Universal Verification Methodology Uvm Based Random

by Phil Moorby The Verilog Hardware Description Language has had an amazing impact on the mod em electronics industry, considering that the essential composition of the language was developed in a surprisingly short period of time, early in 1984. Since its introduc tion, Verilog has changed very little. Over time, users have requested many improve ments to meet new methodology needs. But, it is a complex and time consuming process to add features to a language without ambiguity, and maintaining consistency. A group of Verilog enthusiasts, the IEEE 1364 Verilog committee, have broken the Verilog feature doldrums. These individuals should be applauded. They invested the time and energy, often their personal time, to understand and resolve an extensive wish-list of language enhancements. They took on the task of choosing a feature set that would stand up to the scrutiny of the standardization process. I would like to per sonally thank this group. They have shown that it is possible to evolve Verilog, rather than having to completely start over with some revolutionary new language. The Verilog 1364-2001 standard provides many of the advanced building blocks that users have requested. The enhancements include key components for verification, abstract design, and other new methodology capabilities. As designers tackle advanced issues such as automated verification, system partitioning, etc., the Verilog standard will rise to meet the continuing challenge of electronics design.
This is a workbook for Universal Verification Methodology
Top-Down VLSI Design: From Architectures to Gate-Level Circuits and FPGAs represents a unique approach to learning digital design. Developed from more than 20 years teaching circuit design, Doctor Kaeslin’s approach follows the natural VLSI design flow and makes circuit design accessible for professionals with a background in systems engineering or digital signal processing. It begins with hardware architecture and promotes a system-level view, first considering the type of intended application and letting that guide your design choices. Doctor Kaeslin presents modern considerations for handling circuit complexity, throughput, and energy efficiency while preserving functionality. The book focuses on application-specific integrated circuits (ASICs), which along with FPGAs are increasingly used to develop products with applications in telecommunications, IT security, biomedical, automotive, and computer vision industries. Topics include field-programmable logic, algorithms, verification, modeling hardware, synchronous clocking, and more. Demonstrates a top-down approach to digital VLSI design. Provides a systematic overview of architecture optimization techniques. Features a chapter on field-programmable logic devices, their technologies and architectures. Includes checklists, hints, and warnings for various design situations. Emphasizes design flows that do not overlook important action items and which include alternative options when planning the development of microelectronic circuits.
Since its introduction in 2011, the Universal Verification Methodology (UVM) has achieved its promise of becoming the dominant platform for semiconductor design verification. Advanced UVM delivers proven coding guidelines, convenient recipes for common tasks, and cutting-edge techniques to provide a framework within UVM. Once adopted by an organization, these strategies will create immediate benefits, and help verification teams develop scalable, high-performance environments and maximize their productivity. The second edition updates the chained sequencer, re-organizes the content, and has a few minor corrections. "Written by an experienced UVM practitioner, this book contains lots of great tips on using UVM effectively and example code that actually works!" John Aynsley, Doulos "In 'Advanced UVM', Mr. Hunter, based on his company's real world experiences, provides excellent resources, a well-tested reference verification environment, and advanced best practices on how to apply UVM. If you are ready to move beyond a UVM introduction, this should be the book you add to your library." George Taglieri, Director Verification Product Solutions, Synopsys, Inc.
The Universal Verification Methodology (UVM) package is an open-source SystemVerilog library, which is used to set up a class-based hierarchical testbench. UVM testbenches improve the reusability of Verilog testbenches. Direct Memory Access (DMA) plays an important role in modern computer architecture. When using DMA to transfer data between a host machine and field-programmable gate array (FPGA) accelerator, a modularized DMA core on the FPGA frees the host side Central Processing Unit(CPU) during the transfer, helps to save FPGA resources, and enhances performance. Verifying the functionality of a DMA core is essential before mapping it to the FPGA. In this thesis, we tested an open source DMA core with UVM (Universal Verification Methodology). Bus agents and interface modules are designed for input and output signals of the DMA Design Under Test (DUT). We constructed a Register Level Abstraction (RLA) model to allow both front-door access and back-door access to the register files in the DUT. We designed the sequences, scoreboards, and tests with features to allow reuse. The overall testbench structure is defined by a base-type test. Different tests then extend the base-type test and use type overriding with the UVM configuration database to use different scoreboards and sequences accordingly. With scoreboard and coverage groups, the testbench monitors the correctness of the behavior of the DMA DUT, as well as the functional coverage of all tests. We performed the simulations with the Questa simulator. Several bugs in the open-source DMA core were found and corrected.
Ever increasing silicon design complexity and transistor density, product differentiation and time to market are major factors creating huge pressure on complete design flow. This book covers Verification phase by describing the concepts of Universal Verification Methodology (UVM) and by presenting a pragmatic approach of developing efficient and unified advanced verification environment at all levels using Universal Verification Methodology along with Assertion based verification, hardware acceleration and Transaction Level Modeling. This book is written primarily for verification engineers performing verification of complex IP blocks or entire system-on-chip (SoC) designs. However, much of material will also be of interest to SoC project managers as well as designers to learn more about verification. Furthermore, this book includes detailed information about verification environment for one case which can be easily used as reference for other cases.
The conference aims at providing a platform for researchers, engineers, academics and industrial professionals to present their recent research work and to explore future trends in various areas of engineering and technology
This book is a comprehensive guide to assertion-based verification of hardware designs using System Verilog Assertions (SVA). It enables readers to minimize the cost of verification by using assertion-based techniques in simulation testing, coverage collection and formal analysis. The book provides detailed descriptions of all the language features of SVA, accompanied by step-by-

