VHDL: HARDWARE DESCRIPTION AND DESIGN

VHDL: HARDWARE DESCRIPTION AND DESIGN

by

**ROGER LIPSETT
CARL F. SCHAEFER
CARY USSERY**

Intermetrics, Inc.

KLUWER ACADEMIC PUBLISHERS
Boston/Dordrecht/London

Distributors for North America:

Kluwer Academic Publishers
101 Philip Drive
Assinippi Park
Norwell, Massachusetts 02061 USA

Distributors for all other countries:

Kluwer Academic Publishers Group
Distribution Centre
Post Office Box 322
3300 AH Dordrecht, THE NETHERLANDS

Consulting Editor: Jonathan Allen, Massachusetts Institute of Technology

Library of Congress Cataloging-in-Publication Data

Lipsett, Roger, 1950-

VHDL: Hardware description and design.

Bibliography: p.

Includes index.

1. Computer input-output equipment—Computer simulation. 2. VHDL (Computer program language)

I. Schaefer, Carl F., 1945- . II. Ussery, Cary, 1962- . III. Title.

TK7887.5.L57 1989 621.39 '2 89-11114

ISBN-13: 978-1-4612-8901-2

e-ISBN-13: 978-1-4613-1631-2

DOI: 10.1007/978-1-4613-1631-2

Copyright © 1989 by Kluwer Academic Publishers

Softcover reprint of the hardcover 1st edition 1989

Fifteenth Printing 1998

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, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher, Kluwer Academic Publishers, 101 Philip Drive, Assinippi Park, Norwell, Massachusetts 02061.

Table of Contents

Foreword	xi
Preface	xv
Overview of the Book	xvi
Acknowledgements	xvii
Chapter 1 - Introduction	1
Why VHDL	2
Terminology and Conventions	5
Chapter 2 - A Model of Hardware	7
A Model of Behavior	8
A Model of Time	11
A Model of Structure	13
Chapter 3 - Basics	17
Structure and Behavior	18
Data Types and Objects	23
Data Types	23
Objects	26
Hooking Constructs Together	27
Interface Lists	28
Association Lists	29
Major VHDL Constructs	30
Entity Declarations	31
Architecture Bodies	33
Subprograms	34
Packages and Use Clauses	36
Libraries	39
Library Units and Order of Analysis	39
Visibility of a Primary Unit and Libraries	40

Chapter 4 - Data Types	45
Literals	45
Scalar Types	47
Composite Types	49
Aggregates and String Literals	51
Referencing Elements of Composites	54
Subtypes	55
Attributes	58
Predefined Operators	61
Chapter 5 - Behavioral Description	65
Process Statements	65
The Wait Statement: Activation and Suspension	66
Behavioral Modeling - Sequential View	68
Declarations	68
Sequential Assignment	69
Signal Assignment	70
Signal Drivers	73
Delay in Signal Assignments	75
Variable Assignment	82
Sequential Control	84
Conditional Control	84
Iterative Control	88
Other Sequential Statements	92
The Assertion Statement	92
Procedure Calls	94
The Return Statement	94
The Null Statement	95
Behavioral Modeling - Concurrent View	95
Concurrent Statements and Equivalent Processes	96
Concurrent Signal Assignment	96
Concurrent Assertion Statement	101
Resolved Signals	103
A Counter Element	105
Chapter 6 - Structural Description	107
Basic Features of Structural Description	108

Ports in Entity Declarations	108
Port Modes and Direction of Data Flow	109
Ports in Component Declarations	110
Component Instantiation Statements	110
Example: A Simple ALU	114
Example: A Decoder	118
Example: Data Bus	120
Regular Structures	124
Generate Statements	124
Generics	129
Configuration Specifications	131
Default Values and Unconnected Ports	138
Default Values	138
Unconnected Ports	142
Chapter 7 - Large Scale Design	145
Managing Shared Designs	146
Design Libraries and their Implementation	146
Predefined Design Libraries	147
The Use of Libraries for Revision Management	148
Visibility and the Analysis Context	149
Name Visibility in VHDL	149
Access to External VHDL Libraries	153
Partitioning a Design	155
Concurrent and Sequential Procedure Calls	156
The Block Statement	158
Component Instantiations and Blocks	159
Sharing Data Within a Design	163
Specifying a Design Configuration	167
How Component Binding Occurs	168
Type Incompatibilities in Component Binding	173
Mixing Structure and Behavior	175
Chapter 8 - A Complete Example	177
The Traffic Light Controller	178
Creating the Specification	179
Defining the System Types	179

