

Contents

Preface

xvii

1	Fundamental Concepts	1
1.1	Modeling Digital Systems	1
1.2	Domains and Levels of Modeling	3
1.2.1	Modeling Example	3
1.3	Modeling Languages	7
1.4	VHDL Modeling Concepts	7
1.4.1	Elements of Behavior	8
1.4.2	Elements of Structure	10
1.4.3	Mixed Structural and Behavioral Models	12
1.4.4	Test Benches	13
1.4.5	Analysis, Elaboration and Execution	14
1.5	Learning a New Language: Lexical Elements and Syntax	16
1.5.1	Lexical Elements	17
	<i>Comments</i>	17
	<i>Identifiers</i>	19
	<i>Reserved Words</i>	20
	<i>Special Symbols</i>	22
	<i>Numbers</i>	22
	<i>Characters</i>	23
	<i>Strings</i>	23
	<i>Bit Strings</i>	24
1.5.2	Syntax Descriptions	26
	Exercises	29
2	Scalar Data Types and Operations	31
2.1	Constants and Variables	31
2.1.1	Constant and Variable Declarations	31
2.1.2	Variable Assignment	33
2.2	Scalar Types	34
2.2.1	Type Declarations	34
2.2.2	Integer Types	35
2.2.3	Floating-Point Types	38
2.2.4	Physical Types	39
	<i>Time</i>	42
2.2.5	Enumeration Types	43
	<i>Characters</i>	44
	<i>Booleans</i>	46

	Bits	47
	Standard Logic	48
	Condition Conversion	49
2.3	Type Classification	51
2.3.1	Subtypes	52
2.3.2	Type Qualification	53
2.3.3	Type Conversion	54
2.4	Attributes of Scalar Types	54
2.5	Expressions and Predefined Operations	57
	Exercises	62
3	Sequential Statements	65
3.1	If Statements	65
3.1.1	Conditional Variable Assignments	68
3.2	Case Statements	69
3.2.1	Selected Variable Assignments	74
3.3	Null Statements	75
3.4	Loop Statements	76
3.4.1	Exit Statements	77
3.4.2	Next Statements	80
3.4.3	While Loops	81
3.4.4	For Loops	83
3.4.5	Summary of Loop Statements	86
3.5	Assertion and Report Statements	87
	Exercises	93
4	Composite Data Types and Operations	95
4.1	Arrays	95
4.1.1	Multidimensional Arrays	98
4.1.2	Array Aggregates	99
4.1.3	Array Attributes	103
4.2	Unconstrained Array Types	105
4.2.1	Predefined Array Types	106
	Strings	106
	Boolean Vectors, Integer Vectors, Real Vectors, and Time Vectors	106
	Bit Vectors	107
	Standard-Logic Arrays	108
	String and Bit-String Literals	108
4.2.2	Unconstrained Array Element Types	109
4.2.3	Unconstrained Array Ports	111
4.3	Array Operations and Referencing	114
4.3.1	Logical Operators	114
4.3.2	Shift Operators	116
4.3.3	Relational Operators	117
	Maximum and Minimum Operations	119
4.3.4	The Concatenation Operator	119
4.3.5	To_String Operations	120

