

采用 UVM 方法学实现验证的可重用与自动化

徐金甫, 李森森

(解放军信息工程大学, 河南 郑州 450000)

摘 要: 在分析 UVM 的基础上, 重点研究了验证的可重用与自动化实现方法. 重用体现在验证组件的横向、纵向重用实现和验证场景的继承、组合重用实现. 自动化实现依赖平台构建时的 Field-Automation 机制、脚本控制的 RAL 模型生成以及平台运行时的仿真结果实时在线比对等. 最后, 以 PCI Express Controller 的验证工作为例, 说明了如何以自动化和重用性方法为指导, 快速构建结构化的 UVM 验证平台, 实施功能验证. 验证结果表明, 基于自动化与可重用的验证方法能够显著提升验证效率.

关键词: UVM; 自动化; 可重用; 验证组件; Field-Automation 机制

中图分类号: TP311

文献标识码: A

文章编号: 1000-7180(2014)11-0014-04

Adopting Universal Verification Methodology to Achieve Reusability and Automation Verification

XU Jin-fu, LI Sen-sen

(PLA Information Engineering University, Zhengzhou 450000, China)

Abstract: Based on the analysis on UVM, this paper focuses on the implementation method of reusability and automation. The reusability is reflected by the horizontal and vertical reuse of verification components and the inheritance and combined reuse of verification scenarios. Automation implementation relies on Field-Automation mechanism, script-controlled RAL model generation at platform constructing phase and the simulation results real-time compare online at platform running phase. Finally, the PCI Express Controller verification has been completed to illustrate the method of building the structured UVM verification platform rapidly guided by automate and reuse approach. Verification results show that the automation and reusability implementation method can significantly improve verification efficiency.

Key words: UVM; automation; reusable; verification component; field-automation mechanism

1 引言

随着可重用 IP(Intellectual Property)的广泛使用, SoC(System on Chip)设计朝着超大规模方向发展. 与此同时, 其验证难度不断增长. 验证完备性和验证效率的提升是验证的关键难点, 也是目前验证的主要研究方向^[1-2].

验证方法学的发展有效地提升了验证效率, 并使验证完备成为可能. 当前, 主流的验证方法学包括 UVM(Universal Verification Methodology)、VMM(Verification Methodology Manual)、OVM(Open Verification

Methodology)等. UVM 主体来源于 OVM, 同时继承了 VMM 中回调(callback)函数、寄存器抽象层(Register Abstract Layer, RAL)等技术. 所以, 与 OVM 相比, UVM 功能更加丰富强大. 与 VMM 相比, 不存在兼容问题, 并且架构层次清晰, 源码可读性好, 更加注重可重用与自动化的实现.

本文在研究 UVM 的基础上, 重点实现了验证平台的可重用与自动化, 并在此基础上构建 UVM 验证平台, 对 PCI Express Controller 的不同传输过程实施了快速验证.

2 UVM 验证平台

UVM 提供了一系列标准类,如 `uvm_driver`、`uvm_monitor`、`uvm_sequencer`、`uvm_agent`、`uvm_env`、`uvm_sequence`、`uvm_test` 等,方便继承和重载使用.为了便于重用,UVM 方法学中将 `uvm_sequencer`、`uvm_driver`、`uvm_monitor` 封装成 `uvm_agent` 类,作为最重要的可重用部件.使用扩展自 `uvm_object` 的配置类可以将 `uvm_agent` 配置成 active 或 passive 模式,passive 模式只实例化 monitor,不驱动总线,仅负责监测总线.这些可配置、可重用的通用验证组件(Universal Verification Component, UVC)的使用能够加快验证的实施^[3].

图 1 所示为一个典型的 UVM 验证平台,整个验证平台的构建以 UVC 为基础.在验证平台中例化了两个 agent 类,一个配置成 active 模式,负责驱动和监测总线;一个配置成 passive 模式,负责监控验证对象的响应.在 agent 类中,sequencer 负责调度 sequences 产生操作事务,并通过事务级传输模型(Transaction Level Model, TLM)传送给 driver,由 driver 负责将高级事务“翻译”成符合时序要求的 pin 级时序信号驱动到总线上.collector 则将 pin 级时序信号“翻译”成高级事务,通过 TLM 传送给 monitor.由 monitor 负责基本的协议检查和覆盖率分析,也可把高级事务传送给计分板或者系统覆盖率模型.平台中的组件可以通过 Factory 模式进行替换,实现验证组件的快速更新,具有很强的灵活性.

