

# 一种基于 UVM 加快功能验证收敛的方法

徐文进,田 泽,王世中,王宣明

(中航工业西安航空计算技术研究所 集成电路与微系统设计航空科技重点实验室,  
陕西 西安 710068)

**摘 要:**功能验证是 IC 设计最重要的步骤之一。随着设计复杂度的增加,定位缺陷成本也日益增加,许多设计验证团队将相当一部分精力放在高效验证过程开发上。文中以航电 CNI 系统主机接口模块功能验证为例,采用基于统一覆盖率验证管理技术、UVM 验证技术、接口时序断言监控技术,分别从验证计划制定、平台搭建、验证管理与验证结果分析等方面,介绍一种新型的基于统一覆盖率加速功能验证收敛的闭环验证方法。结合 UVM 验证方法学和断言技术,详细论述构建一种高效率可重用、快速收敛的验证平台的过程。

**关键词:**UVM; VM; 验证计划; 验证平台; 覆盖率

**中图分类号:**TP39

**文献标识码:**A

**文章编号:**1673-629X(2016)06-0111-04

doi:10.3969/j.issn.1673-629X.2016.06.024

## A Method of Speeding up Convergence Functional Verification Based on UVM

XU Wen-jin, TIAN Ze, WANG Shi-zhong, WANG Xuan-ming

(Key Laboratory of Aeronautics Science and Technology of Integrate Circuit and Micro-system Design,  
AVIC Computing Technique Research Institute, Xi'an 710068, China)

**Abstract:**Functional verification is one of the most important steps in the IC design. With the increase of design complexity, the cost of locating defects is also raised. Many designers will be a great deal of efforts on the development of efficient verification. In this paper, a new type of closed-loop verification method based on uniform coverage is introduced, taking functional verification of the host interface module of CNI system as example, using authentication management technology based on uniform coverage, UVM verification technology, and monitoring interface timing assertions respectively from the verification plan, platform building, management and authentication results analysis, etc. The process of constructing an efficient, reusable and fast convergent verification platform is discussed in detail by combining UVM verification methodology and assertion technology.

**Key words:**UVM; VM; test plan; test bench; coverage

## 0 引 言

过去 10 年,设计方法和技术变得越来越成熟和稳定,而功能验证技术的发展已经超越人们的认知水平,验证方法学和技术不断的演化,新的流程和工具不断被发明出来。在这种快速变化的大背景下,设计的稳定增长将更多更复杂的 IP 集成到越来越大的 SoC 复杂系统当中<sup>[1]</sup>,如何采取更好的验证流程和验证方法加快复杂设计功能验证的收敛成为难题。

文中以某航电系统控制接口模块为例,采用基于

统一覆盖率验证管理技术、UVM 验证技术、接口时序断言监控技术,分别从验证计划制定、平台搭建、验证管理与验证结果分析等方面,介绍一种基于统一覆盖率加快功能验证收敛的闭环验证方法。

## 1 主机接口模块

应用于航空电子的通信、导航、识别(CNI)系统中的某产品主机接口模块主要实现了外部处理器 MPC8270 对模块内部资源的访问控制,将处理器端的

收稿日期:2015-08-31

修回日期:2015-12-09

网络出版时间:2016-05-25

基金项目:中国航空工业集团公司创新基金(2010BD63111)

作者简介:徐文进(1983-),男,工程师,研究方向为集成电路设计与验证;田 泽,博士,研究员,中国航空工业集团首席技术专家,研究方向为 SoC 设计、嵌入式系统设计、VLSI 设计等。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20160525.1700.010.html>



```
clocking Cb_host_if @ (posedge HSTCLK);
default input #setup_time output #hold_time;
input HST_RST_N;
output HSTREGCSn HSTWRn;
inout HSTPSDVAL;
...
```

在多接口的复杂设计中,团队成员完成了设计集成后,模块间接口互操作时,难免出现非期望的结果。如果根据仿真最终结果判断出现异常情况,而逐个排查必然要花费大量的时间,虽然并发断言很难添加到 UVM 验证环境当中,但依然可以采用基于 Systemverilog 的 SVA 并发断言技术,精确描述模块接口处信号的时序关系,实时监测和诊断模块运行时是否与设计的接口时序关系一致,在发现错误时尽可能减少调试时间。

