

一种基于 UVM 的混合信号验证环境

耿 睿¹, 王小力²

(1 西安交通大学 电子与信息工程学院, 陕西 西安, 710049; 2 西安交通大学 理学院, 陕西 西安, 710049)

摘 要: 面向一款 MCU 中的 Radio 模块子系统, 介绍了一种混合信号的验证环境, 利用 UVM 验证方法学的灵活便捷, 将主要为数字电路服务的 UVM 验证平台扩展到混合信号验证方面, 并引入模拟断言机制, 完成验证组件. 验证结果表明, UVM 作为一种验证方法同样适用于混合信号的验证工作, 并能有效发挥其优势; 同时, 利用 Verilog-AMS 为模拟电路提供的多种建模方案加强了验证可靠性.

关键词: 混合信号; UVM; Verilog-AMS; 模拟断言; 可靠性

中图分类号: TN4

文献标识码: A

文章编号: 1000—7180(2016)09—0024—04

An UVM Based Mixed-signal Verification Environment

GENG Rui¹, WANG Xiao-li²

(1 School of Electronics And Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China;

2 School of Science, Xi'an Jiaotong University, Xi'an 710049, China)

Abstract: This paper introduced a mixed-signal verification environment based on UVM, a flexible and convenient verification methodology which was usually used in digital verification task. The verification component is completed with integrating analog assertions. The verification result shows that UVM well meets the needs of mixed-signal verification and shows great performance. At the same time, multi selectable Verilog-AMS models can enhance reliability.

Key words: mixed-signal; UVM; Verilog-AMS; analog assertion; reliability

1 引言

半导体产业的快速发展正促使集成电路的设计由单个 IC、ASIC 向着更为复杂的片上系统(System on Chip, SoC)转变, 由单纯的数字或模拟芯片向混合信号的转变^[1]. 如果将模拟电路和数字电路的验证工作分开进行, 再将两者集成到一起, 从目前的技术发展看来是不切实际的, 两者之间的连接性以及协同工作表现都无法保证. 真正意义上的混合信号设计验证是从整体上的考虑, 而不是数字与模拟设计的简单相加^[2]. 而数字电路与模拟电路的描述方法不同, 不同的设计理念, 数字信号与模拟信号表示方法不同, 两者之间仿真器也不同, 这就为混合信号的设计验证工作带来了很大的挑战.

验证的目的是通过发现并修复错误来减少

DUT 中出现 bug 的可能性, 而不是证明 DUT 中不存在 bug. 因此, 如何快速的完成验证工作, 并尽可能的提高验证的完备性或可靠性, 是当前验证工作的重点. UVM(The Universal Verification Methodology)是基于 System Verilog 的验证环境开发库, 对于数字电路来说, 正逐渐成为一个约定俗成的验证方法标准^[3]. UVM 采用了最佳的验证框架去实现覆盖率驱动验证, 具有测试随机产生、自测试平台和随机化约束有效结合等特点^[4], 能有效减少验证周期并加强了验证的可靠性.

本文针对一款 MCU 中 Radio 射频子系统(sub-system), 将主要面向数字电路验证的 UVM 验证平台扩展到混合信号验证的领域, 采用了多种建模方式以增强验证可靠性, 并引入了模拟断言, 构建了一种 UVM 混合信号验证环境.