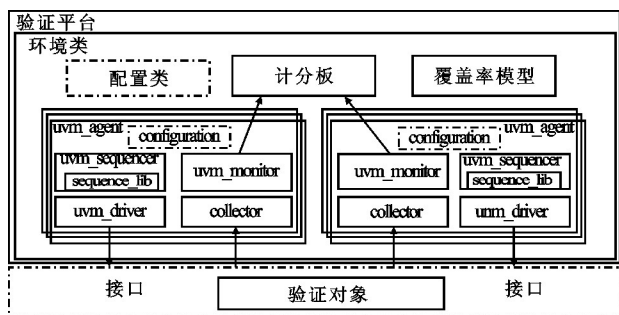


图 1 典型的 UVM 验证平台结构

3 可重用与自动化研究

3.1 平台可重用性实现研究

重用可以避免重复开发,有效减少验证时间开销.其主要内容包括如何重用已有的验证平台结构和验证测试用例.具体而言,就是最大限度地重用已

经开发完成的验证组件和验证序列.

3.1.1 验证组件的可重用性实现

验证平台部分组件可以实现重用.重用主要体现在两个方面:一是横向的可重用,二是纵向的可重用.横向可重用是指验证组件在不同的 IP 验证中的重用.横向重用的基础是不同的 IP 具有相同的总线接口.而在 SoC 高速发展的背景下,很多 IP 的设计都遵循某一接口规范,如 AMBA、Wishbone 或 Avalon 总线接口规范等.如图 2 所示,基于相同接口的 IP1 和 IP2 的验证,可以重用总线 agent1.需要变动的是更改和替换 agent1 类中 sequence_lib 的 sequence 队列.这种改动在基于 UVM 的验证平台中,使用 Factory 模式能够方便地得以实现.

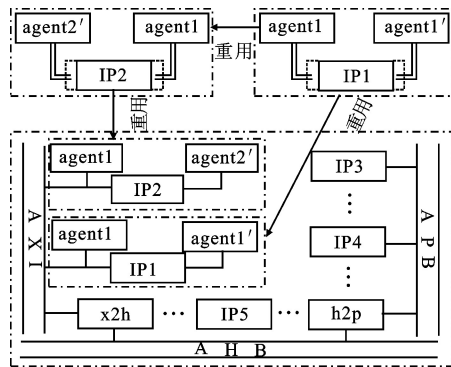


图 2 验证组件可重用示意图

纵向可重用是指验证组件从模块级验证重用到系统级.如图 2 所示,IP1 的模块级验证环境可以直接重用到系统级.在系统级验证环境中,先前由 agent1 驱动的激励信号改由系统的 CPU 核提供,因此 agent1 需要使用 `cfg` 类配置成 passive 模式,agent1 不需要任何改变,能够直接重用到系统级.

3.1.2 验证序列的可重用性实现

可重用性不仅仅体现在验证组件的可重用上,也表现在测试用例的可重用上.在 UVM 平台中,测试用例的重用通过序列的重用得以体现.主要有以下三种重用形式:一是继承重用,二是组合重用,三是模块级测试用例到系统级的重用.继承重用是指通过类的继承,重用先前类的属性和方法,继承已有的序列生成新的序列,如图 3 中虚线圆圈所示.主要是通过修改已有的约束条件或者添加新的控制属性,实现不同的测试类的生成.组合重用是指通过不同的排列组合规划已有的序列,生成新的序列,如图 3 中虚线方框所示.譬如,已经开发完成的读操作、写操作,可以组合成写后读或者读后写等新的操作.在模块级验证环境到系统级的重用中,如图 2 中针

对 agent1 开发的序列不需要任何改动就可以被重用到系统级。

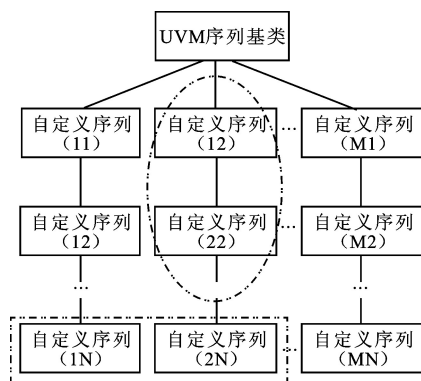


图3 验证序列可重用示意图

3.2 平台的自动化实现研究

平台的自动构建和运行,是验证效率提升的有力手段。本文重点研究 Field-Automation 机制、RAL 模型自动生成和仿真结果实时在线比对。

3.2.1 Field-Automation 机制

Field-Automation 机制使得常用的类方法能够自动生成,这种自动生成依赖于已经开发完成的、便于使用的一系列 Factory 机制的宏。具体来说,通过宏的使用自动实现了 print 函数、copy 函数、compare 函数、pack 函数、unpack 函数等^[4]。表1中给出了 Field-Automation 机制宏的使用方法。其中,A 表示基本的数据类型,如 int、real、enum、object、event、string 等;B 表示的是联合数组的索引类型,常使用 string 和 object 类型。