step examples of how to employ them to construct powerful and reusable sets of properties. The book also shows how SVA fits into the broader System Verilog language, demonstrating the ways that assertions can interact with other System Verilog components. The reader new to hardware verification will benefit from general material describing the nature of design models and behaviors, how they are exercised, and the different roles that assertions play. This second edition covers the features introduced by the recent IEEE 1800-2012. System Verilog standard, explaining in detail the new and enhanced assertion constructs. The book makes SVA usable and accessible for hardware designers, verification engineers, formal verification specialists and EDA tool developers. With numerous exercises, ranging in depth and difficulty, the book is also suitable as a text for students.

Functional verification is an art as much as a science. It requires not only creativity and cunning, but also a clear methodology to approach the problem. The Open Verification Methodology (OVM) is a leading-edge methodology for verifying designs at multiple levels of abstraction. It brings together ideas from electrical, systems, and software engineering to provide a complete methodology for verifying large scale System-on-Chip (SoC) designs. OVM defines an approach for developing testbench architectures so they are modular, configurable, and reusable. This book is designed to help both novice and experienced verification engineers master the OVM through extensive examples. It describes basic verification principles and explains the essentials of transaction-level modeling (TLM). It leads readers from a simple connection of a producer and a consumer through complete self-checking testbenches. It explains construction techniques for building configurable, reusable testbench components and how to use TLM to communicate between them. Elements such as agents and sequences are explained in detail.

From past decades, Computational intelligence embraces a number of nature-inspired computational techniques which mainly encompasses fuzzy sets, genetic algorithms, artificial neural networks and hybrid neuro-fuzzy systems to address the computational complexities such as uncertainties, vagueness and stochastic nature of various computational problems practically. At the same time, Intelligent Control systems are emerging as an innovative methodology which is inspired by various computational intelligence process to promote a control over the systems without the use of any mathematical models. To address the effective use of intelligent control in Computational intelligence systems, International Conference on Intelligent Computing, Information and Control Systems (ICICCS 2019) is initiated to encompass the various research works that helps to develop and advance the next-generation intelligent computing and control systems. This book integrates the computational intelligence and intelligent control systems to provide a powerful methodology for a wide range of data analytics issues in industries and societal applications. The recent research advances in computational intelligence and control systems are addressed, which provide very promising results in various industry, business and societal studies. This book also presents the new algorithms and methodologies for promoting advances in common intelligent computing and control methodologies including evolutionary computation, artificial life, virtual infrastructures, fuzzy logic, artificial immune systems, neural networks and various neuro-hybrid methodologies. This book will be pragmatic for researchers, academicians and students dealing with mathematically intransigent problems. It is intended for both academicians and researchers in the field of Intelligent Computing, Information and Control Systems, along with the distinctive readers in the fields of computational and artificial intelligence to gain more knowledge on Intelligent computing and control systems and their real-world applications.

