

1. Record Nr.	UNINA9910491025803321
Autore	Mehta Ashok B.
Titolo	Introduction to SystemVerilog / / Ashok B. Mehta
Pubbl/distr/stampa	Cham, Switzerland : , : Springer, , [2021] ©2021
ISBN	3-030-71319-9
Descrizione fisica	1 online resource (866 pages)
Disciplina	621.392
Soggetti	SystemVerilog (Computer hardware description language)
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	Intro -- Foreword -- Preface -- Recommended Books to Supplement the Material Presented in this Book -- Chapters of the Book -- Acknowledgments -- Contents -- List of Figures -- List of Tables -- About the Author -- Chapter 1: Introduction -- 1.1 SystemVerilog Language Evolution -- Chapter 2: Data Types -- 2.1 Integer Data Types -- 2.1.1 Integer, int, longint, shortint, logic, byte, reg -- 2.1.2 Signed Types -- 2.1.3 Bits vs. Bytes -- 2.2 Real Data Types -- 2.2.1 "real" Data-Type Conversion Functions -- 2.3 Nets -- 2.3.1 "wire" and "tri" -- 2.3.2 Unresolved "wire" Type: "uwire" -- 2.3.3 Resolved vs. Unresolved Type -- 2.3.4 "wand" and "triand" -- 2.3.5 "wor" and "trior" -- 2.3.6 "tri0" and "tri1" -- 2.4 Drive Strengths -- 2.5 Variable vs. Net -- 2.6 "var" -- 2.7 Variable and Net Initialization -- 2.8 Static, Automatic, and Local Variables -- 2.8.1 Static vs. Local Variables -- 2.8.2 Automatic vs. Static Variable -- 2.8.3 Variable Lifetimes -- 2.9 Enumerated Types -- 2.9.1 Enumerated-Type Methods -- 2.9.2 Enumerated Type with Ranges -- 2.10 User-Defined Type: Typedef -- 2.11 String Data Type -- 2.11.1 String Operators -- 2.11.2 String Methods -- 2.12 Event Data Type -- 2.12.1 Event Sequencing: wait_order () -- 2.13 Static Casting -- 2.13.1 Bit-Stream Casting -- 2.14 Dynamic Casting -- Chapter 3: Arrays -- 3.1 Packed and Unpacked Arrays -- 3.1.1 2-D Packed Array -- 3.1.2 3-D Packed Array -- 3.1.3 1-D Packed and 1-D Unpacked Array -- 3.1.3.1 Multidimensional Arrays -- 3.1.4 4-D Unpacked Array -- 3.1.5 1-D Packed and 3-D Unpacked Array -- 3.1.6 2-D Packed and 2D-

Unpacked Array -- 3.1.7 3-D Packed and 1-D Unpacked Array -- 3.2 Assigning, Indexing, and Slicing of Arrays -- 3.2.1 Packed and Unpacked Arrays as Arguments to Subroutines -- 3.3 Dynamic Arrays -- 3.3.1 Dynamic Arrays - Resizing -- 3.3.2 Copying of Dynamic Arrays. 3.3.3 Dynamic Array of Arrays -- 3.4 Associative Arrays -- 3.4.1 Wild Card Index -- 3.4.2 String Index -- 3.4.3 Class Index -- 3.4.4 String Index - Example -- 3.4.5 Associative Array Methods -- 3.4.6 Associative Array - Default Value -- 3.4.7 Creating a Dynamic Array of Associative Arrays -- 3.5 Array Manipulation Methods -- 3.5.1 Array Locator Methods -- 3.5.2 Array Ordering Methods -- 3.5.3 Array Reduction Methods -- Chapter 4: Queues -- 4.1 Queue Methods -- 4.2 Queue of SystemVerilog Classes -- 4.3 Queue of Queues: Dynamic Array of Queues -- Chapter 5: Structures -- 5.1 Packed Structure -- 5.2 Unpacked Structure -- 5.3 Structure as Module I/O -- 5.4 Structure as an Argument to Task or Function -- 5.5 Structure Within a Structure -- Chapter 6: Union -- 6.1 Packed and Unpacked Unions -- 6.1.1 Unpacked Unions -- 6.1.2 Tagged Unions -- 6.1.3 Packed Union -- Chapter 7: Packages -- Chapter 8: Class -- 8.1 Basics -- 8.2 Base Class -- 8.3 Extended Class and Inheritance -- 8.3.1 Inheritance Memory