

# 基于 UVM 的验证平台设计研究

王国军, 景为平

(南通大学 江苏省专用集成电路设计重点实验室, 江苏 南通 226019)

**摘 要:** 提出了一种采用 UVM(Universal Verification Methodology)验证方法学设计出的适合超高频 RFID 芯片解码系统的验证平台. 该验证平台相比于传统验证平台具有重用性强、层次化合理、效率高、验证自动化等优点, 且该验证平台以覆盖率驱动, 显著提高了验证的效率和质量. 现该验证平台已成功应用于一款超高频 RFID 标签芯片解码系统的设计验证, 得到了期望的覆盖率, 满足设计要求.

**关键词:** UVM; RFID; 验证平台; 覆盖率; DUT

**中图分类号:** TN47

**文献标识码:** A

**文章编号:** 1000-7180(2016)07-0164-05

## The Design and Research of the Verification Platform Based on UVM

WANG Guo-jun JING Wei-ping

(Jiangsu Key Laboratory of ASIC Design, Nantong University, Nantong 226019, China)

**Abstract:** This paper presents a new verification platform. Suite for decoder system of ultrahigh frequency RFID, the new platform is based on the method of UVM (Universal Verification Methodology). Compared to traditional verification platform, the new verification platform has many advantages, such as stronger reuse and extensibility, more reasonable hierarchical structure, higher efficiency and verification automation. And the platform is coverage-driven. Significantly, it enhances the efficiency and quality of verification. Now the verification platform has been successfully applied to the design and verification of decoder system in an ultrahigh frequency RFID tag chip. The verification report shows that coverage is expected and meet the requirement.

**Key words:** UVM; RFID; verification platform; coverage; DUT

### 1 引言

随着集成电路技术的不断发展, 芯片的规模和复杂度在快速提升, 这对验证提出了更高的要求. 芯片的一次流片成功率很大程度上取决于验证的充分性<sup>[1]</sup>. 所以, 验证已经成为芯片设计的发展瓶颈.

传统的功能验证方法, 是针对被测设计 (Design Under Test, DUT) 的信号接口, 手工编写测试向量, 并通过检查 DUT 的响应来判断设计功能的正确性. 这种方法抽象级别低, 对于复杂度高的设计来说, 无法达到理想的边界条件, 且难以准确定位错

误. 更严重的是其可重用性极低, 浪费了大量的开发时间. 所以, 如何搭建出一个质量好效率高的验证平台, 已经成为验证方法学发展的一个巨大挑战.

Cadence、Synopsys 和 Mentor 公司联合推出的 UVM 验证方法学, 是基于 System Verilog 验证语言建立的一套新一代的验证方法学, 在许多国外 IC 设计公司得到了广泛的应用. 随着国内 IC 设计的发展, UVM 也越来越得到验证工程师的重视.

### 2 UVM 验证方法学介绍

UVM 是一套高效、开源的验证方法学<sup>[2]</sup>. 它继

**收稿日期:** 2015-09-28; **修回日期:** 2015-11-13

**基金项目:** 江苏省高校自然科学研究重大项目 (13KJA510005); 江苏省物联网和新一代信息技术研发及产业化项目; 江苏省科技支撑计划-工业部分 (BE2013008-3)

承了 System Verilog 验证语言面向对象的思想,实现了验证的重用性和清晰的层次结构. 由于其开源的特质,使得它相比于其他的验证方法学,具备更好的兼容性. 目前,主流的 EDA 工具均支持 UVM 验证方法学.

UVM 平台的架构主要是由通用的验证组件(Universal Verification Component, UVC)构成的. 而每个 UVC 都已经是封装好、功能完善且可配置的验证环境<sup>[3]</sup>. UVC 可以分为接口 UVC 和模块 UVC. 接口 UVC 是对 DUT 的接口按照规定的协议建模,主要包括事务数据、序列器、序列发生器、驱动器、监视器、代理、环境等. 而模块 UVC 主要是对 DUT 的功能建立模型,包括参考模型、记分板等.

在验证工程项目搭建验证平台时,通过按照需求,配置功能齐全的 UVC,可以快速有效地搭建一个验证平台. 而在不同的工程项目中,当接口协议相同时,就可以调用相应的接口 UVC,当 DUT 的主要功能相同时,也可以调用相应的模块 UVC. 这种不同工程复用 UVC 的方法可以缩短开发周期,加快验证进度.