2 DUT 简介

本文所面向的 DUT 是一个带有多种无线通讯

协议,如 ANT+、Zigbee、BTLE 等的 Radio 模块, Radio 的工作模式分为发射(TX)与接收(RX)两部分,如图 1 所示。

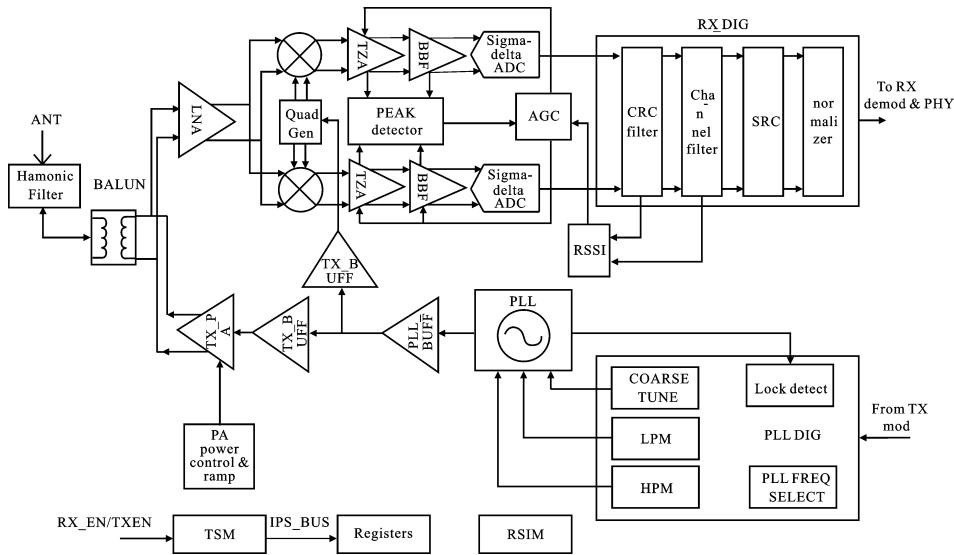


图 1 DUT 一种带有多种协议的无线通讯模块

(1)TX 阶段数据由协议调制,由 tx_dig 将基带信息 baseband_word 送到 PLL,PLL 通过两点调制为基带信号提供载波,产生信号的频率在 2.36 到 2.48 GHz 之间,经功率放大器(Power Amplifier)放大后,由 BALUN 送到天线发射出去;

(2)RX 阶段信号由天线接收,经过 BALUN 以及低噪声放大器(LNA)之后通过混频器(mixer)将载波消除降频到基带频率,经滤波放大后,通过 sigma-delta ADC 转换成数字信号进入 rx_dig、PHY 解调。

RSIM(Radio System Integration Module)提供 Radio 顶层的一些信号,如 reset 信号、参考晶振的 request 等,而整个 Radio 的时序由 TSM(Transceiver Sequence Machine)来控制。

从上述描述可以看出,这是一个复杂的混合信号芯片,不管是从内部的单个模块,如 sigma-delta ADC、2.4G PLL、自动增益控制(AGC)、直流偏移控制(DCOC)等,还是从整个 radio 收发机的工作机制来看,数字信号都紧密的与模拟信号结合在一起。

本文的测试用例(test case)所面向的为该 Radio 系统中的 PLL 以及部分 TX/RX 功能。

3 验证环境的建立

3.1 UVM 验证平台

验证平台最基本的功能在于根据要求产生不同

的激励,并检测 DUT 的输出结果,通过与期望结果的比较来判断 DUT 中是否存在功能性错误。

图 2 展示了 UVM 验证结构的树形层次化管理结构,树形结构中所示的各个单元均派生自 uvm_component。

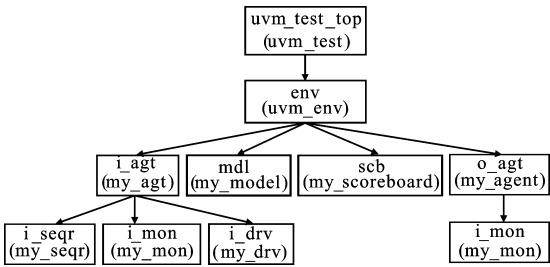


图 2 UVM 树形层次化管理结构

作为 UVM 最重要的机制之一,sequence 机制在 UVM 的验证平台中用来产生测试激励并设置 covergroup 的收集。Sequence 派生自 uvm_object,所以树形结构中并没有显示出 sequence。随着验证的进展,sequence_lib 也在不断完善,验证工作的进展过程其实就是 sequence_lib 完善的过程。Sequence 的随机化约束与求解通过 constraint 与 randomize 函数完成,并在环境外部给予随机扰乱种子(random seed)。Sequence 产生的 transaction 需要通过 sequencer 送出去,每一个 sequence 对应一个测试用例,仿真时通过+UVM_TEST_NAME 来指定 sequence 名称,并通过 connect 函数完成 se-

quence 与 sequencer 的绑定。

该验证平台通过 monitor 检测 DUT 的行为,收集 DUT 端口数据. Reference model 可以看成 DUT 的一个镜像,用来完成和 DUT 相同的作用. Score-board 通过 exp_port 和 act_port 接收从 reference model 和 DUT 的数据并完成比较,比较结果通过 uvm_error 或者 uvm_info 的报告机制,可以通过仿真生成 log 文件查看。