Allocation -- 8.4 Class Constructor -- 8.4.1 Base Class Constructor -- 8.4.2 Chain Constructor (super.new()) -- 8.5 Static Properties -- 8.6 Static Methods -- 8.7 "this" -- 8.8 Class Assignment -- 8.9 Shallow Copy -- 8.10 Deep Copy -- 8.11 Upcasting and Downcasting -- 8.12 Polymorphism -- 8.13 Virtual Methods, Pure Virtual Methods, and Virtual Classes -- 8.13.1 Virtual Methods -- 8.13.2 Virtual (Abstract) Class and Pure Virtual Method -- 8.14 Data Hiding ("local," "protected," "const") and Encapsulation -- 8.14.1 Local Members -- 8.14.2 Protected Members -- 8.14.3 "const" Class Properties -- 8.15 Class Scope Resolution Operator (::) and "extern" -- 8.16 Parameterized Class -- 8.16.1 Value Parameters -- 8.16.2 Type Parameters -- 8.16.3 Parameterized Class with Static Properties -- 8.16.4 Extending Parameterized Class -- 8.17 Difference Between Class and Struct. Chapter 9: SystemVerilog "module" -- 9.1 Module Header Definition -- 9.1.1 ANSI-Style Module Header -- 9.1.2 ANSI-Style first Port Rules -- 9.1.3 ANSI-Style subsequent Port Rules -- 9.1.4 Non-ANSI-Style Module Header -- 9.2 Default Port Values -- 9.3 Module Instantiation -- 9.3.1 root -- 9.4 Nested Modules -- 9.5 Module Parameters -- 9.5.1 Overriding Module Parameters: defparam -- 9.5.2 Overriding Module Parameters: Module Instance Parameter Value Assignment -- 9.5.3 Localparam -- 9.5.4 Parameter Dependence -- Chapter 10: SystemVerilog "program" -- 10.1 Eliminating Testbench Races -- Chapter 11: SystemVerilog "interface" -- 11.1 Interfaces -- 11.2 Modports -- 11.3 Tasks and Functions in an Interface -- 11.3.1 "import" Tasks in a Modport -- 11.3.2 "export" Tasks in a Modport -- 11.4 Parameterized Interface -- 11.5 Clocking Block in an Interface -- Chapter 12: Operators -- 12.1 Assignment Operators -- 12.2 Increment and Decrement Operators -- 12.3 Arithmetic Operators -- 12.4 Relational Operators -- 12.5 Equality Operators -- 12.5.1 Wildcard Equality Operators -- 12.6 Logical Operators -- 12.7 Bitwise Operators -- 12.8 Unary Reduction Operators -- 12.9 Shift Operators -- 12.10 Conditional Operators -- 12.11 Concatenation Operators -- 12.12 Replication Operators -- 12.13 Streaming Operators (pack/unpack) -- 12.13.1 Packing of Bits -- 12.13.2 Unpacking of Bits -- 12.14 "inside" Operator (Set Membership Operator) -- Chapter 13: Constrained Random Test Generation and Verification -- 13.1 Productivity Gain with CRV -- 13.2 Constrained Random Verification

(CRV) Methodology -- 13.3 SystemVerilog Support for CRV -- 13.4 Constraints -- 13.4.1 Constraints: Turning On and OFF -- 13.5 Random Variables (rand and randc) -- 13.5.1 Static Random Variables -- 13.5.2 Randomizing Arrays and Queues -- 13.5.3 Randomizing Object Handles -- 13.6 Constraint Blocks. 13.6.1 External Constraint Blocks -- 13.6.2 Weighted Distribution -- 13.6.3 "unique" Constraint -- 13.6.4 Implication and If-Else -- 13.6.5 Iterative Constraint (foreach) -- 13.6.6 Array Reduction Methods for Constraint -- 13.6.7 Functions in Constraints -- 13.6.8 Soft Constraints -- 13.7 Randomization Methods -- 13.7.1 Pre-randomization and Post-randomization -- 13.7.2 Local Scope Resolution (local::) -- 13.8 rand_mode(): Disabling Random Variables -- 13.9 constraint_mode(): Control Constraints -- 13.10 