This book provides a hands-on, application-oriented guide to the entire IEEE standard 1800 SystemVerilog language. Readers will benefit from the step-by-step approach to learning the language and methodology nuances, which will enable them to design and verify complex ASIC/SoC and CPU chips. The author covers the entire spectrum of the language, including random constraints, SystemVerilog Assertions, Functional Coverage, Class, checkers, interfaces, and Data Types, among other features of the language. Written by an experienced, professional end-user of ASIC/SoC/CPU and FPGA designs, this book explains each concept with easy to understand examples, simulation logs and applications derived from real projects. Readers will be empowered to tackle the complex task of multi-million gate ASIC designs. Provides comprehensive coverage of the entire IEEE standard SystemVerilog language; Covers important topics such as constrained random verification, SystemVerilog Class, Assertions, Functional coverage, data types, checkers, interfaces, processes and procedures, among other language features; Uses easy to understand examples and simulation logs; examples are simulatable and will be provided online; Written by an experienced, professional end-user of ASIC/SoC/CPU and FPGA designs. This is quite a comprehensive work. It must have taken a long time to write it. I really like that the author has taken apart each of the SystemVerilog constructs and talks about them in great detail, including example code and simulation logs. For example, there is a chapter dedicated to arrays, and another dedicated to queues - that is great to have! The Language Reference Manual (LRM) is quite dense and difficult to use as a text for learning the language. This book explains semantics at a level of detail that is not possible in an LRM. This is the strength of the book. This will be an excellent book for novice users and as a handy reference for experienced programmers. Mark Glasser Cerebras Systems.

Connectivity verification on a complex and large-scale chip can be a daunting and one of the crucial tasks to ensure the silicon to work on the first attempt of fabrication. Mixed-signal connectivity originates at a digital control bit and terminates at an analog node. Connectivity is verified on the entire hierarchy and not just at the boundary of analog and digital domains. The Verilog-AMS framework adopted for this methodology provides the ease of using a SystemVerilog or a Universal Verification Methodology (UVM) based testbench that is widely used for verification of the digital design. This methodology supports connectivity verification in an existing testbench environment without the need of analog-digital co-simulation setup.

This book is an “A-Z” guide to using SystemVerilog for ASIC design, from conception to RTL coding, to synthesis and verification. Readers will benefit from a thorough

introduction to the powerful constructs and features of SystemVerilog. In addition, the verification methodology of Universal Verification Methodology (UVM) is used to build test-benches that allow for verification of complicated designs and synthesis basics are discussed, using the Synopsys Design Compiler (DC). To complete this book's package as a practical guide, readers are introduced to the fundamentals of static timing analysis.

The Accellera Universal Verification Methodology (UVM) standard is architected to scale, but verification is growing and in more than just the digital design dimension. It is growing in the SoC dimension to include low-power and mixed-signal and the system integration dimension to include multi-language support and acceleration. These items and others all contribute to the quality of the SOC so the Metric-Driven Verification (MDV) methodology is needed to unify it all into a coherent verification plan. This book is for verification engineers and managers familiar with the UVM and the benefits it brings to digital verification but who also need to tackle specialized tasks. It is also written for the SoC project manager that is tasked with building an efficient worldwide team. While the task continues to become more complex, Advanced Verification Topics describes methodologies outside of the Accellera UVM standard, but that build on it, to provide a way for SoC teams to stay productive and profitable.