4.3.6	Array Slices	120
4.3.7	Array Type Conversions	122
4.3.8	Arrays in Case Statements	124
4.3.9	Matching Case Statements	125
	Matching Selected Variable Assignments	127
4.4	Records	128
4.4.1	Record Aggregates	131
4.4.2	Unconstrained Record Element Types	131
	Exercises	134
5	Basic Modeling Constructs	137
5.1	Entity Declarations and Architecture Bodies	137
5.1.1	Concurrent Statements	141
5.1.2	Signal Declarations	141
5.2	Behavioral Descriptions	143
5.2.1	Signal Assignment	143
	Conditional Signal Assignments	146
	Selected Signal Assignments	147
5.2.2	Signal Attributes	149
5.2.3	Wait Statements	151
5.2.4	Delta Delays	155
5.2.5	Transport and Inertial Delay Mechanisms	158
5.2.6	Process Statements	164
5.2.7	Concurrent Signal Assignment Statements	166
	Concurrent Simple Signal Assignments	166
	Concurrent Conditional Signal Assignment	167
	Concurrent Selected Signal Assignments	171
5.2.8	Concurrent Assertion Statements	173
5.2.9	Entities and Passive Processes	174
5.3	Structural Descriptions	176
5.4	Design Processing	186
5.4.1	Analysis	186
5.4.2	Design Libraries and Contexts	188
	Context Declarations	190
5.4.3	Elaboration	193
5.4.4	Execution	195
	Exercises	197
6	Subprograms	207
6.1	Procedures	207
6.1.1	Return Statement in a Procedure	212
6.2	Procedure Parameters	213
6.2.1	Signal Parameters	217
6.2.2	Default Values	220
6.2.3	Unconstrained Array Parameters	221
6.2.4	Summary of Procedure Parameters	224
6.3	Concurrent Procedure Call Statements	225

6.4	Functions	227	
6.4.1	Functional Modeling	230	
6.4.2	Pure and Impure Functions	230	
6.4.3	The Function <code>now</code>	232	
6.5	Overloading	233	
6.5.1	Overloading Operator Symbols	234	
6.6	Visibility of Declarations	236	
	Exercises	240	
7	Packages and Use Clauses		245
7.1	Package Declarations	245	
7.1.1	Subprograms in Package Declarations	250	
7.1.2	Constants in Package Declarations	250	
7.2	Package Bodies	252	
7.2.1	Local Packages	255	
7.3	Use Clauses	257	
7.3.1	Visibility of Used Declarations	261	
	Exercises	264	
8	Resolved Signals		267
8.1	Basic Resolved Signals	267	
8.1.1	Composite Resolved Subtypes	272	
8.1.2	Summary of Resolved Subtypes	278	
8.1.3	IEEE <code>std_logic_1164</code> Resolved Subtypes	278	
8.2	Resolved Signals, Ports, and Parameters	280	
8.2.1	Resolved Ports	282	
8.2.2	Driving Value Attribute	285	
8.2.3	Resolved Signal Parameters	286	
	Exercises	287	
9	Predefined and Standard Packages		293
9.1	The Predefined Packages <code>standard</code> and <code>env</code>	293	
9.2	IEEE Standard Packages	296	
9.2.1	Standard VHDL Mathematical Packages	296	
	<i>Real Number Mathematical Package</i>	296	
	<i>Complex Number Mathematical Package</i>	299	
9.2.2	The <code>std_logic_1164</code> Multivalued Logic System	301	
9.2.3	Standard Integer Numeric Packages	304	
9.2.4	Standard Fixed-Point Packages	313	
9.2.5	Standard Floating-Point Packages	318	
9.2.6	Package Summary	322	
	<i>Operator Overloading Summary</i>	323	
	<i>Conversion Function Summary</i>	326	
	<i>Strength Reduction Function Summary</i>	334	
	Exercises	335	

