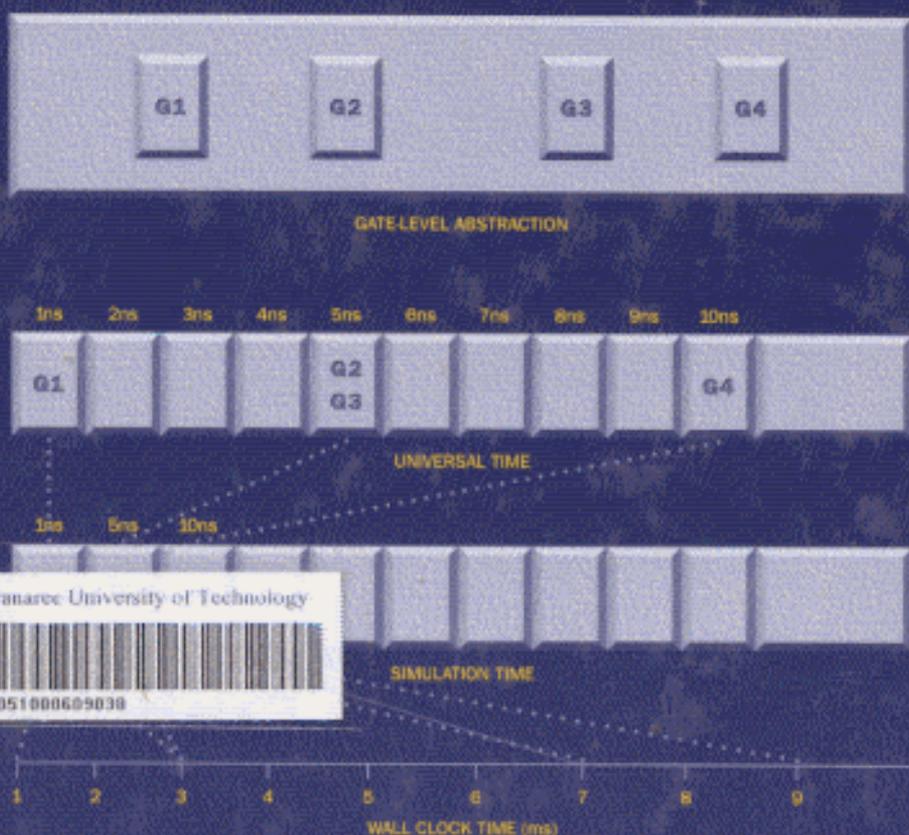


Hardware Description Languages

CONCEPTS AND PRINCIPLES

Sumit Ghosh



IEEE Press Series on
Microelectronic Systems

Stuart K. Tewksbury and Joe E. Brewer,
Series Editors

PREFACE	xiii
ACKNOWLEDGMENTS	xvii
LIST OF FIGURES	xix
LIST OF TABLES	xxvii
CHAPTER 1	The Origin of HDLs 1
1.1	Introduction 1
1.2	On the Nature of Hardware versus Software 4
1.3	The Origin and Definition of Hardware Description Languages 5
CHAPTER 2	Evolutionary Development of the Early HDLs 7
2.1	LAMP (LSI-LOCAL) 8
2.2	CDL 10
2.3	AHPL 12
2.4	ISP 13
2.5	Comparative Analysis of HDLs 15
2.6	On the Relationship between HDLs and General-Purpose Programming Languages (PLs) 17
CHAPTER 3	Fundamental Requirements of Behavior-level HDLs 19
3.1	Entity 19
3.2	Connectivity 20
3.3	Concurrency 21

3.4	Timing	23
3.4.1	Is Timing Fundamental in HDLs?	26
3.4.2	The Different Views of Time in HDLs	27
3.4.3	Timing Delays	27
3.4.4	Preemptive Scheduling: Principle, Philosophy, and Implementation	29
3.4.4.1	Introduction	29
3.4.4.2	Inadequacy of Anticipatory Timing Semantics for Reliable Behavior-level Simulation	31
3.4.4.3	Switching Behavior of Components and Input Pulse Widths	33
3.4.4.4	The Fundamental Principle of Causality as a Basis for Preemptive Scheduling	36
3.4.4.5	An Alternate Implementation of Preemptive Scheduling in HDLs	39
3.4.4.6	Comparative Analysis of the Two Implementations of the Preemptive Scheduling Principle	40
3.4.5	Asynchronous Timing Behavior and Asynchronous Interactions	41
3.4.6	Timing Constraints	41
3.5	Hierarchy	42
3.6	Consistency of HDLs with the Underlying Hardware	43

