend towards application-specific instruction set processors (ASIPs) that are designed in order to meet the required flexibility, energy efficiency and performance constraints of the targeted application. In certain application domains the programmability of the designed ASIP is very limited. The processor is built to offer just enough flexibility as needed for the application while delivering an energy efficiency and computing performance which is comparable to that of custom hardware. Such processors are called weakly programmable IPs (WPIP). A main characteristic of these WPIPs is a customized memory architecture with memories being accessed from several pipeline stages.

It turns out that the design methodology followed by the designers of such WPIP modules hampers the validation of these IPs. On the one hand simulation-based approaches cannot achieve sufficient verification coverage. Formal verification techniques, on the other hand, can achieve sufficient coverage, however, today's techniques are oriented at standard processor designs, not ASIPs. The design flow and distributed memory architecture of WPIP designs forbid the usage of the verification techniques as they are used in today's formal verification of standard processors. The main reason for this is the absence of a well defined ISA model that serves as a basis in today's formal processor verification. Furthermore, the distributed memory access makes it extremely difficult or even impossible to apply the existing techniques to ensure that a property set is complete.

This dissertation presents a systematic approach to completely verifying these WPIP designs. A main aspect of the presented technique is that the verification engineer is required only to specify the intended behavior at the level of well documented pipeline operations, resulting in the operational ISA (OISA) model introduced in this work. The additional information required to handle the side effects of the distributed memory access as well the behavior ensuring that the property set is complete are automatically generated by an analysis of the OISA model.

For WPIPs software and hardware are tightly coupled, making it important to ensure that the combined hardware/software system is working correctly. The second part of this work discusses how to reuse the property set from the hardware verification for the verification of the software.

The presented approach is demonstrated at the example of two WPIPs for channel decoding. Even though these WPIPs have undergone intensive simulation-based verification before our techniques were applied, several bugs were discovered with the presented formal verification methodology.

Contents