SystemVerilog language consists of three categories of features -- Design, Assertions and Testbench. Assertions add a whole new dimension to the ASIC verification process. Engineers are used to writing testbenches in verilog that help verify their design. Verilog is a procedural language and is very limited in capabilities to handle the complex ASICs built today. SystemVerilog assertions (SVA) is a declarative language. The temporal nature of the language provides excellent control over time and allows mulitple processes to execute simultaneously. This provides the engineers a very strong tool to solve their verification problems. The language is still new and the thinking is very different from the user's perspective when compared to standard verilog language. There is not enough expertise or intellectual property available as of today in the field. While the language has been defined very well, there is no practical guide that shows how to use the language to solve real verification problems. This book is a practical guide that will help people to understand this new language and adopt assertion based verification methodology quickly.

All areas of Computer Science Engineering, Electrical Engineering, as well as Electronics and Communication Engineering

SystemVerilog is a rich set of extensions to the IEEE 1364-2001 Verilog Hardware Description Language (Verilog HDL). These extensions address two major aspects of HDL based design. First, modeling very large designs with concise, accurate, and intuitive code. Second, writing high-level test programs to efficiently and effectively verify these large designs. This book, SystemVerilog for Design, addresses the first aspect of the SystemVerilog extensions to Verilog. Important modeling features are presented, such as two-state data types, enumerated types, user-defined types, structures, unions, and interfaces. Emphasis is placed on the proper usage of these enhancements for simulation and synthesis. A companion to this book, SystemVerilog for Verification, covers the second aspect of SystemVerilog.

Based on the highly successful second edition, this extended edition of SystemVerilog for Verification: A Guide to Learning the Testbench Language Features teaches all verification features of the SystemVerilog language, providing hundreds of examples to clearly explain the concepts and basic fundamentals. It contains materials for both the full-time verification engineer and the student learning this valuable skill. In the third edition, authors Chris Spear and Greg Tumbush start with how to verify a design, and then use that context to demonstrate the language features, including the advantages and disadvantages of different styles, allowing readers to choose between alternatives. This textbook contains end-of-chapter exercises designed to enhance students' understanding of the material. Other features of this revision include: New sections on static variables, print specifiers, and DPI from the 2009 IEEE language standard Descriptions of UVM features such as factories, the test registry, and the configuration database Expanded code samples and explanations Numerous samples that have been tested on the major SystemVerilog simulators SystemVerilog for Verification: A Guide to Learning the Testbench Language Features, Third Edition is suitable for use in a one-semester SystemVerilog course on SystemVerilog at the undergraduate or graduate level. Many of the improvements to this new edition were compiled through feedback provided from hundreds of readers.

This book provides a comprehensive overview of the VLSI design process. It covers end-to-end system on chip (SoC) design, including design methodology, the design environment, tools, choice of design components, handoff procedures, and design infrastructure needs. The book also offers critical guidance on the latest UPF-based low power design flow issues for deep submicron SOC designs, which will prepare readers for the challenges of working at the nanotechnology scale. This practical guide will provide engineers who aspire to be VLSI designers with the techniques and tools of the trade, and will also be a valuable professional reference for those already working in VLSI design and verification with a focus on complex SoC designs. A comprehensive practical guide for VLSI designers; Covers end-to-end VLSI SoC design flow; Includes source code, case studies, and application examples.

This book concentrates on common classes of hardware architectures and design problems, and focuses on the process of transitioning design requirements into synthesizable HDL code. Using his extensive, wide-ranging experience in computer architecture and hardware design, as well as in his training and consulting work, Ben provides numerous examples of real-life designs illustrated with VHDL and Verilog code. This code is shown in a way that makes it easy for the reader to gain a greater understanding of the languages and how they compare. All code presented in the book is included on the companion CD, along with other information, such as application notes.