Creating the Interface	180
The Body of the Specification	181
Creating a Test Bench	186
Partitioning the Design	188
Choosing a Type Representation	189
Revising the Specification	193
The First Partition	195
The Second Partition	198
Starting the Implementation	201
Setting Up the PLA	204
Chapter 9 - Advanced Features	209
Overloading	209
Access Types	212
File Types and I/O	216
User-Defined Attributes	219
Signal-Related Attributes	221
Aliases	224
Association by Subelement	225
Guarded Assignment Statements	227
Disconnection Specifications	228
Null Transactions	230
Chapter 10 - VHDL in Use	233
A Device Controller	234
Setup and Hold Timing	240
A Neural Net	247
A Systolic Array Multiplier	253
Summary	259
Appendix A - Predefined Environment	261
Reserved Words	261
Attributes	262
Type and Subtype Attributes	262
Array Attributes	263
Signal-Valued Attributes	264
Signal-Related Attributes	265

Packages	266
The Package STANDARD	266
The Package TEXTIO	268
Appendix B - VHDL Syntax	273
Appendix C - Suggested Reading	293
Index	295

Foreword

VHDL is a comprehensive language that allows a user to deal with design complexity. Design, and the data representing a design, are complex by the very nature of a modern digital system constructed from VLSI chips. VHDL is the first language to allow one to capture all the nuances of that complexity, and to effectively manage the data and the design process. As this book shows, VHDL is not by its nature a complex language.

In 1980, the U.S. Government launched a very aggressive effort to advance the state-of-the-art in silicon technology. The objective was to significantly enhance operating performance and circuit density for Very Large Scale Integration (VLSI) silicon chips. The U.S. Government realized that in order for contractors to be able to work together to develop VLSI products, to document the resulting designs, to be able to reuse the designs in future products, and to efficiently upgrade existing designs, they needed a common communication medium for the design data. They wanted the design descriptions to be computer readable and executable. They also recognized that with the high densities envisioned for the U.S. Government's Very High Speed Integrated Circuit (VHSIC) chips and the large systems required in future procurements, a means of streamlining the design process and managing the large volumes of design data was required. Thus was born the concept of a standard hardware design and description language to solve all of these problems.

In 1983, the U.S. Government issued a Request for Proposal to develop the language and to implement a set of tools for use with the language. The winner of the competitive bidding was a team composed of Intermetrics, International Business Machines, and Texas Instruments. As the lead engineer for the IBM part of the development effort, and as the chairman of the IEEE Computer Society Design Automation Standards Subcommittee, I saw directly the benefits that would accrue to industry from a standard hardware description language. VHDL is a descriptive language, one that is human and machine readable, so that it is well suited to preserve design specifications. VHDL is also a leading edge design language, one that may be used with many types of design tools such as simulators, design synthesizers, silicon compilers, placement and wiring tools, test generators, architectural specification and analysis tools, and timing analyzers, and can support research in all of these areas. At this time, many such tools are coming to the market place and university research in the named areas, which has been underway for years before the advent of VHDL, is proceeding to address itself to the use of VHDL. Why is this so?

As previously mentioned, a major power of VHDL is that it is a standard. Thus, industry can more easily communicate designs among participants in a design process. This ability to communicate designs is equally important in the research field, since, with VHDL, collaboration between researchers at various institutions becomes easier. But, standardization is not sufficient for the ground-swell of interest in the language. VHDL also has to be technically excellent and capable of allowing designers and researchers to describe the concepts they are developing and utilize the descriptions with tools in a way that simplifies the design or research process. VHDL achieves these goals.

The scope of VHDL covers the description of architectural description to gate level description. The language is hierarchical and mixed-level simulation is supported. The concepts embodied in the timing model for the language mirror real hardware -- the VHDL models of designs behave like real hardware. Many other excellent languages cover subsets of the capabilities that exist within VHDL, but none are as comprehensive. They do not cover the wide range covered by VHDL. Equally important, because VHDL is an IEEE standard, the language will have a significant effect on life-cycle support of products described in VHDL. At high levels of abstraction, the language makes an excellent specification medium for future designs to be created in new technologies or with alternative architectures. At lower levels of abstraction, the language serves well as a specification of what is to be fabricated.

All in all, VHDL is a "language for all seasons," one that supports design automation research, design and test, and product life-cycle. I take my hat off to the scores of people that have participated in the development of the VHDL concept and had the foresight to help bring it

to fruition. I also salute the authors of this book for helping to bring understanding in the use of VHDL.

Ronald Waxman
University of Virginia
Charlottesville, Virginia
March 20, 1989

Preface

The VHSIC Hardware Description Language (VHDL) is a new hardware description language developed, starting in 1981, by the Very High Speed Integrated Circuits (VHSIC) Program Office of the Department of Defense for use as a standard language in the microelectronics community. This language represents a new step in the evolution of language support for hardware design. The recognized need for managing the complexity of information needed for digital design has driven the development of VHDL.

The acceptance process for a new language can be slow, especially for a language as rich in features as VHDL. The largest factor in this process is the dissemination of information about what the language is and, more importantly, how to use it. To date, little information has been available to the design community to explain what VHDL is, how it relates to what designers know, and how it can be integrated into the design process.