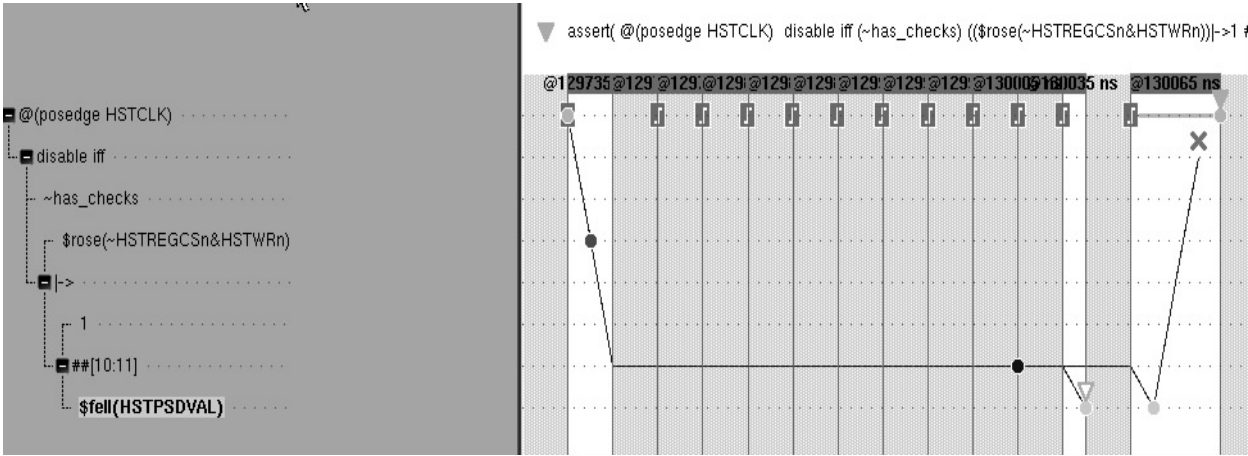


图4 调试断言的 ATV 窗口

(2) virtual interface,即虚接口,是连接 DUT 和验证环境接口的例化,验证平台通过驱动虚接口信号翻转,把激励施加到 DUT 上。

(3) driver,相当于总线功能模型(BFM),有 pull 和 push 两种模式。推荐采用 pull 模式,通过事务级 TLM 接口的方法发起新事务请求,并将获得的事务转换为接口信号的翻转,代码如下所示。

```
class host_driver extends uvm_driver #(host_trans) ;
forever begin
seq_item_port.get_next_item(req)
case (trans.kind)
REG_WR:reg_write(req) ;
REG_RD:reg_read(req) ;
...
```

(4) uvm\_transaction,是验证组件 sequencer 和 driver 之间通信时的事务载体,如该平台 host\_agent 中,host\_trans 就是事务载体,其包含了由 sequence 发出的读写类型、地址和数据,其部分代码如下。

```
class host_trans extends uvm_sequence_item;
randtrans_t kind ;
```

图4 为主机接口仿真运行到  $t = 1\,220\text{ ns}$  时发生断言失败的 ATV (Assertion Thread Viewer) 窗口,该窗口可方便调试 SystemVerilog 断言。

```
always @ (posedge HSTCLK)
begin
assertHSTPSDVALnUnkown_a: assert property (disable iff (!
has_checks)
($rose(~HSTREGCSn&HSTWRn) |->##[10:11] $fell
(HSTPSDVAL)) );
assertHSTPSDVALnUnkown_c: cover property (disable iff (!
has_checks)
($rose(~HSTREGCSn&HSTWRn) |-> ##[10:11] $fell
(HSTPSDVAL)) );
...
```

```
randbit [15:0] addr ;
randbit [31:0] data ;
...
```

(5) Sequencer,控制 sequences 的产生,并把 sequence 激励传递到 driver。

(6) Sequence\_lib,即包含了很多 sequence 的集合,sequence 是事务级激励。

(7) Monitor,用来监控接口事务,并将监控的数据广播到覆盖率收集组件。

(8) Coverage,主要完成功能覆盖率统计,功能覆盖率的实现不是自动的过程,需要对功能规范进行详细分析,使用 SystemVerilog 覆盖率语法构建覆盖率模型结合仿真运行的一种统计过程。如下代码是用覆盖组进行读写信息收集,结合地址信息,可以判断整个验证过程对所有分配地址空间是否进行覆盖。

```
covergroup host_cov;
OPCODE: coverpoint analysis_txn.we {
bins write = {0} ;
bins read = {1} ;
}
```

endgroup

2.3 验证管理与验证结果分析

该模块验证采用 Mentor 的 VM (Verification Management) 技术来管理整个验证过程,通过验证计划的输入、平台构建和运行,验证计划和回归测试后,统计