CHAPTER 4	The First Behavior-Level HDL—ADLIB-SABLE	45
4.1	Modeling Asynchronous Systems in ADLIB-SABLE	45
4.2	The Key Language Constructs of ADLIB	49
4.2.1	Assign Construct: assign (expression) to (net name) (timing clause)	49
4.2.2	Waitfor Construct: wait for (boolean expression) (control clause)	50
4.2.3	Abstract Nets	51
4.2.4	Internal Nets	51
4.3	A Key Contribution of ADLIB-SABLE	52
4.4	Limitations of ADLIB-SABLE	55
4.4.1	Comptype as an Incomplete Entity	55
4.4.2	Connectivity in ADLIB-SABLE	56
4.4.3	Concurrency in ADLIB-SABLE	56
4.4.4	Incompleteness of Timing in ADLIB-SABLE	56
4.4.5	Inability to Represent Hierarchy in ADLIB-SABLE	57
4.4.6	Unnatural and Non-intuitive ADLIB Language Constructs	58
4.5	Draft Request for Proposal (DRFP) for VHDL	60

CHAPTER 5	Verilog HDL	63
5.1	Entity	63
5.2	Connectivity	65
5.3	Timing	66
5.3.1	Timing Assignments	66

- 5.3.2 Modeling Asynchronous Timing Behavior in Verilog HDL 67
- 5.3.3 Timing Constraints 70
- 5.4 Hierarchy 72
- 5.5 Naturalness and Consistency with Underlying Hardware 72

CHAPTER 6 Design of a Concurrent HDL in Ada from First Principles 75

- 6.1 Introduction 75
- 6.2 Characteristics of an HDL 76
- 6.3 Generic Modeling 76
- 6.4 Logical Behavior, Timing, and Synchronization 77
- 6.5 Communication Among Model Instances 80
- 6.6 Distributed Scheduling 83
 - 6.6.1 Combinational Designs 84
 - 6.6.2 Sequential Designs 84
- 6.7 Hierarchical Representation and Dynamic Multi-Level Simulation (Zooming) 85
 - 6.7.1 The Principle of Zooming 85
 - 6.7.2 Zooming in Combinational Designs 86
 - 6.7.3 Zooming in Sequential Designs 87
 - 6.7.3.1 Simulation of Sequential Designs in the Absence of Zooming 87
 - 6.7.3.2 Simulation of Sequential Designs in the Presence of Zooming 89
 - 6.7.3.3 Requirements 91
 - 6.7.4 Implementation of Hierarchy and Zooming 91
 - 6.7.5 Results of Conventional Simulation versus Simulation Under Zooming 93

CHAPTER 7 VHDL 95

- 7.1 Characteristics of VHDL Relative to the Fundamental Requirements of HDLs 75
 - 7.1.1 Entity and Concurrency 95
 - 7.1.2 Connectivity 100
 - 7.1.3 Timing 100
 - 7.1.4 Hierarchy 103
 - 7.1.5 Naturalness and Consistency with Underlying Hardware 104
- 7.2 Inadequacies of VHDL and Inconsistencies with the Underlying Hardware Design 104
 - 7.2.1 The “VHDL Entity” as an Incomplete Entity and Difficulties with Concurrency 104
 - 7.2.2 The Difficulty with Connectivity in VHDL 106
 - 7.2.3 Timing Inadequacies in VHDL 108
 - 7.2.4 Limitations of Hierarchy in VHDL 115
 - 7.2.5 Need for Naturalness and Consistency with Underlying Hardware 115

