1. Record Nr. UNINA9910809174903321

Autore	Rushton Andrew
Titolo	VHDL for logic synthesis / / Andrew Rushton
Pubbl/distr/stampa	Chichester, West Sussex, U.K., : Wiley, 2011
ISBN	9786613373892 9781119995739 1119995736 9780470977972 0470977973 9781283373890 1283373890 9780470977927 0470977922 9781119995852 111999585X
Edizione	[3rd ed.]
Descrizione fisica	1 online resource (486 p.)
Classificazione	COM059000
Disciplina	621.39/5
Soggetti	VHDL (Computer hardware description language) Logic design - Data processing Computer-aided design
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Description based upon print version of record.
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	VHDL FOR LOGIC SYNTHESIS; Contents; Preface; List of Figures; List of Tables; 1 Introduction; 1.1 The VHDL Design Cycle; 1.2 The Origins of VHDL; 1.3 The Standardisation Process; 1.4 Unification of VHDL Standards; 1.5 Portability; 2 Register-Transfer Level Design; 2.1 The RTL Design Stages; 2.2 Example Circuit; 2.3 Identify the Data Operations; 2.4 Determine the Data Precision; 2.5 Choose Resources to Provide; 2.6 Allocate Operations to Resources; 2.7 Design the Controller; 2.8 Design the Reset Mechanism; 2.9 VHDL Description of the RTL Design; 2.10 Synthesis Results; 3 Combinational Logic 3.1 Design Units 3.2 Entities and Architectures; 3.3 Simulation Model; 3.4 Synthesis Templates; 3.5 Signals and Ports; 3.6 Initial Values; 3.7

Simple Signal Assignments; 3.8 Conditional Signal Assignments; 3.9 Selected Signal Assignment; 3.10 Worked Example; 4 Basic Types; 4.1 Synthesisable Types; 4.2 Standard Types; 4.3 Standard Operators; 4.4 Type Bit; 4.5 Type Boolean; 4.6 Integer Types; 4.7 Enumeration Types; 4.8 Multi-Valued Logic Types; 4.9 Records; 4.10 Arrays; 4.11 Aggregates, Strings and Bit-Strings; 4.12 Attributes; 4.13 More on Selected Signal Assignments; 5 Operators
5.1 The Standard Operators 5.2 Operator Precedence; 5.3 Boolean Operators; 5.4 Comparison Operators; 5.5 Shifting Operators; 5.6 Arithmetic Operators; 5.7 Concatenation Operator; 6 Synthesis Types; 6.1 Synthesis Type System; 6.2 Making the Packages Visible; 6.3 Logic Types - Std_Logic_1164; 6.4 Numeric Types - Numeric_Std; 6.5 Fixed-Point Types - Fixed_Pkg; 6.6 Floating-Point Types - Float_Pkg; 6.7 Type Conversions; 6.8 Constant Values; 6.9 Mixing Types in Expressions; 6.10 Top-Level Interface; 7 Std_Logic_Arith; 7.1 The Std_Logic_Arith Package; 7.2 Contents of Std_Logic_Arith
7.3 Type Conversions 7.4 Constant Values; 7.5 Mixing Types in Expressions; 8 Sequential VHDL; 8.1 Processes; 8.2 Signal Assignments; 8.3 Variables; 8.4 If Statements; 8.5 Case Statements; 8.6 Latch Inference; 8.7 Loops; 8.8 Worked Example; 9 Registers; 9.1 Basic D-Type Register; 9.2 Simulation Model; 9.3 Synthesis Model; 9.4 Register Templates; 9.5 Register Types; 9.6 Clock Types; 9.7 Clock Gating; 9.8 Data Gating; 9.9 Asynchronous Reset; 9.10 Synchronous Reset; 9.11 Registered Variables; 9.12 Initial Values; 10 Hierarchy; 10.1 The Role of Components; 10.2 Indirect Binding; 10.3 Direct Binding
10.4 Component Packages 10.5 Parameterised Components; 10.6 Generate Statements; 10.7 Worked Examples; 11 Subprograms; 11.1 The Role of Subprograms; 11.2 Functions; 11.3 Operators; 11.4 Type Conversions; 11.5 Procedures; 11.6 Declaring Subprograms; 11.7 Worked Example; 12 Special Structures; 12.1 Tristates; 12.2 Finite State Machines; 12.3 RAMs and Register Banks; 12.4 Decoders and ROMs; 13 Test Benches; 13.1 Test Benches; 13.2 Combinational Test Bench; 13.3 Verifying Responses; 13.4 Clocks and Resets; 13.5 Other Standard Types; 13.6 Don't Care Outputs; 13.7 Printing Response Values
13.8 Using TextIO to Read Data Files

Sommario/riassunto

Making VHDL a simple and easy-to-use hardware description language Many engineers encountering VHDL (very high speed integrated circuits hardware description language) for the first time can feel overwhelmed by it. This book bridges the gap between the VHDL language and the hardware that results from logic synthesis with clear organisation, progressing from the basics of combinational logic, types, and operators; through special structures such as tristate buses, register banks and memories, to advanced themes such as developing your own packages, writing test benches and using the