randomize() with Arguments: In-Line Random Variable Control -- 13.11 Random Number Generation System Functions and Methods -- 13.11.1 Random Number Generator (RNG) -- 13.11.2 urandom() and urandom_range() -- 13.11.3 srandom(), get_randstate(), and set_randstate() -- 13.12 Random Stability -- 13.13 Randcase -- 13.14 randsequence -- 13.14.1 Random Production Weights and If-Else Statement -- 13.14.2 Repeat Production Statement -- 13.14.3 rand join -- 13.14.4 break and return -- 13.14.5 Passing Values Between Productions -- 13.14.5.1 Passing Return Value -- Chapter 14: SystemVerilog Assertions -- 14.1 SystemVerilog Assertions Evolution -- 14.2 What Is an Assertion? -- 14.3 Why Assertions? What Are the Advantages? -- 14.3.1 Assertions Shorten Time to Develop -- 14.3.2 Assertions Improve Observability -- 14.3.3 Other Major Benefits -- 14.3.4 One-Time Effort, Many Benefits -- 14.4 Assertions in Static Formal -- 14.5 Methodology Components -- 14.5.1 Types of Assertions to Add -- 14.5.2 How to Add Assertions? What Is the Protocol? -- 14.5.3 How Do I Know I Have Enough Assertions? -- 14.5.4 A Simple PCI Read Example: Creating an Assertion Test Plan -- 14.6 Assertion Types -- 14.7 Conventions Used in This Chapter -- 14.8 Immediate Assertions -- 14.8.1 Deferred Immediate Assertions -- 14.9 Concurrent Assertions: Basics -- 14.9.1 Implication Operator. 14.10 Clocking Basics -- 14.10.1 Default Clocking Block -- 14.10.2 Sampling Edge (Clock Edge) -- 14.10.3 Active Region -- 14.10.4 Observed Region -- 14.10.5 Reactive Region -- 14.10.6 Preponed Region -- 14.11 Concurrent Assertions Are Multi-threaded -- 14.12 Formal Arguments -- 14.13 Disable (Property) Operator: disable iff -- 14.14 Severity Levels -- 14.15 Binding Properties -- 14.15.1 Binding Properties (Scope Visibility) -- 14.15.2 VHDL DUT Binding with SystemVerilog Assertions -- 14.16 Difference Between "sequence" and "property" -- 14.17 Sampled Value Functions -- 14.17.1 rose: Edge Detection in Property/Sequence -- 14.17.2 fell: Edge Detection in Property/Sequence -- 14.17.3 Edge Detection Is Useful Because... -- 14.17.4 stable -- 14.17.5 past -- 14.17.5.1 Application: past() -- 14.18 Operators -- 14.18.1 ##m: Clock Delay -- 14.18.2 ##[m:n]: Clock Delay Range -- 14.18.2.1 Clock Delay Range Operator: ##[m:n] - False Positive -- 14.18.3 [*m]: Consecutive Repetition Operator -- 14.18.4 [*m:n]: Consecutive Repetition Range Operator -- 14.18.4.1 Applications: Consecutive Repetition Range Operator -- 14.18.5 [=m]: Non-consecutive Repetition -- 14.18.6 [=m:n]: Non-consecutive Repetition Range Operator -- 14.18.6.1 Application: Non-consecutive Repetition Operator -- 14.18.7 [-> --] Non-consecutive GoTo Repetition Operator -- 14.18.8 Difference Between [=m:n] and [-> -- m:n] -- 14.18.9 Application: GoTo Repetition - Non-consecutive Operator -- 14.18.10 Sig1 throughout Seq1 -- 14.18.11 Seq1 within Seq2 -- 14.18.12 Application: Seq1 "within" Seq2 -- 14.18.13 Seq1

and Seq2 -- 14.18.14 Application: "and" Operator -- 14.18.15 Seq1 or
Seq2 -- 14.18.16 Seq1 "intersect" Seq2 -- 14.18.17 Application:
"intersect" Operator -- 14.18.18 first_match -- 14.18.19 Application:
first_match -- 14.18.20 not Operator.
14.18.21 if (Expression) property_expr1 Else property_expr2.