3 芯片验证平台设计

基于射频识别技术(RFID)的标签芯片,是一种非接触式标签芯片,在现代的工业、生活当中已经得到了极大的应用. 本次研究针对一款高频 RFID 标签,设计出了一个高效的验证平台,有效的节约了芯片验证资源,降低了芯片开发成本.

3.1 DUT 说明

该验证平台的 DUT 为基于 EPC Class-1 Generation-2 协议的超高频芯片的解码系统,该协议规定的通信频率为 860~960 MHz<sup>[4]</sup>. RFID 标签芯片

包含模拟电路、存储器、数字电路三个部分,如图 1 所示. 模拟电路主要负责射频信号的调制、解调以及提供芯片工作的电源和时钟. 存储器负责存储芯片的识别号(TID)及其他芯片信息. 数字电路部分实现标签芯片的编码解码、逻辑控制以及与存储器交互数据. 该验证平台主要对该芯片的解码系统部分进行仿真验证.

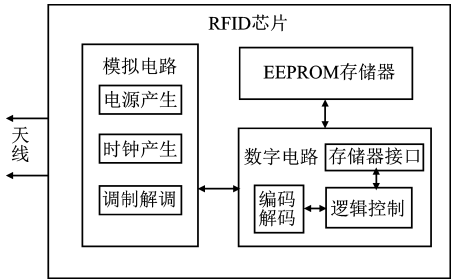


图 1 RFID 标签芯片结构图

支持 EPC 协议的阅读器和标签芯片的通信采用命令帧格式传输,由阅读器发往标签芯片的命令帧分为帧头(preamble)或者帧同步(frame-sync)和数据. 其中, preamble 和 frame-sync 的格式如图 2 所示. 从图中可以看出, preamble 第一部分是 delimiter,其时间长度为标准的  $12.5\text{ }\mu\text{s}$ ,误差在 5%之内,用以芯片校准时钟使用,之后是一个数据 0,然后是一个 RTcal 用以系统解码参照使用,最后是 TRcal,规定了芯片返回数据的速率. frame-sync 比 preamble 少了 TRcal 项. 阅读器和标签芯片的通信,第一条通信命令必须使用带有 TRcal 的 preamble 作为帧开始标志,用以通知标签芯片的返回数据速率,接下来的一组命令就可以使用以 frame-sync 为帧开始标志的命令帧了.

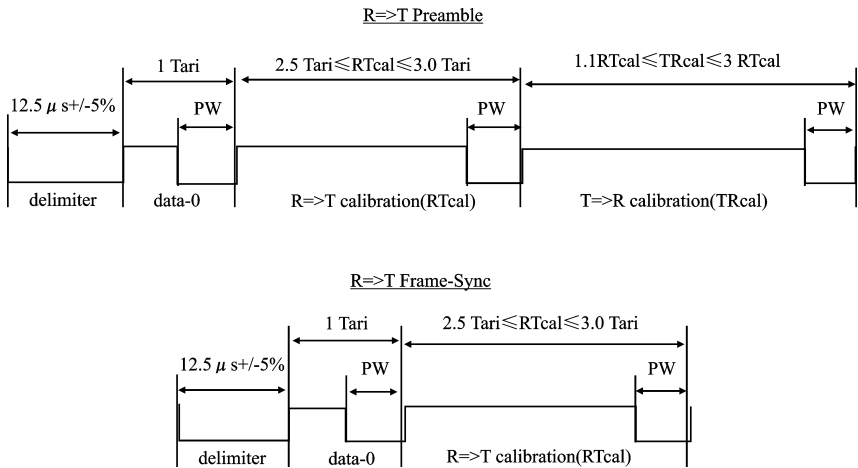


图 2 preamble 和 frame-sync 格式

而数据部分的编码使用脉冲宽度编码 (pulse-interval encoding, PIE). 其编码方式如图 3 所示. 图中上面部分为数据 0 的编码方式, 下面为数据 1 的编码方式, 根据图中给出的编码长度的关系比较, 可以从命令帧中解码出数据信息.

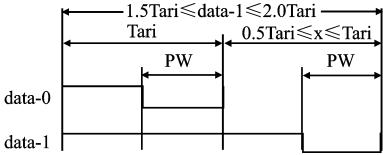


图 3 PIE 编码方式

3.2 验证平台的主要模块

针对该 DUT, 根据 UVM 验证方法学的思想所设计的验证平台, 如图 4 所示. 该验证平台通过最顶层 Top 将由 UVM 搭建的平台和 Interface 以及 DUT 连接起来. Interface 是 System Verilog 方便模块间通信的接口建模, 可以看作是一组智能的连线. 接口包含了连接、同步、甚至两个或者多个块之间的通信功能, 它用来连接 DUT 和验证平台<sup>[5]</sup>.

