Contents

List of Figures xi
List of Tables xv
Preface xvii
Acknowledgements xix

1. INTRODUCTION 1
   1 General Objective of the Book 2
   2 Summary of Contributions 5
   3 Book Outline 7

2. ESL DESIGN AND VERIFICATION 9
   1 ESL Design 9
      1.1 ESL Design Flow 10
      1.2 System Level Language SystemC 15
      1.3 Transaction Level Modeling in SystemC 16
   2 ESL Verification 18
      2.1 Simulation 19
      2.2 Formal Verification 19
      2.3 Semi-Formal Verification 20
      2.4 Verifying SystemC Models 21
         2.4.1 Simulation in SystemC 21
         2.4.2 Semi-Formal Verification in SystemC 22
         2.4.3 Formal Verification in SystemC 24
   3 Our Debugging Approach 25
      3.1 Terms 25
      3.2 General Debug Process 26
      3.3 Hierarchy of Debugging Techniques 27
3.4 SIMD Data Transfer Example

3. EARLY ERROR DETECTION

1 Deduction Techniques in a Nutshell
   1.1 Preliminaries
   1.2 Related Work

2 Static Analysis Framework
   2.1 Requirements
   2.2 General Architecture
   2.3 Generic Framework Components
      2.3.1 Generic Symbol Table
      2.3.2 Generic Dataflow Analysis
      2.3.3 Generic Structural Analysis
   2.4 Configuration and Adaptation
      2.4.1 Implementation of Analyses
      2.4.2 Tailoring to the Analyzed Language
   2.5 Approach Rating
      2.5.1 General Benefits
      2.5.2 General Limitations and Risks
      2.5.3 Benefits of REGATTA
      2.5.4 Limitations of REGATTA

3 SystemC Design Analysis System
   3.1 Implementation Effort
   3.2 Configuration and Adaptation
   3.3 Example Analysis Flow

4 Experimental Results
   4.1 SIMD Data Transfer Example Continued
   4.2 Industrial SystemC Verification Environment

5 Summary and Future Work

4. HIGH-LEVEL DEBUGGING AND EXPLORATION

1 Observation Techniques in a Nutshell
   1.1 Overview
      1.1.1 Logging
      1.1.2 Debugging
      1.1.3 Visualization
   1.2 Related Work

2 System-Level Debugging
   2.1 Requirements
   2.2 Methodology
2.2.1 Debug Levels 78
2.2.2 Debug Flow 80
2.2.3 Debugging Support 82

3 High-Level SystemC Debugging 83
  3.1 General Architecture 83
  3.2 Debug Pattern Support 84
    3.2.1 Scenario-Based Guidance 84
    3.2.2 Partially Automated Process 85
    3.2.3 Supplied Debug Patterns 86
  3.3 SystemC Debugger 86
    3.3.1 User Layer 88
    3.3.2 API Layer 88
    3.3.3 Data Layer 88
  3.4 SystemC Visualizer 91
  3.5 Implementation Issues 93

4 Experimental Results 95
  4.1 SIMD Data Transfer Example Continued 95
  4.2 Industrial Examples 97
    4.2.1 Efficiency Discussion 98
    4.2.2 SHIELD in Practice 99
    4.2.3 SHIELD Debugger vs. CoWare SystemC Shell 101

5 Summary and Future Work 103

5. LEARNING ABOUT THE DESIGN 105
  1 Induction Techniques in a Nutshell 106
    1.1 Overview 106
    1.2 Related Work 108
  2 Automatic Generation of Properties 110
    2.1 Generation Methodology 110
    2.2 Generation Algorithm 111
      2.2.1 Preliminaries 111
      2.2.2 Algorithm Description 112
      2.2.3 Complexity and Termination 114
    2.3 Design Flow Integration 115
  3 Dynamic Invariant Analysis on Simulation Traces 116
    3.1 General Architecture 117
    3.2 Basic Property Generation 118
    3.3 Property Filtering 122
    3.4 Property Encoding 123
    3.5 Complex Property Generation 123
3.6 Property Selection 126
3.7 Implementation Issues 128

4 Experimental Results 128
4.1 SIMD Data Transfer Example Continued 129
  4.1.1 Improve Design Understanding 130
  4.1.2 Detect Abnormal Behavior 131
4.2 Industrial Hardware Designs 131
  4.2.1 Generated Properties 132
  4.2.2 Generation Statistics 135
  4.2.3 Case Study: Traffic Light Control 138
  4.2.4 Case Study: SIMD MP Design 139

5 Summary and Future Work 140

6. ISOLATING FAILURE CAUSES 143
1 Experimentation Techniques in a Nutshell 143
  1.1 Overview 144
  1.2 Related Work 145

2 Automatic Isolation of Failure Causes 147
  2.1 Requirements 147
  2.2 Methodology 148
  2.3 Approach Rating 148

3 Automatic Isolation of Failure Causes in SystemC 150
  3.1 Debugging Process Schedules 150
    3.1.1 Deterministic Record/Replay Facility 152
    3.1.2 Isolating Failure Causes 154
    3.1.3 Root-Cause Analysis 156
  3.2 Debugging Program Input 156

4 Experimental Results 157
  4.1 SIMD Data Transfer Example Continued 157
  4.2 Failure Causes in Process Schedules 160
    4.2.1 Producer–Consumer Application 160
    4.2.2 Deadlock Example 161

5 Summary and Future Work 162

7. SUMMARY AND CONCLUSION 165

Appendix A. FDC Language 169
  1 FDC Syntax 169
  2 FDC Semantic 169
Appendix B. Debug Pattern Catalog

1 General Format

2 COMPETITION Pattern

3 TIMELOCK Pattern

References

List of Acronyms

Index of Symbols

Index
Debugging at the Electronic System Level
Rogin, F.; Drechsler, R.
2010, XIX, 199 p., Hardcover
ISBN: 978-90-481-9254-0