表1 宏的使用

位置	使用方法	含义
宏开始	<code>`uvm_object_utils_begin(class_name)</code>	-----
	<code>`uvm_field_A(arg,flag);</code>	基本类型
	<code>`uvm_field_array_A(arg,flag);</code>	动态数组
宏体	<code>`uvm_field_sarray_A(arg,flag);</code>	静态数组
	<code>`uvm_field_queue_A(arg,flag);</code>	队列
	<code>`uvm_field_aa_A_B(arg,flag);</code>	联合数组
宏结尾	<code>`uvm_object_utils_end</code>	-----

如果类中没有定义任何新的属性,可以使用更加简化的宏来实现 Field-Automation 自动化机制,如 ``uvm_object_utils()`、``uvm_component_utils()`、``uvm_sequencer_utils()` 等。

3.2.2 RAL 模型的自动生成

寄存器抽象层(Register Abstract Layer,RAL)是将寄存器和存储器的验证单独抽象出来,更加简

化验证流程。寄存器抽象层的作用体现在两方面:一是寄存器、存储器的验证,二是其他功能验证前将设计配置到特定的可用状态。在验证平台中,寄存器模型和寄存器测试例占据了代码量的很大比例,并且具有重复特性^[5]。RAL 模型的自动生成将极大减轻验证人员负担、减少代码错误。为此,将寄存器抽象模型的代码分为固定部分和可变部分,使用脚本语言开发出模板代码。用户只需以 RALF 文件格式提供相应的寄存器名、寄存器字段、寄存器位宽、寄存器地址、寄存器复位值、寄存器层次化路径等基本信息,模板代码依据这些信息,在脚本控制下就能自动生成相应的寄存器模型^[6]。

3.2.3 结果自动比对

模拟仿真结果比对方式通常有如下三种:波形查看、后处理方式、在线实时对比。波形查看方式效率低下,只在定向测试时使用。后处理方式要等待仿真结束之后进行,需要记录大量 log 数据,对存储空间的占用通常较大。在线实时对比能够在仿真过程中实时检查验证对象的响应,具有很大优势。将验证对象分为接口类和运算类两大类。接口类的自动比对机制如图4(a)所示,在不同的接口上安置监视器,负责收集数据并“翻译”成高级事务,送给计分板比对。运算类的自动比对机制则如图4(b)所示,模拟激励同时施加到验证对象和参考模型上,计分板负责实时比对两者响应是否一致,进而判断验证对象是否按预期工作。

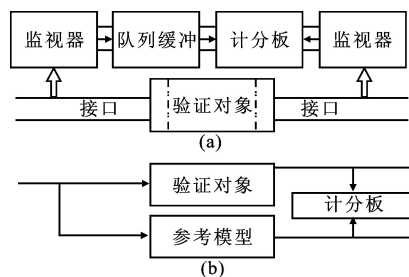


图4 自动比对机制

4 基于可重用与自动化的 PCIe 验证

本节以 PCI Express Controller 的快速验证为例,进一步说明文中所提的可重用与自动化方法,并对其验证效率进行评估。

4.1 PCIe 验证平台构建

应用可重用与自动化方法,针对 PCIe 控制器设计了如图5所示的 UVM 验证平台。图中带有底纹部分验证模块从已有的验证库中选取,不需要重

新设计开发. 本平台中设计了总线边带信号的驱动模型、时钟/复位控制模型、控制信号处理模型. 适配器的设计是为了统一不同的数据格式^[7]. 适配器 1 负责将 PCIe VIP 的数据包格式转化为总线 VIP 的数据格式, 计分板以总线 VIP 数据格式完成数据的自动比对过程. 适配器 2 负责将寄存器型的数据格式转化为总线 VIP 的数据格式, 由总线 VIP 负责驱动到验证对象上. 寄存器模型使用自动工具生成, 生成命令为 `ralgen -t pcie_top_block -uvm -b -l sv pcie. ralf`. 寄存器模型使用两种操作方式实施验证过程: 前门操作和后门操作^[8]. 前门操作是指通过驱动接口信号施加模拟激励. 后门操作是指通过层次化引用, 直接查看寄存器、存储器的值. 寄存器验证采用前后门交叉访问, 即前门 DBI 接口写, 后门读; 后门写, 前门 DBI 接口读.