The field of application-specific integrated circuits (ASICs) is fast-paced being at the very forefront of modern nanoscale fabrication and presents a deeply engaging career path. ASICs can provide us with high-speed computation in the case of digital circuits. For example, central processing units, graphics processing units, field-programmable gate arrays, and custom-made digital signal processors are examples of ASICs and the transistors they are fabricated from. We can use that same technology complementary metal-oxide semiconductor processes to implement high-precision sensing of or interfacing to the world through analog-to-digital converters, digital-to-analog converters, custom image sensors, and highly integrated micron-scale sensors such as magnetometers, accelerometers, and microelectromechanical machines. ASIC technologies now transitioning toward magneto-resistive and phase-changing materials also offer digital memory capacities that have aided our technological progress. Combining these domains, we have moved toward big data analytics and the new era of artificial intelligence and

machine learning. This book provides a small selection of chapters covering aspects of ASIC development and the surrounding business model.
Getting Started with UVM: A Beginner's Guide is an introductory text for digital verification (and design) engineers who need to ramp up on the Universal Verification Methodology quickly. The book is filled with working examples and practical explanations that go beyond the User's Guide.
This book provides a hands-on, application-oriented guide to the language and methodology of both SystemVerilog Assertions and SystemVerilog Functional Coverage. Readers will benefit from the step-by-step approach to functional hardware verification using SystemVerilog Assertions and Functional Coverage, which will enable them to uncover hidden and hard to find bugs, point directly to the source of the bug, provide for a clean and easy way to model complex timing checks and objectively answer the question ‘have we functionally verified everything’. Written by a professional end-user of ASIC/SoC/CPU and FPGA design and Verification, this book explains each concept with easy to understand examples, simulation logs and applications derived from real projects. Readers will be empowered to tackle the modeling of complex checkers for functional verification, thereby drastically reducing their time to design and debug. This updated second edition addresses the latest functional set released in IEEE-1800 (2012) LRM, including numerous additional operators and features. Additionally, many of the Concurrent Assertions/Operators explanations are enhanced, with the addition of more examples and figures. · Covers in its entirety the latest IEEE-1800 2012 LRM syntax and semantics; · Covers both SystemVerilog Assertions and SystemVerilog Functional Coverage language and methodologies; · Provides practical examples of the what, how and why of Assertion Based Verification and Functional Coverage methodologies; · Explains each concept in a step-by-step fashion and applies it to a practical real life example; · Includes 6 practical LABs that enable readers to put in practice the concepts explained in the book.
A Practical Guide to Adopting the Universal Verification Methodology (UVM) Second EditionLulu.comThe Uvm PrimerA Step-By-Step Introduction to the Universal Verification Methodology
This book describes for readers a methodology for dynamic power estimation, using Transaction Level Modeling (TLM). The methodology exploits the existing tools for RTL simulation, design synthesis and SystemC prototyping to provide fast and accurate power estimation using Transaction Level Power Modeling (TLPM). Readers will benefit from this innovative way of evaluating power on a high level of abstraction, at an early stage of the product life cycle, decreasing the number of the expensive design iterations.