- CHAPTER 8 Case Studies: Developing Hardware Descriptions for Real-world Digital Systems 117**
- 8.1 AM2903 Bit-slice Processor—A Synchronous Design 118
 - 8.2 Asynchronous VME Bus—Processor—Memory Interaction 125
 - 8.3 Modeling a VME Bus System with Multiple RAM Modules 134
- CHAPTER 9 Simulation Algorithms for Concurrent Execution of HDLs on Loosely-coupled Parallel Processors 139**
- 9.1 Review of the Literature 139
 - 9.2 An Asynchronous Approach for Simulating Behavior-level Models on Parallel Processors 143
 - 9.2.1 A Novel Asynchronous Distributed Approach to Simulation 143
 - 9.2.1.1 Simulation of Combinational Digital Designs 144
 - 9.2.1.2 Simulation of Sequential Digital Designs 150
 - 9.2.2 Implementation Issues 153
 - 9.2.3 Performance of the Algorithm 154
 - 9.2.3.1 Combinational Behavior-level Design 154
 - 9.2.3.2 Sequential Behavior-level Design 159
 - 9.3 P^2EDAS : An Algorithm for Accurate Execution of VHDL Descriptions on Parallel Processors 163
 - 9.3.1 The P^2EDAS Approach 163
 - 9.3.1.1 The Role of Event Prediction Network in P^2EDAS 164
 - 9.3.1.2 Synthesis of Event Prediction Network 167
 - 9.3.1.3 The P^2EDAS Algorithm 169
 - 9.3.2 An Example to Illustrate P^2EDAS 172
 - 9.3.3 Issues of Correctness, Freedom from Deadlock, and Termination of Simulation 179
 - 9.3.3.1 Execution of Events in the Correct Order: Monotonicity in W_i Values 179
 - 9.3.3.2 Proof of Freedom from Deadlock 183
 - 9.3.3.3 Termination of Simulation 184
 - 9.3.4 Implementation and Performance Analysis of P^2EDAS 185
 - 9.3.4.1 Performance Results for Shared Memory Multiprocessor Implementation 185
 - 9.3.4.2 Performance Results for Loosely-coupled Parallel Processor Implementation 185
 - 9.3.4.3 Performance Analysis 189
- CHAPTER 10 On the Concept of Transport Delay in HDLs 191**
- 10.1 Introduction 191
 - 10.2 Classical Methods for Transmission Line Analysis 194
 - 10.2.1 Modeling a Digital Bus 196

- 10.2.2 Representation of a Wire Junction 197
- 10.2.3 Discretizing Voltage Computations along a Transmission Line through Line-Events 198
- 10.2.4 Accuracy of Modeling Interconnects as Transmission Lines 198
- 10.3 Failure of VHDL Timing Semantics to Model Transmission Lines 200
 - 10.3.1 Transport Delays 200
 - 10.3.2 Signals 201
 - 10.3.3 Semantic Inconsistencies 202
 - 10.3.4 Issues of Logic “0” and Logic “1” on a Transmission Line 202
- 10.4 Proposed Extensions to Timing Semantics and the VHDL Grammar 203
 - 10.4.1 Extensions to the Timing Semantics 203
 - 10.4.2 Extensions to the VHDL Grammar 204
 - 10.4.2.1 Declaration of Transmission Line Signals 204
 - 10.4.2.2 Assignments to Transmission Line Signals 205
- 10.5 Modeling and Discrete Simulation of Representative Transmission Lines 208
 - 10.5.1 Simulation Algorithm for Discrete Transmission Line Analysis 209
 - 10.5.2 Modeling and Simulation of an Ethernet Wire 209
 - 10.5.2.1 Transmission Without Collisions 211
 - 10.5.2.2 Transmission Under Collisions 211
 - 10.5.3 Modeling and Simulation of a PCI Speedway Bus 214
 - 10.5.4 Modeling and Simulation of a TTL Wired-OR Bus 217
 - 10.5.4.1 Behavior Under Single Driver 217
 - 10.5.4.2 Glitches in TTL Wired-OR Bus Line Under Multiple Drivers 218

CHAPTER 11 The Future of HDLs and Philosophical Reflections 225

BIBLIOGRAPHY 229

INDEX 237

ABOUT THE AUTHOR 241