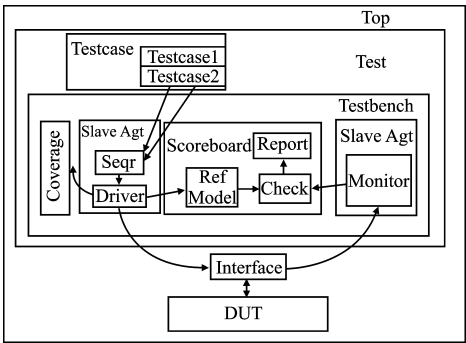


图 4 验证平台结构图

验证平台中的 Test 包含 Testcase 和 Testbench 两个部分. Testcase 包含了大量的测试用例, 用来产生测试场景. 每一个 testcase 是一个测试场景, 当一个 testcase 被送到 Testbench 下的序列发生器 (Seqr) 后, 序列发生器产生事务序列, 并送至 Driver 驱动器. Driver 也可以称为总线功能模型 (BFM), 即按照协议规定的物理接口协议向 DUT 发送有效的物理激励. 同时, Driver 也将激励送给参考模型, 参考模型根据激励得出期望结果, 并送给 Scoreboard (记分板). 另外 Monitor (监视器) 对 DUT 的响应进行监测, 并收集数据, 也会将收集到的数据送到 Scoreboard. Scoreboard 通过 Check 检验器对两组数据进行比对, 将比对结果记录在 Report 中, 并形成日志文件, 以方便查看. Coverage 的功能为覆盖率收集, 它是在 Driver 将激励送出后,

通过 callback (回调) 机制进行覆盖率统计.

4 验证过程和结果

4.1 测试激励产生

在验证开始之前, 必须按照测试计划, 编写全面的测试用例. 这里测试用例的产生, 使用约束随机产生的方式. 通过编写随机约束, 仿真器根据约束要求, 产生满足约束的测试激励. 在图 1 和图 2 中, 可以看到帧的编码长度是一个变化的范围, 所以对于不同的情况, 都要测试, 以验证 DUT 是否能够满足协议规定的所有情况. 通过随机测试, 验证平台可以命中一些难以预期的边界条件, 提高验证覆盖率. 以下程序是一个简单针对物理规范编写的随机约束:

```
class my_transaction extends uvm_sequence_item;
    rand bit[3:0] r_del; // 0-10
    rand bit[2:0] r_tari; // 0-3
    rand bit[4:0] r_pw; // 0-26
    rand bit[2:0] r_RTcal; // 0-5
    rand bit[4:0] r_TRcal; // 0-19
    rand bit[2:0] r_x; // 0-5
    rand bit f_TRcal; // 命令帧是否含有 TRcal
    .....

    constraint cons
    {
        r_del <= 10;
        r_tari <= 3;
        r_pw <= 26;
        r_RTcal <= 5;
        r_TRcal <= 19;
        r_x <= 5;
        f_TRcal dist {0:=1, 1:=4};
        .....
    }

    function new(string name = "my_transaction");
        super.new(name);
    endfunction

    `uvm_object_utils_begin(my_transaction)
    .....
    `uvm_field_int(r_del, UVM_ALL_ON | UVM_NOPACK)
    .....
    `uvm_object_utils_end
endclass
```

程序中, 对定义的各个变量进行了约束, 使得约束后产生的数据经过简单运算后, 就可以使得命令帧中各个部分的物理关系满足协议要求. 如 r\_del 变量, 经过约束后产生 0~10 的随机数, 然后 r\_del 会被送至驱动器, 驱动器中将 r\_del 经过简单的运

算,使得 delimiter 的时间长度为 11. 875~13. 125  $\mu\text{s}$  之间,满足协议规定的 12. 5  $\mu\text{s}$  长度 5% 的误差. 其他参数同样,保证了在随机测试下,依然能够得到满足协议规范的测试激励.

4. 2 事务级建模

所谓的事务级建模指的是 Driver 驱动器在获取序列发生器的测试激励后,根据协议规定,将其建模为 DUT 需要的物理激励. 这个物理激励可以称为一个事务. 此处驱动器需要将测试激励建模为包含如图 2 和图 3 所示的命令帧. 驱动器将一个事务通过 Interface 送至 DUT,开始对 DUT 的验证.