功能覆盖率、代码覆盖率、断言覆盖率的映射结果,分析优化验证方案。通过整个过程不断迭代,加快功能验证的收敛,整个验证过程具有可控性。

回归测试 1 次完成后,验证计划与覆盖率数据统计映射关系如图 5 所示。

Testplan section	Linked Items	Covered Items	Coverage	% of Goal	Type	Description	Weight	Goal	Eng
0 testplan	2676	1327	63.92%	63.92%	Testplan		1	100%	--
1 All_coverage	(none)	(none)	(none)	(none)	Testplan	给定设计单元的所有覆盖率,包括代码,功能,断言,自定义覆盖矩阵	1	100%	wsz
▶ 2 All_code_coverage	2635	1295	67.31%	67.31%	Testplan	代码覆盖率	10	100%	wsz
▶ 3 Assertions	5	4	80.00%	80.00%	Testplan	主机接口时序断言	1	100%	wsz
▼ 4 test	36	28	77.77%	77.77%	Testplan	策划的验证项	1	100%	wsz
▼ 4.1 register_reset	1	1	100.00%	100.00%	Testplan	寄存器复位功能验证	1	100%	wsz
host_reg_reset_test	--	--	100.00%	--	Test	--	--	--	--
▼ 4.2 register_rdwrr	1	1	100.00%	100.00%	Testplan	寄存器读写功能验证	1	100%	wsz
host_reg_rdwrr_test	--	--	100.00%	--	Test	--	--	--	--
▼ 4.3 register_endian	1	0	0.00%	0.00%	Testplan	大小端访问寄存器功能验证	1	100%	wsz
big_or_little_endian_access_reg_test	--	--	0.00%	--	Test	--	--	--	--
▼ 4.4 single_beat_access	1	1	100.00%	100.00%	Testplan	缓冲区测试	1	100%	wsz
host_buf_rdwrr_test	--	--	100.00%	--	Test	--	--	--	--
▼ 4.5 burst_access	1	1	100.00%	100.00%	Testplan	以burst方式访问缓冲区	1	100%	wsz

图 5 回归测试后覆盖率的映射关系

在测试项中,深色 0% 的验证项是没有执行过的,验证管理人员可以督促验证人员尽快进行验证项的开发。所有验证项运行后,显示的所有代码覆盖率、功能覆盖率统计如果达不到要求,需要验证人员对验证结果进行分析优化,删除对覆盖率没有帮助的冗余验证项,根据未覆盖到的场景修改开发更多的验证项,并可能与设计人员沟通,删除可能存在的冗余逻辑,从而使整个验证过程可控和快速收敛。当覆盖率达到 100% 或预期覆盖率时,验证过程结束。

3 结束语

在验证主机接口模块过程中,在任意时刻都可以很容易采用图形化界面或 HTML 报告的形式,回答项目管理者的问题,如“验证进程到达哪一步了”“何时验证可以收敛”。闭环的验证流程增加了验证项目的可见性,使验证工程师和项目管理人员交流起来方便容易,进一步提高了验证效率,降低了验证工作的时间<sup>[1]</sup>。

参考文献:

[1] Bhattacharya B, Decker J, Hall G. Advanced Verification Topics[M]. [s. l.]: Cadence Design Systems, Inc., 2012.  
[2] May D. Process management: are you driving in the dark with

faulty headlights[M]. [s. l.]: [s. n.], 2013.  
[3] Idrissi R E. ST-Ericsson speeds time to functional verification closure with the questa verification platform[M]. [s. l.]: ST-Ericsson, 2012.  
[4] Lam W K. Hardware design verification: simulation and formal method-based approaches[M]. [s. l.]: [s. n.], 2005.  
[5] Processor local bus functional model toolkit[M]. USA: IBM, 2003.  
[6] 韩霞, 杨洪斌, 吴悦. 面向 SoC 的事务级验证研究[J]. 计算机技术与发展, 2007, 17(3): 33-36.  
[7] 华为技术有限公司. 一个成功的百万门级芯片验证平台[J]. 中国集成电路, 2005(6): 36-43.  
[8] 杨海波, 田泽, 蔡叶芳, 等. FC IP 软核的仿真与验证[J]. 计算机技术与发展, 2009, 19(9): 168-172.  
[9] 李哲, 田泽, 张荣华. AFDX 网络交换机芯片关键性能验证方法研究[C]//第十六届计算机工程与工艺年会暨第二届微处理器技术论坛文集. 出版地不详: 出版者不详, 2012: 483-486.  
[10] 田靖, 田泽. AFDX-ES SoC 虚拟仿真平台的构建与应用[J]. 计算机技术与发展, 2010, 20(8): 192-194.  
[11] 郭蒙, 田泽, 蔡叶芳, 等. 1553B 总线接口 SoC 验证平台的实现[J]. 航空计算技术, 2008, 38(6): 99-101.  
[12] 吴晓成, 田泽, 郭蒙, 等. AFDX 交换芯片虚拟验证关键技术研究[J]. 计算机技术与发展, 2013, 23(8): 177-180.