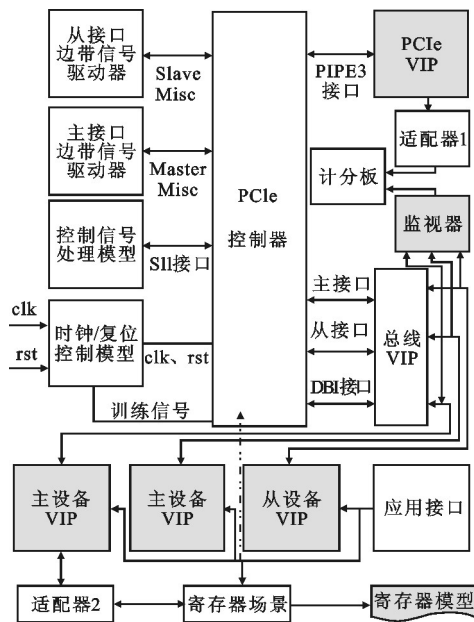


图 5 PCI Express Controller 验证结构图

4.2 测试用例设计

不同的测试用例可以简化为对 `sequence_lib` 中不同 `sequence` 的选取过程. 因此, 验证的实施过程就是 `sequence_lib` 逐渐完善的过程. 编写完成的 `sequence` 调用宏 `uvm_declare_p_sequencer()` 注册到相应 `sequencer` 的 `sequence_lib` 中去. PCIe 验证过程中设计了如表 2 所示的一系列测试用例. 这些测试用例针对 PCIe 的存储器读写功能、IO 读写功能、配置空间访问功能、DBI 接口访问功能、内部 DMA 猝发传输功能、MSI-X 中断功能、低功耗、链路训练等进行了完备验证. 其中首先设计了基本的测试用例 `pcie_base_test`, 作为其它测试用例的公用父类,

此类中定义各个测试用例中的共有部分, 如平台层次打印、仿真结果报告等.

表 2 PCIe 测试用例列表

测试用例	描述
<code>pcie_base_test</code>	公用父类
<code>pcie_remote_mem_test</code>	测试远端存储器读写数据包的场景
<code>pcie_local_mem_test</code>	测试本地存储器读写数据包的场景
<code>pcie_remote_io_test</code>	测试远端 IO 读写数据包的场景
<code>pcie_local_io_test</code>	测试本地 IO 读写数据包的场景
<code>pcie_remote_cfg_test</code>	测试远端配置读写数据包的场景
<code>pcie_dbi_test</code>	测试本地 DBI 接口读写访问的场景
<code>pcie_msi_test</code>	测试 PCIe 消息中断的场景
<code>pcie_iatu_test</code>	测试内部地址转换单元的场景
<code>pcie_idma_test</code>	测试内置 DMA 猝发传输的场景
<code>pcie_aspm_test</code>	测试 PCIe 电源管理的场景
<code>pcie_ltssm_test</code>	测试 PCIe 链路训练的场景

4.3 效率分析

PCIe 验证平台以重用与自动化方法为原则. 整个验证平台的代码量在数十万行之多, 由于重用 PCIe VIP 完成协议数据封装、链路自动训练, 重用总线 VIP 和主/从设备 VIP 完成总线抽象事务到信号级事务转换, 使得验证过程只关注顶层测试用例的编写, 自己开发的代码仅在数万行之内, 80% 以上的代码通过重用得以实现. 同时, 分析 VMM 代码文件, 常用的属性方法代码量占据总代码量的 60% 以上, 而在 UVM 平台中, 使用 Field-Automation 机制仅需数行代码就可以完成. RAL 模型自动生成, 将复杂的寄存器模型编写简化为简单的 RALF 格式的寄存器描述, 然后借助脚本工具快速完成转换生成. 更少的新生代码以及自动化的代码生成, 保证了更少的代码错误, 使得花费在平台正确性调试上的验证时间也相应减少. 结果自动在线比对能够大幅减少结果校验的时间. 可见, 可重用与自动化在平台构建、平台调试以及平台运行中能够, 极大提高验证效率.

5 结束语

UVM 方法学吸收主流方法学的优点, 使得验证语言统一于 SystemVerilog, 验证方法学统一于 UVM, 并且能够最大限度重用, 方便集成先进的验证技术, 使得验证难度被有效分解. 借助开发完成的脚本工具能够自动化地生成验证平台的诸多组件, 加快了验证的实施. 本文基于可重用与自动化方法构建 UVM 平台, 快速完成了 PCIe 的验证工作. 结果表明, 验证效率得到了大幅提升.