UVC(Universal Verification Component)是验证方法学发展可重用化 VIP(Verification IP)的成功案例之一^[5]。所以,前人所做的一些基于总线的 UVC 如 ips_uvc 或者寄存器模型(register model)等就可以直接在其基础上使用。但是由于数字信号与模拟信号性质上的差异,如何发展 analog UVC 是 UVM 在混合信号验证领域的难点。以 behavior model 为例,高速度带来的低精度使得电路完成不了发送到接收的一整条路径。本文方法就是利用 force 语句,直接将数据信息通过一定的算法转化成模拟信息给对应的模拟端口或者节点赋值:

```
force `I_in_p1 = mix_i_p * sfact * 1e9 * gain * iq_scale;
force `Q_in_p1 = mix_q_p * sfact * 1e9 * gain * iq_scale;
```

对于 AMS model 以及 spice 网表来说,模拟电路的功能是以上电与掉电的方式来实现。由模拟信号的连续性,从 UVC 到 DUT 利用 Vsrc_switched_spice 通过配置电平变化的斜率驱动 analog 部分;同样,由于数字信号的离散性,从 DUT 到 UVC 利用 check_analog_range 通过判断信号电平是否符合期望值预期范围或通过采样将连续转换成离散信号进行处理。模拟的 electrical 型信号转化成数字的 real 型信号后,也会有利于覆盖率的收集:

```
covergroup pmc_vref_lvw_cg
option, per_instance=1;
coverpoint pmc_vif, vref_hv {
bins vref_lvw1= {[3.55;3.65]};
bins vref_lvw2= {[3.85;4.05]};
bins vref_lvw3= {[4.15;4.45]};
}
```

3.2 可配置模拟模型

一个完整的验证环境不仅有 testbench,相比于数字电路的功能验证来说,混合信号的验证有一个非常重要而且繁琐的任务,即需要对模拟电路进行建模(modeling)处理。

同 design 一样,modeling 也需要考虑到精度、速度等因素的影响。本文所介绍的环境中配置了三

种 analog model 来满足不同的验证需求:

(1) 行为级模型 (behavior model), behavior model 的精度最低,仿真速度最快,可以满足大部分情况下的仿真需求。Behavior model 实际上是用数字电路的方法来模拟模拟电路的运行,其与数字部分的端口如 enable、trim 等均以 logic 型表示,而对模拟信号采用 real 或 wreal 进行描述;

(2) AMS model 的仿真速度较慢,精度较高,对于一些特定需求的 test case 适用,如需要 PLL 处于精确 lock 状态的时候。AMS model 的端口信号均用 electrical 型信号表述,会包含一些电压电流甚至负载的信息;

(3) spice 网表则代表了实际情况,但是仿真速度较慢,通常会有几十个小时甚至数日的仿真时间,并且仿真波形也很大,在较大程度上占用服务器资源,所以一般会在项目后期挑选一些关键 case 来配合验证发现使用 model 没有发现的 bug。用 spice 网表进行验证会生成很大的仿真波形,甚至能达到几十到上百 G。因此,为了实现仿真,会将一些关键信号加入 probe. tcl 中来控制仿真波形的输出:

```
probe -create -database wavetestbench -depth 1 -tasks -
functions -all
```

本文对于 DUT 的配置情况,如图 3 所示。环境中对不同的模型配置不同的 bc(block config)文件。根据不同的验证需求,在 command line 中配置不同的 bc 来选取对应的 model。由于 AMS model 端口的 electrical 型表述方法,很大程度上与 spice 网表是兼容的,因此,特定情况下也会有 AMS model+spice 的 case,称之为 AMS-mixed-spice 仿真环境。在 Spice. scs 中指定替换为 spice 网表的部分:

```
amsd{
portmap subckt=a2_rf_anatop autobus=yes busdelim
="_" casemap=keep portcase=keep
config inst= testbench. top. a_ip_2p4ghz_transceiver_
c90tfs. IRF_ANATOP
use=spice
}
```