Since its introduction in 2011, the Universal Verification Methodology (UVM) has achieved its promise of becoming the dominant platform for semiconductor design verification. Advanced UVM delivers proven coding guidelines, convenient recipes for common tasks, and cutting-edge techniques to provide a framework within UVM. Once adopted by an organization, these strategies will create immediate benefits, and help verification teams develop scalable, high-performance environments and maximize their productivity. "Written by an experienced UVM practitioner, this book contains lots of great tips on using UVM effectively and example code that actually works!" John Aynsley, Doulos "In 'Advanced UVM', Mr. Hunter, based on his company's real world experiences, provides excellent resources, a well-tested reference verification environment, and advanced best practices on how to apply UVM. If you are ready to move beyond a UVM introduction, this should be the book you add to your library." George Taglieri, Director Verification Product Solutions, Synopsys, Inc.
The proceeding is a collection of research papers presented, at the 9th International Conference on Robotics, Vision, Signal Processing & Power Applications (ROVISP 2016), by researchers, scientists, engineers, academicians as well as industrial professionals from all around the globe to present their research results and development activities for oral or poster presentations. The topics of interest are as follows but are not limited to: • Robotics, Control, Mechatronics and Automation • Vision, Image, and Signal Processing • Artificial Intelligence and Computer Applications • Electronic Design and Applications • Telecommunication Systems and Applications • Power System and Industrial Applications • Engineering Education
This book features the manuscripts accepted for the Special Issue “Applications in Electronics Pervading Industry, Environment and Society—Sensing Systems and Pervasive Intelligence” of the MDPI journal Sensors. Most of the papers come from a selection of the best papers of the 2019 edition of the “Applications in Electronics Pervading Industry, Environment and Society” (APPLEPIES) Conference, which was held in November 2019. All these papers have been significantly enhanced with novel experimental results. The papers give an overview of the trends in research and development activities concerning the pervasive application of electronics in industry, the environment, and society. The focus of these papers is on cyber physical systems (CPS), with research proposals for new sensor acquisition and ADC (analog to digital converter) methods, high-speed communication systems, cybersecurity, big data management, and data processing including emerging machine learning techniques. Physical implementation aspects are discussed as well as the trade-off found between functional performance and hardware/system costs.
FPGA Simulation: A Complete Step-by-Step Guide shows FPGA design engineers how to avoid long lab debug sessions by simulating with SystemVerilog. The book helps engineers to have never simulated their designs before by bringing them through seven steps that can be added incrementally to a design flow. Engineers start with code coverage as the first step. Succeeding steps introduce test planning, assertions, and SystemVerilog simuation techniques. By the end of the process engineers who have never simulated before will know how to create complete self-checking test benches that generate their own stimulus, and demonstrate complete functional coverage. This book is a must for engineers who are facing DO-254 certification requirements on their next FPGA project.
The book presents selected papers from the Fifteenth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, in conjunction with the Twelfth International Conference on Frontiers of Information Technology, Applications and Tools, held on July 18–20, 2019 in Jilin, China. Featuring the latest research, it provides valuable information on problem solving and applications for engineers in computer science-related fields, and is a valuable reference resource for academics, industry practitioners and students.
The Universal Verification Methodology is an industry standard used by many companies for verifying ASIC devices. In this book, you will find step-by-step instructions, coding guidelines and debugging features of UVM explained clearly using examples. The book also covers the changes from UVM-1.1d to UVM 1.2 and provides details of the enhancements in the upcoming IEEE 1800.2 UVM standard: http: //www.accellera.org/community/uvm/faq The Table of Contents, Preface, Foreword from UVM committee members and detailed information on this book is available on www.uvmbook.com.