4. 3 自动化验证

该验证平台验证的过程为自动化流程. 其中验证平台的外部控制通过脚本实现,如开始验证、结束验证、是否产生波形文件、形成严重报告等动作. 另外,验证平台同时具备自动检查数据的功能,这个省

去了人工检查波形的工作,大大提高了验证效率. Driver 驱动器在将事务送至 DUT 的同时,也会送一份同样的事物至 Ref\_Model(参考模型). Ref\_Model 含有一个使用 System Verilog 编写的参考模型,参考模型经过数据分析,得出一组期望的结果数据,将期望数据送至 Scoreboard 记分板. 同时,记分板也会收到从 Monitor 监视器送来的数据. 在收到两组数据后,记分板通过 Check 检验器对两组数据进行比对. 并将比对结果写入 log 文件,方便形成验证报告.

在验证过程中,会出现错误的情况,这个时候,通过日志文件,查看产生错误的 testcase,将此 testcase 再次送入序列发生器,验证平台会根据上文中所讲到的动作过程再次仿真验证,并产生波形文件,方便调试. 一个正确的验证过程波形记录如图 5 所示.

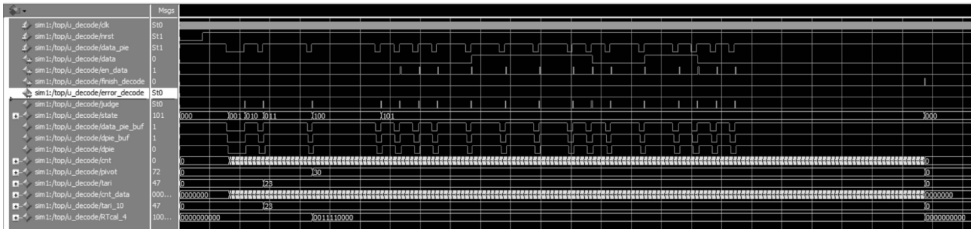


图 5 仿真波形图

图中 clk 信号为解码系统的工作时钟,在 nrst 系统复位后,开始准备解码工作. 当 data\_pie 信号,收到 Driver 驱动器的物理信号驱动时,DUT 检测到信号后开始解码. data\_pie 的第一个 pulse(凹槽)为 delimiter(见图 1),第二个 pulse 为数据 0,然后紧接着两个 pulse 分别为 RTcal 和 TRcal. 此时 preamble 部分解码完毕,后面是数据解码. 当数据区域的第一个 pulse 结束时,en\_data 产生一个脉冲,表示 1 bit 数据解码完毕,data 读取解码数据为 0. 波形中可以看出 en\_data 共有 14 次脉冲,即解码得到了 14 bit 的数据. 最终 finish\_decode 产生一个脉冲,表示解码结束. 整个解码过程中 error\_decode 始终为低电平,表示无错误产生.

4. 4 覆盖率分析

覆盖率分析是一种衡量验证效果的有效手段. 在验证过程中,收集覆盖率,必须先要在验证平台中编写覆盖点,仿真器对编写的覆盖点进行覆盖率统计. 一个简单的覆盖组编写代码如下:

```
covergroup cover_arg;
  cov_r_del: coverpoint r_del { ignore_bins c_del =
```

```
{[11:15]}; }
  cov_r_tari: coverpoint r_tari { ignore_bins c_tari =
{[4:7]}; }
  ....
  cov_f_TRcal: coverpoint f_TRcal {
    bins no_RTcal = {0};bins have_RTcal = {1};}
  ....
  cov_tari_pw:cross cov_r_tari,cov_r_pw;
  cov_tari_RTcal:cross cov_r_tari,cov_r_RTcal;
  cov_tra_i_x:cross cov_r_tari,cov_r_x;
  cov_RTcal_TRcal:cross cov_r_RTcal,cov_r_TRcal;
endgroup: cover_arg
```

程序中 cov\_r\_del 覆盖点对 r\_del 变量数据的收集,忽略了该变量不会产生的数据,即这些数据不在覆盖率统计内. 然后对其余数据进行自动分仓(bins). 程序下半部分通过 cross 进行了交叉覆盖率统计,这样就可以查看各种数据组合情况的覆盖率了. 这种交叉覆盖率统计的方式对测试激励更大程度上提高了要求,DUT 的更多工作情况将可以得到验证.

覆盖组编写好后,放在 Coverage 中,Driver 驱

动器在每次将事务送出后,通过 callback 回调 Coverage,进行覆盖率统计.