由于数字信号与模拟信号的表述方法不同,这会导致两者之间连接处比较复杂。当采用 AMS model 或 spice 网表进行仿真时,插入如下的 A2D/D2A(analog to digital/digital to analog)的连接模型(connect model)来完成数字信号与模拟信号之间的转换:

```
analog begin
Vds = V(\vdd! ,\vss! );
```

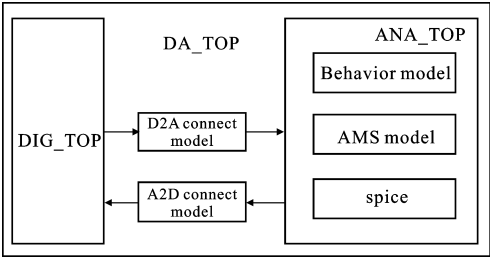


图 3 DUT 配置情况

```
Vas = V(Ain,\vss! );
@(above(Vds - vsup_min, ttol, vtol))
Vdsflag = 1;
@(above(vsup_off - Vds, ttol, vtol))
Vdsflag = 0;
Kin=Vas/max(Vds,1m) * transition(Vdsflag,tdelay,
tr,tf,ttol_t);
end
```

3.3 自检测并发断言

一个优秀的测试平台除了产生高质量的激励之外,还应该提供优秀的自检机制^[6]. 自检过程通常着眼于两个特定的方面:协议检验,即控制信号的正确性;以及数据检验,即正在处理数据的完整性.

通常,SVA(System Verilog Assertion)会广泛的应用在数字电路中提供电路的自检测. 本文将 SVA 通过一定的扩展,将模拟领域连续的 electrical 型信号转化成离散的 real 型信号,加入混合信号验证环境中用以实现混合电路一些关键信号的自检测,如观察电源电压是否正确上电掉电,平均电流消耗是否过大,关键信号是否有毛刺等等. 使用 \$ cds_get_analog_value() 来将所期望的模拟信号转换成相应的 real 型信号,便于后续的处理. 利用 absdelta() 生成的 clk,可以实现采样时钟的自适应,即模拟信号缓变的时候采样时钟频率低,模拟信号突变的时候采样时钟频率高. 同时,absdelta() 的引入不会影响到模拟仿真器仿真速度与仿真步长^[7]. 一个简单的 analog assertion,如下所示:

```
propertycheck_vdd_lv_pll_ok;
disableiff(pll_monitors_disable)
@(posedge ref_clk)
$ rose(pll_ldo_en_chk) | -> # RAMP_UP(vdd_lv_
pll_ok);
endproperty
assert_check_vdd_lv_pll_ok;
assertproperty (check_vdd_lv_pll_ok)
else `ERROR("check vdd_lv_pll result is failed");
```

经过以上说明,本文基于 UVM 的混合信号验

证环境,如图 4 所示.

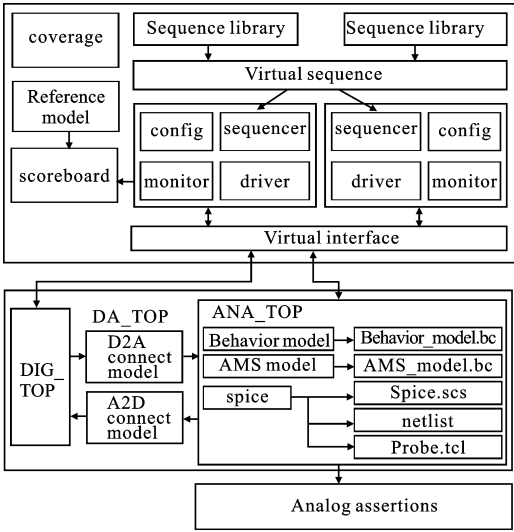


图 4 基于 UVM 的混合信号验证环境

3.4 验证结果分析

本文根据测试计划(test plan)对测试用例以及模型的使用进行了分类,如表 1 所示. 从表 1 中可以看出,大部分的验证测试用例是以 behavior model 为验证模型,这样可以快速的完成仿真工作. Pll_lock_detect 会用到 AMS model 是因为只有在使用 AMS model 或者是 spice 网表时 PLL 才会精准的锁定频率,否则会有 1 MHz 左右的频率误差. 前面提到,由于 TX/RX 协议层的验证是采用 I/Q injection 或者 force AMS model 混频器处信号的验证方法,将跳过 mixer、BALUN、天线等部件,所以会将 tx_rx_basic 一些基本测试用例使用 AMS 或者是 spice 进行验证,从而提高验证的可靠性.