10	Case Study: A Pipelined Multiplier Accumulator		337
10.1	Algorithm Outline	337	
10.2	A Behavioral Model	340	
10.2.1	Testing the Behavioral Model	342	
10.3	A Register-Transfer-Level Model	346	
10.3.1	Testing the Register-Transfer-Level Model	350	
	Exercises	353	
11	Aliases		355
11.1	Aliases for Data Objects	355	
11.2	Aliases for Non-Data Items	360	
	Exercises	363	
12	Generics		365
12.1	Generic Constants	365	
12.2	Generic Types	372	
12.3	Generic Lists in Packages	376	
12.3.1	Local Packages	381	
12.3.2	Abstract Data Types Using Packages	384	
12.4	Generic Lists in Subprograms	389	
12.5	Generic Subprograms	394	
12.6	Generic Packages	407	
	Exercises	414	
13	Components and Configurations		417
13.1	Components	417	
13.1.1	Component Declarations	417	
13.1.2	Component Instantiation	419	
13.1.3	Packaging Components	420	
13.2	Configuring Component Instances	422	
13.2.1	Basic Configuration Declarations	422	
13.2.2	Configuring Multiple Levels of Hierarchy	425	
13.2.3	Direct Instantiation of Configured Entities	428	
13.2.4	Generic and Port Maps in Configurations	429	
13.2.5	Deferred Component Binding	435	
13.3	Configuration Specifications	437	
13.3.1	Incremental Binding	438	
	Exercises	444	
14	Generate Statements		449
14.1	Generating Iterative Structures	449	
14.2	Conditionally Generating Structures	455	
14.2.1	Recursive Structures	462	
14.3	Configuration of Generate Statements	465	
	Exercises	473	

15	Access Types	479
15.1	Access Types 479	
15.1.1	Access Type Declarations and Allocators 479	
15.1.2	Assignment and Equality of Access Values 482	
15.1.3	Access Types for Records and Arrays 483	
15.2	Linked Data Structures 486	
15.2.1	Deallocation and Storage Management 490	
15.3	An Ordered-Dictionary ADT Using Access Types 491	
	Exercises 495	
16	Files and Input/Output	499
16.1	Files 499	
16.1.1	File Declarations 499	
16.1.2	Reading from Files 501	
16.1.3	Writing to Files 504	
16.1.4	Files Declared in Subprograms 507	
16.1.5	Explicit Open and Close Operations 509	
16.1.6	File Parameters in Subprograms 512	
16.1.7	Portability of Files 514	
16.2	The Package Textio 514	
16.2.1	Textio Read Operations 518	
16.2.2	Textio Write Operations 523	
16.2.3	Reading and Writing Other Types 527	
	<i>Standard Package Read and Write Operations</i> 528	
	Exercises 530	
17	Case Study: A Package for Memories	535
17.1	The Memories Package 535	
17.2	Using the Memories Package 546	
17.2.1	Common Address and Data Conversions 551	
	Exercises 558	
18	Test Bench and Verification Features	559
18.1	External Names 559	
18.2	Force and Release Assignments 570	
18.3	Embedded PSL in VHDL 575	
	Exercises 582	
19	Shared Variables and Protected Types	585
19.1	Shared Variables and Mutual Exclusion 585	
19.2	Uninstantiated Methods in Protected Types 597	
	Exercises 601	
20	Attributes and Groups	603
20.1	Predefined Attributes 603	
20.1.1	Attributes of Scalar Types 603	

20.1.2	Attributes of Array Types and Objects 604	
20.1.3	Attributes Giving Types 605	
20.1.4	Attributes of Signals 606	
20.1.5	Attributes of Named Items 607	
20.2	User-Defined Attributes 616	
20.2.1	Attribute Declarations 616	
20.2.2	Attribute Specifications 616	
20.3	Groups 628	
	Exercises 630	
21	Design for Synthesis	633
21.1	Synthesizable Subsets 633	
21.2	Use of Data Types 634	
21.2.1	Scalar Types 635	
21.2.2	Composite and Other Types 636	
21.3	Interpretation of Standard Logic Values 637	
21.4	Modeling Combinational Logic 638	
21.5	Modeling Sequential Logic 641	
21.5.1	Modeling Edge-Triggered Logic 642	
21.5.2	Level-Sensitive Logic and Inferring Storage 650	
21.5.3	Modeling State Machines 652	
21.6	Modeling Memories 654	
21.7	Synthesis Attributes 658	
21.8	Metacomments 666	
	Exercises 667	
22	Case Study: System Design Using the Gumnut Core	669
22.1	Overview of the Gumnut 669	
22.1.1	Instruction Set Architecture 669	
22.1.2	External Interface 674	
	<i>The Gumnut Entity Declaration</i> 676	
	<i>Instruction and Data Memories</i> 677	
22.2	A Behavioral Model 681	
22.2.1	The Gumnut Definitions Package 681	
22.2.2	The Gumnut Behavioral Architecture Body 687	
	<i>Overview of the Interpreter</i> 690	
	<i>Resetting the Interpreter</i> 691	
	<i>Acknowledging an Interrupt</i> 691	
	<i>Fetching an Instruction</i> 692	
	<i>Performing an Arithmetic/Logical Operation</i> 693	
	<i>Performing a Shift Operation</i> 694	
	<i>Performing a Memory-I/O Instruction</i> 695	
	<i>Performing a Branch Instruction</i> 697	
	<i>Performing a Jump Instruction</i> 697	
	<i>Performing a Miscellaneous Instruction</i> 698	
22.2.3	Verifying the Behavioral Model 699	
22.3	A Register-Transfer-Level Model 704	