(下转第 22 页)

由于移动站和固定站的 RSSI 模式完全不同,本文实现的认证方法的应用环境是固定站模式.在该模式下,在离线 RSSI 采集阶段可获得稳定的 Radio Map.但由于 RSSI 易受外界环境变化的影响,所以要定期更新 Radio Map 以保证高的认证准确度.

4 结束语

本文设计和实现了一种新颖的无感知 WLAN 认证方案,该方案利用无线信号的 RSSI 值很难被伪造的特性,采用改进的 K-means 聚类算法降低环境因素对 RSSI 带来的波动影响,构造了基于 RSSI 的 Radio Map,然后通过 RSSI 位置指纹区配对用户访问 WLAN 进行审核认证.测试结果表明,该方法具有抗干扰能力强,认证成功率高优点.

参考文献:

- [1] IEEE Standards Board 802 part 11. Wireless LAN medium access control and physical layer specifications [S]. IEEE Standard 802.11, 1999.
- [2] IEEE Standard 802.11X: Port-based network access control for local and metropolitan area networks [S]. IEEE Standard 802.11X, 2004.
- [3] 中国移动 WLAN 用户接入流程技术规范 [S]. 北京:中国移动通信有限公司, 2008.
- [4] Koski L, Perala T, Piche R. Indoor positioning using WLAN coverage area estimates[C]// International Conference on Indoor Positioning and Indoor Navigation, Switzerland, 2010.
- [5] Kamol, Kaemarungsi, Krishnamurthy. Properties of

indoor received signal strength for WLAN location fingerprinting ubiquitous [C] // First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous'04). USA: Boston, 2004.

- [6] 文柳. WLAN 无线接入网络规划 [D]. 北京:北京邮电大学, 2010.
- [7] Azin Arya, Philippe Godlewski. An analysis of radio fingerprints behavior in the context of RSSI-based location fingerprinting systems [C] // Proceedings of the 5th ACM international workshop on mobility management and wireless access. Canada: Toronto, ACM, 2007:1-8.
- [8] Kamol Kaemarungsi, Prashant Krishnamurthy. Analysis of WLAN's received signal strength indication for indoor location fingerprinting [J]. Pervasive and Mobile Computing, 2012, 8(2):292-316.
- [9] 冯柳平, 刘祥南, 刘明业. IEEE 802.11 访问控制与 MAC 地址欺骗 [J]. 微电子学与计算机, 2006, 23(1): 25-27.
- [10] 周爱武, 于亚飞. K-Means 聚类算法的研究 [J]. 计算机技术与发展, 2011(2):48-61.
- [11] 许虎寅, 王治和. 一种改进的基于密度的聚类算法 [J]. 微电子学与计算机, 2012, 29(2):44-47.

作者简介:

马飞 男, (1976-), 博士, 副教授. 研究方向为网络安全, 云计算、社交网络分析、隐私保护.

李娟 女, (1975-), 硕士, 副教授. 研究方向为网络安全、云计算.

(上接第 17 页)

参考文献:

- [1] Wang Hongwei, Zhan Yanglin, Liang Liping. A coverage-driven auto-verification platform [J]. Electronic test, 2013(5):82-84.
- [2] Zhong Wenfeng. System verilog and function verification [M]. Beijing: China Machine Press, 2010.
- [3] 田劲, 王小力. 基于 UVM 验证方法学的 AES 模块级验证 [J]. 微电子学与计算机, 2012, 29(8):86-90.
- [4] 谢峥, 王腾, 雍姗姗, 等. 一种基于 UVM 面向 RISC CPU 的可重用功能验证平台 [J]. 北京大学学报:自然科学版, 2014, 50(2):221-227.
- [5] Namdo Kim, YoungNam Yun, YoungRae Cho, et al. How to automate millions lines of top-level UVM test-bench and handle huge register classes [J]. IEEE, 2012:405-407.
- [6] Zheng Xie, Wang Xin'an, Lian Zhibin, et al. A novel intelligent verification platform based on a structured

analysis model [J]. Science China Information Sciences, 2013, 56(6):1-14.

- [7] Wang Shizhong, Tian Ze, Wu Xiaocheng, et al. Function verification of time synchronization circuit based on UVM [EB/OL]. (2013-04-18). [2014-01-10]. <http://www.cnki.net/kcms/detail/61.1450.TP.20130418.1701.005.html>.
- [8] Cappelli F, Lavagno A, Canegallo R, et al. A reconfigurable processor architecture and software development environment for embedded systems [C] // IPDPS'03. France, Nice, 2003.

作者简介:

徐金甫 男, (1965-), 博士, 副教授. 研究方向为专用集成电路设计与验证.

李森森 男, (1988-), 硕士. 研究方向为专用集成电路设计与验证.