



FP6-IST-507219

PROSYD

Property-Based System Design

Instrument: Specific Targeted Research Project

Thematic Priority: Information Society Technologies

Improved Static Property Checking Tool

(Deliverable D3.2/8 – Public Version)

Due date of deliverable: April 30, 2005

Actual Delivery date: July 24, 2005

Start date of project: 01.01.2004

Duration: 3 years

Organisation name of lead contractor for this deliverable:

OneSpin Solutions (replacing Infineon Technologies as PROSYD partner from May 15, 2005, subject to Commission approval)

Revision: 1.0

Project co-funded by the European Commission within the Sixth Framework Programme (2000-2006)		
Dissemination Level		
PU	Public	<input checked="" type="checkbox"/>
PP	Restricted to other programme participants (including the Commission Services)	<input type="checkbox"/>
RE	Restricted to a group specified by the consortium (including the Commission Services)	<input type="checkbox"/>
CO	Confidential, only for members of the consortium (including the Commission Services)	<input type="checkbox"/>

Notices

For information, contact klaus.winkelmann@onespin-solutions.com.

This document is intended to fulfil the obligations of the PROSYD project concerning deliverable D3.2/8, described in contract number 507219.

The information in this document is provided "as is", and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

© Copyright PROSYD 2005. All rights reserved.

Table of Revisions

Version	Date	Description and Reason	By	Affected Sections
0.1	April 2005	Created	KW	new
0.9	May 9, 2005	first complete internal version	KW	all
0.91	May 10 2005	minor corrections from RK	KW	3.
0.93	June 22, 2005	comments C. Eisner	KW, RB	all
0.93p	July 11, 2005	Public version	kw	all
1.0	July 24, 2005	Approval by project management	Ce	Version number

Authors

Raik Brinkmann, Klaus Winkelmann

Executive Summary

The CVE static verification engine has been tuned using several optimisations. On a test-bench of PSL properties, an overall speed-up of more than 100% has been observed.

Purpose

The purpose of this document is to report the technical optimisations implemented in the CVE tool as a result of the PROSYD task 3.2.

In this public version, the focus is on the users' view, and some technical details have been omitted.

Intended Audience

This guide is intended for individuals designing static proof engines for PSL. It is assumed that readers are familiar with model checking, SAT and automata construction.

Background

Task 3.2 aims at improving prover performance for the specific requirements derived from PSL properties. Several approaches have been studied in previous reports 3.2/1, 3.2/2 and 3.2/3. While the RT reductions reported in 3.2/2 and 3.2/3 did not produce the improvements hoped for, the improvements at the SAT level presented in 3.2/1 were very successful, and have been implemented into the tool.

Contents

- 1 Introduction.....1
- 2 Technical approach2
- 3 Results3
- 4 Tool manual5
- 5 References.....6

Table of Figures

Figure 1 Internal tool flow.....**Error! Bookmark not defined.**

Table of Tables

Table 1 Performance Summary.....**Error! Bookmark not defined.**
Table 2 CVE User Documentation (3,999) Table of Contents.....**Error! Bookmark not defined.**

Glossary

DUV: Design under Verification. An RT level model of a hardware design.

HDL: Hardware Design Language, such as VHDL, Verilog.

Macro machine: a finite state machine which remains in a stable state as long as all its inputs remain stable.

Model Checking: The automatic or almost automatic verification of a property of a model of a hardware component or in some cases of software.

SAT: satisfiability, the question whether a given set of Boolean formula has a solution.

SAT solver: algorithm that decides a SAT problem and returns a solution if it exists.

1 Introduction

1.1 Background

The major goal of PROSYD task 3.2 is to enhance the verification engines of the partners enhance to be able to reliably and robustly deal with standard digital and analog blocks occurring in today's and upcoming SoC's.

In deliverable D3.2/1, a comparison of various SAT engines using different decision heuristics was reported. In D3.2/2, ways to reduce the proof problems were explored, based on the symmetry reduction concepts of [Bri01]. D3.2/3 explored further ways to exploit RT level information for optimizing the overall proof procedure. However, the results of these latter experiments did not significantly reduce overall verification time, and hence were not implemented.

Further ideas for improvements were required, in order to meet the goals of the present report, namely to develop an improved static property checking tool, built upon the Infineon GateProp tool, and adjusted to the specifics of the PSL language. Such ideas will be presented below.

2 Technical approach

The following concepts for improving the overall model checker performance have been implemented for the Infineon tool:

- SAT improvements
- Macro machine caching
- Term substitution
- Forwarding assumptions

Details are discussed in the full version of this report.

3 Results

3.1 Test bench

We used a test-bench consisting of 800 properties from more than 20 Verilog and VHDL designs of different size, such as

- parts of the TriCore2 processor
- shift registers
- ALU
- ATM packet error handler
- various arbiters
- bus bridge
- simple counters
- two FIFO implementations
- several stack implementations
- RAM interface
- small modules designed to exercise certain PSL language features

The corresponding PSL properties cover all of Infineon's current PSL subset.

3.2 Test results

The test-bench outlined above was used to measure the total effects from the three improvements presented in section 2. The whole test-bench was executed twice:

- once for the current productive tool version CVE 3.999,
- and once for a development version incorporating all the improvements.

The test were run on a Dell Precision Workstation 650 with 2 2666 MHz CPUs and 2 GB memory, running Red Hat Linux Release 9.

The full test results are available as a table of about 60 pages. Table 1 gives a condensed view, by listing the total computation times in seconds. It shows quite substantial improvements in the pre-processing steps – these comprise all

computations except the actual SAT solving. Equally important is the reduction of the “wall-clock time” by more than 50%. This is the elapsed time as observed by an interactive user, and is obviously relevant for practical ease of use of the tool.

4 Tool manual

In the PROSYD work-plan the present deliverable is described as “Manual for Improved static property checking tool”. As outlined above, the result of the present work has been integrated into an enhanced version of the existing tool rather than into a new separate tool.

Also the improvements are transparent to the user: they contribute to the tool performance, but do not require any special options or actions from the user. Thus the user manual for the improved tool is identical to the latest version of the CVE user manual itself, and is completely independent of the work presented herein. We have therefore chosen not to submit the full tool manual, but rather this description of the work done.

There is a vast amount of documentation available for the GateProp tool, including

- a comprehensive on-line manual
- a “cook-book” with standard examples
- a tutorial
- a methodology hand-book
- a three-day training course
- a “man page”.

This documentation is available to GateProp users.

5 References

- [D3.1/1] Research report on improved decision heuristics for high-performance SAT-based static property checking. PROSYD report 2004
- [D3.2/2] Research report on improved symbolic search strategies and model reduction for static property checking. PROSYD report 2005
- [D3.2/3] Research report on exploitation of RT information in static property checking algorithms. PROSYD report 2004
- [Bri01] R. Brinkmann, Using Symmetry for Problem Reduction in Bounded Model Checking on the Register-Transfer Level, SymCon 01, Paphos, Cyprus, 2001.