- 22.3.1 The Architecture Body 706
- 22.3.2 Verifying the RTL Model 720
- 22.4 A Digital Alarm Clock 721
 - 22.4.1 System Design 722
 - 22.4.2 Synthesizing and Implementing the Alarm Clock 729
- Exercises 731

23 Miscellaneous Topics

733

- 23.1 Guards and Blocks 733
 - 23.1.1 Guarded Signals and Disconnection 733
 - The Driving Attribute* 737
 - Guarded Ports* 738
 - Guarded Signal Parameters* 739
 - 23.1.2 Blocks and Guarded Signal Assignment 739
 - Explicit Guard Signals* 742
 - Disconnection Specifications* 743
 - 23.1.3 Using Blocks for Structural Modularity 744
 - External Names and Blocks* 747
 - Generics and Ports in Blocks* 748
 - Configuring Designs with Blocks* 748
- 23.2 IP Encryption 750
 - 23.2.1 Key Exchange 769
- 23.3 VHDL Procedural Interface (VHPI) 770
 - 23.3.1 Direct Binding 771
 - 23.3.2 Tabular Registration and Indirect Binding 773
 - 23.3.3 Registration of Applications and Libraries 775
- 23.4 Postponed Processes 776
- 23.5 Conversion Functions in Association Lists 779
- 23.6 Linkage Ports 785
- Exercises 786

A Standard Packages

793

- A.1 The Predefined Package `standard` 793
- A.2 The Predefined Package `env` 797
- A.3 The Predefined Package `textio` 797
- A.4 Standard VHDL Mathematical Packages 799
 - A.4.1 The `math_real` Package 799
 - A.4.2 The `math_complex` Package 801
- A.5 The `std_logic_1164` Multivalued Logic System Package 802
- A.6 Standard Integer Numeric Packages 806
 - A.6.1 The `numeric_bit` Package 806
 - A.6.2 The `numeric_std` Package 812
 - A.6.3 The `numeric_bit_unsigned` Package 813
 - A.6.4 The `numeric_std_unsigned` Package 815
- A.7 Standard Fixed-Point Packages 816
 - A.7.1 The `fixed_float_types` Package 816
 - A.7.2 The `fixed_generic_pkg` Package 816

- A.7.3 The `fixed_pkg` Package 829
- A.8 Standard Floating-Point Packages 829
 - A.8.1 The `float_generic_pkg` Package 829
 - A.8.2 The `float_pkg` Package 840

B VHDL Syntax

841

- B.1 Design File 843
- B.2 Library Unit Declarations 843
- B.3 Declarations and Specifications 845
- B.4 Type Definitions 848
- B.5 Concurrent Statements 850
- B.6 Sequential Statements 852
- B.7 Interfaces and Associations 855
- B.8 Expressions and Names 856

C Answers to Exercises

859

References

889

Index

891