This book describes in detail all required technologies and methodologies needed to create a comprehensive, functional design verification strategy and environment to tackle the toughest job of guaranteeing first-pass working silicon. The author first outlines all of the verification sub-fields at a high level, with just enough depth to allow an engineer to grasp the field before delving into its detail. He then describes in detail industry standard technologies such as UVM (Universal Verification Methodology), SVA (SystemVerilog Assertions), SFC (SystemVerilog Functional Coverage), CDV (Coverage Driven Verification), Low Power Verification (Unified Power Format UPF), AMS (Analog Mixed Signal) verification, Virtual Platform TLM2.0/ESL (Electronic System Level) methodology, Static Formal Verification, Logic Equivalency Check (LEC), Hardware Acceleration, Hardware Emulation, Hardware/Software Co-verification, Power Performance Area (PPA) analysis on a virtual platform, Reuse Methodology from Algorithm/ESL to RTL, and other overall methodologies.

The UVM Primer uses simple, runnable code examples, accessible analogies, and an easy-to-read style to introduce you to the foundation of the Universal Verification Methodology. You will learn the basics of object-oriented programming with SystemVerilog and build upon that foundation to learn how to design testbenches using the UVM. Use the UVM Primer to brush up on your UVM knowledge before a job interview to be able to confidently answer questions such as "What is a uvm_agent?," "How do you use uvm_sequences?," and "When do you use the UVM's factory." The UVM Primer's downloadable code examples give you hands-on experience with real UVM code. Ray Salemi uses online videos (on www.uvmprimer.com) to walk through the code from each chapter and build your confidence. Read The UVM Primer today and start down the path to the UVM.

The Universal Verification Methodology is an industry standard used by many companies for verifying ASIC devices. It has now become an IEEE standard IEEE 1800.2. This book provides step-by-step instructions, coding guidelines and debugging features of UVM explained clearly using examples. It also contains porting instructions from UVM 1.2 to UVM 1800.2 along with detailed explanations of many new features in the latest release of UVM. The Table of Contents, Preface, and detailed information on this book is available on www.uvmbook.com.

This book describes the life cycle process of IP cores, from specification to production, including IP modeling, verification, optimization, and protection. Various trade-offs in the design process are discussed, including those associated with many of the most common memory cores, controller IPs and system-on-chip (SoC) buses. Readers will also benefit from the author’s practical coverage of new verification methodologies. such as bug localization, UVM, and scan-chain. A SoC case study is presented to compare traditional verification with the new verification methodologies. Discusses the entire life cycle process of IP cores, from specification to production, including IP modeling, verification, optimization, and protection; Introduce a deep introduction for Verilog for both implementation and verification point of view. Demonstrates how to use IP in applications such as memory controllers and SoC buses. Describes a new verification methodology called bug localization; Presents a novel scan-chain methodology for RTL debugging; Enables readers to employ UVM methodology in straightforward, practical terms.

What are the different types of verification approaches in SV? What is UVM VLSI? Universal Verification Methodology Tutorial Universal Verification Methodology Books Uvm Verification Interview Questions This book is an introductory text for digital verification (and design) engineers who need to ramp up on the Universal Verification Methodology quickly. The book is filled with working examples and practical explanations that go beyond the User's Guide.

mental improvements during the same period. What is clearly needed in verification techniques and technology is the equivalent of a synthesis productivity breakthrough. In the second edition of Writing Testbenches, Bergeron raises the verification level of abstraction by introducing coverage-driven constrained-random transaction-level self-checking testbenches all made possible through the introduction of hardware verification languages (HVLs), such as e from Verisity and OpenVera from Synopsys. The state-of-art methodologies described in Writing Test benches will contribute greatly to the much-needed equivalent of a synthesis breakthrough in verification productivity. I not only highly recommend this book, but also I think it should be required reading by anyone involved in design and verification of today's ASIC, SoCs and systems. Harry Foster Chief Architect Verplex Systems, Inc. xviii Writing Testbenches: Functional Verification of HDL Models PREFACE If you survey hardware design groups, you will learn that between 60% and 80% of their effort is now dedicated to verification.

Universal asynchronous receiver/ transmitter (UART) is a computer hardware device that is used to transform user data to parallel and serial forms. The UART protocol is typically integrated in an Integrated Circuit (IC) which is used for serial communication over a computer or a peripheral device. The UART protocol accepts the input data as bytes and transmits the single bits in a sequential way. Usually, there will one or more UART peripherals which are integrated in microcontrollers and related devices which can support synchronous operation. The UART communication method may be done in three different ways: simplex (in one direction only), full duplex (both devices can receive and send data simultaneously) and half duplex (both devices can take turns to receive and transmit data). The structure of the data frame contains a single start bit, followed by next five to nine bits, depends on the code set employed. If there is a parity bit, then it will be placed after all the data bits. The next one or two bits are always logic high to indicate the stop bit(s). This technical report will demonstrate a verification methodology to verify WISHBONE UART IP Core using the Universal Verification Methodology (UVM) process