The authors of this book have, since the beginning of the VHDL program, been involved in developing the language, building tools that support the language, and using the language to build hardware descriptions. Intermetrics, Inc., for whom the authors work, was the Government's prime contractor for the original development of VHDL. Both Intermetrics and the authors are committed to the widespread use of VHDL as a hardware description language in the design community. This book represents an opportunity to inform; to share the knowledge

of VHDL we have gained over the past five years.

This book describes the VHDL language and discusses ways in which VHDL can be used. It is intended to introduce the reader to all aspects of VHDL and to specific language constructs. It is not theoretical in nature and does not rigorously discuss the formal definition of the language; this approach is left to the Reference Manual for the language. What the book does do is take a look at the language feature by feature to provide an understanding of how each feature works and how it is integrated into the modeling process. This book is intended to give hardware designers and software model developers a thorough understanding of the VHDL language both as a language and as a design tool. We do not suggest particular ways of designing given hardware devices, but rather seek to provide the designer with an understanding of VHDL sufficient to allow him or her to best choose how to perform design.

The book is suitable for working professionals as well as graduate or advanced undergraduate study. Designers can view this book as a way to get acquainted with VHDL and as a reference for work with VHDL, while students and professors will find the book useful as a teaching tool for hardware design and hardware description languages in general, as well as VHDL in particular.

The presented material assumes a working knowledge of digital hardware design, as well as some familiarity with a high level programming language such as C, Pascal, or Ada.

Overview of the Book

The book is divided into ten chapters.

The first chapter is an introductory chapter which gives a brief history of VHDL, examines reasons for using VHDL and discusses terminology and conventions used throughout the book. The second chapter introduces the model of digital devices on which VHDL is based. It discusses the main semantic content of the language without going into the syntax of the constructs used within the model.

Chapter 3 introduces some basic elements of VHDL. These elements will be used throughout the rest of the book. Chapter 4 examines data types in the language in detail. Some readers might want to skim this chapter at first and then return to it as the sophistication of their VHDL models evolves to require the inclusion of user-defined data types.

The next three chapters, Chapters 5, 6, and 7, discuss VHDL from three viewpoints. Chapter 5 discusses using VHDL to describe the behavior of a hardware device. Chapter 6 discusses using VHDL to describe the structure of a hardware device. Chapter 7 discusses

combining these two viewpoints into a model of a large hardware device. All these views of modeling are supported by the language. These chapters introduce the reader to the bulk of the VHDL language from a functional perspective instead of from a language definition perspective.

Chapter 8 ties the concepts presented in the previous chapters into a single design example. The design is a traffic light controller which is developed from a high level specification into a PLA implementation.

Chapter 9 looks at some of the more advanced features of the language. These features are used for high level modeling, test vector input, and other specialized functions.

Chapter 10 works through a few design examples as a way of illustrating the use of VHDL in system modeling and abstract design. Other books which treat VHDL tend toward the detailed gate level or board level modeling techniques. Here we try to illustrate some of the higher level features of VHDL and its uses in abstract modeling.

Acknowledgements

Many people gave us support, advice, and commentary on the earlier drafts of this book. In particular, our peers at Intermetrics provided extensive assistance and criticism. Special thanks are due to Al Gilman, Doug Dunlop, Victor Berman, and, especially, Kathy McKinley. They were all subjected to questionable prose and countless technical errors which they handled gracefully, and with beneficial results.

We would like to express our appreciation for the support and resources provided by Intermetrics, Inc. This book was developed on three different computers in three different cities and would not have been possible without the cooperation of Intermetrics. We would especially like to thank Bill Carlson, Victor Berman and Bill Bail at Intermetrics for the direct support they have given this book.

We would like to thank Carl Harris at Kluwer Academic and Rachael Rusting at Intermetrics, whose patience and forbearance have helped to make this book a reality and to motivate the authors. Paula Gillis of Intermetrics is responsible for the cover art. Peter Connor contributed time and ideas during the early stages.

A number of people have provided valuable comments on drafts of this book. Dr. David Hemmendinger and John van Tassel of Wright State University, Capt. John Evans of the Air Force Foreign Technology Division, and Lt. Karen Serafino of the Electronic Technology Laboratory at the Wright Research and Development Center deserve special recognition for their careful review of the text and the analysis and simulation of all the examples.

Over the past five years, many people have been involved with VHDL in one form or another. While we cannot mention all of these people here, we salute the effort and creativity of those who helped to bring VHDL to life. A few individuals deserve attention for keeping VHDL alive and well over the course of the past 5 years: Allen Dewey, who was the original Project Officer for the Air Force at Wright Field; Dr. John Hines, who has been a key factor in getting VHDL going and keeping the air in its sails; and Larry Saunders who, in addition to being a major proponent of VHDL in and out of IBM, chaired the IEEE VHDL Analysis and Standardization Group which carved out the current VHDL standard.

VHDL: HARDWARE DESCRIPTION AND DESIGN