经过大量的激励测试,得到覆盖率报告,报告如图 6 所示.从报告中可以看到,cover\_arg 覆盖率组中的各个变量覆盖点以及交叉覆盖点的覆盖率均达到了 100%,最终总覆盖率为 100%,达到了期望的覆盖率.这说明测试点已经得到了充分的验证<sup>[6]</sup>.在验证过程中,如果发现还未被覆盖的地方,可以通过增加特定的测试激励,实现覆盖,直到最终覆盖率为 100%为止.

Questa Covergroup Coverage Report

Scope: /my\_pkg/my\_coverage

Show All Show Covered Show Missing

| Covergroups / Instances                | Total Bins | Covered Bins | Goal    | Coverage | % of Goal |
|----------------------------------------|------------|--------------|---------|----------|-----------|
| Covergroup cover_arg                   | 345        | 345          | 100.00% | 100.00%  | 100.00%   |
| Instance /my_pkg/my_coverage/cover_arg | 345        | 345          | 100.00% | 100.00%  | 100.00%   |

|                                                     |          |         |         |
|-----------------------------------------------------|----------|---------|---------|
| Covergroup type: cover_arg                          | 100.00%  | 100.00% | 100.00% |
| Coverpoints / Bins                                  | At Least | Hits    | Goal    |
| Coverpoint: cov_r_del                               |          |         | 100.00% |
| Coverpoint: cov_r_tari                              |          |         | 100.00% |
| Coverpoint: cov_r_pw                                |          |         | 100.00% |
| Coverpoint: cov_r_RTcal                             |          |         | 100.00% |
| Coverpoint: cov_r_TReal                             |          |         | 100.00% |
| Coverpoint: cov_r_x                                 |          |         | 100.00% |
| Coverpoint: cov_r_TReal                             |          |         | 100.00% |
| Coverpoint: cov_f_cg                                |          |         | 100.00% |
| Coverpoint: cov_b_cnt                               |          |         | 100.00% |
| Cross: cov_tari_pw<br><cov_r_tari,cov_r_pw>         |          |         | 100.00% |
| Cross: cov_tari_RTcal<br><cov_r_tari,cov_r_RTcal>   |          |         | 100.00% |
| Cross: cov_tari_x<br><cov_r_tari,cov_r_x>           |          |         | 100.00% |
| Cross: cov_RTcal_TReal<br><cov_r_RTcal,cov_r_TReal> |          |         | 100.00% |

图 6 覆盖率报告

5 结束语

通过采用 UVM 验证方法学设计的验证平台,与传统的验证平台相比,对于不同工程具备相同特

征的,可以实现部分复用;受约束的随机激励可以有效覆盖人工难以预期的边界条件;覆盖率分析有助于验证的完备性.这缩短了开发周期,提高了验证效率.验证结果表明,该验证平台的覆盖率满足了出口条件,达到了预期效果,有效提高了验证效率和质量.该验证平台的一些思想,为其他系统级的芯片验证提供了很好借鉴意义.

参考文献:

[1] 罗莉,何鸿君,窦强,等.覆盖率驱动的芯片功能验证设计与实现[J].计算机工程与技术,2013,35(1):36-40.

[2] 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.

[3] Accellera, Universal Verification Methodology(UVM) 1.1 User’s Guide[S]. 2011.

[4] EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860MHz-960MHz Version 1.2.0[S]. 2008.

[5] 克里斯 斯皮尔. System Verilog 验证测试平台编写指南[M].北京:科学出版社,2009:78-87.

[6] 朱车壮,陈岚.基于覆盖率驱动的 SoC 验证技术研究[J].微电子学与计算机,2011,28(11):48-51.

作者简介:

王国军 男,(1989-),硕士研究生.研究方向为大规模数字集成电路设计与测试. E-mail:wgjntu@163.com.

景为平 男,(1954-),研究员,博士生导师.研究方向为集成电路设计及其在智能交通领域的应用.

(上接第 163 页)

[5] 罗正军,罗鹏辉,王晓娟,等.基于 Linux 系统和 ZigBee 的智能家居系统[J].现代电子技术,2012,35(1):37-39.

[6] 黄媛.无线传感器网络中一种基于可靠性的数据收集算法[J].计算机工程,2015,41(2):85-90.

作者简介:

孙泽鸿 男,(1990-),硕士研究生.研究方向为通信信号处理. E-mail:912405172@qq.com.

关维国 男,(1973-),副教授,博士,硕士生导师.研究方向为移动网络定位、通信信号处理.

刘志建 男,(1990-),硕士研究生.研究方向为移动网络定位.