表 1 测试用例

测试用例	模型	功能描述
pll_channel map	Behavior	各种协议各个 channel 的频率是否正确
pll_calibration	Behavior	PLL ctune 和 HPM calibration 结果是否正确
pll_high_port	Behavior	high port modulate 的结果是否正确
pll_low_port	Behavior	low port modulate 的结果是否正确
pll_lock_dectet	behavior/ AMS	ctune,frequency target,cy- cle slip 是否正确 lock
fast_warm_up	behavior	fast warm up 模式下的工 作情况
tx_rx_basic	AMS/spice	测试从协议到天线基本 功能
scan_test	spice	测试 scan mode 下信号是 否 safe 以及电流消耗

仿真分析表明,解码结果与码表一致,同时能够检查 RD 值或者是单比特位的错误.表 1 给出了与同类解码电路设计的逻辑资源占用和最高工作频率比较(下面的比较数据使用到了 TSMC130 工艺下的标准单元,由于该模块收外部的触发器控制,主要频率受总体的限制),从中可以看出,本文的并行逻辑优化解码方法使用的逻辑资源为 30LE(Logic Elements),虽然使用资源有部分增加,但是编码速度达到了提高,而且能够进行错误检测,增加了其功能;同时由于 6B/5B 解码和 4B/3B 解码码并行执行,减小了电路时序路径,工作频率达到工艺下的最高速度,提高了编码速度.

表 1 解码方案指标比较

技术指标 设计选择	使用资源	最高工作频率 /MHz
文献[4]	16LE	536.48
文献[5]	10LE	500.00
文献[7]	29LE	500
本设计	30LE	625

参考文献:

[1] Widmer A X, Franaszek P A. A DC-balance, partitioned-block, 8B/10B transmission code [J]. IBM

Journal of research and development, 1983, 23 (5):441.
[2] Chen H Y, Lin C H, Jou S J. Low-jitter transmission code for 4-PAM signaling in serial links[J]. IEEE Asia-Pacific Conference on Advanced System Integrated Circuits, 2004, 8(4):334-337.
[3] 许军,许西荣. PCI-Express 中 8b/10b 编解码器的设计与实现[J]. 微电子学与计算机, 2006, 23(3):37-39.
[4] 孙科达,石旭刚,史故臣. 一种新的 8B/10B 编解码方案设计与实现[J]. 光通信技术, 2009, 33(7):27-30.
[5] 武小强,田小平. 一种基于 FPGA 的 8B/10B 编解码电路的设计与实现[J]. 西安邮电学院报, 2010, 15(5): 27-29.
[6] 刘智,宁红英,王普昌. 一种新的 8B/10B 编码电路设计[J]. 通信技术, 2009, 42(7):27-30.
[7] 刘文杰,施仨,郭林,等,一种新的 8B/10B 编解码设计[J]. 光通信技术, 2012, 36(12):52-55.

作者简介:

邹陈女, (1990-), 硕士研究生. 研究方向为研究数字集成电路设计. E-mail: sandra@mail. edu. cn.
黄鲁男, (1961-), 副教授, 硕士生导师. 研究方向为混合信号 IC 芯片设计.
张步青男, (1981-), 博士. 研究方向为模拟与射频集成电路设计.

(上接第 27 页)

与传统的 SV 验证相比,本文利用 UVM 验证方法学优势产生随机化约束,并以 100% 的覆盖率为目标不断完善测试用例,在可重用性方面既可以承用前人的 UVC,又为以后的 IP 验证工作提供了方便.

4 结束语

本文面向一款 MCU 中的 Radio 模块子系统,介绍了一种基于 UVM 验证方法学的混合信号验证环境,并成功应用于 Freescale(现 NXP)半导体 KW41 MCU 验证工作中.验证结果表明,该方法可以有效的发挥 UVM 验证方法学的在随机化约束、可重用性等方面的优势,可配置 Verilog-AMS 模型和模拟断言的引入使得验证更具可靠性和完备性,为混合信号的验证工作提供了一种思路和解决办法.

参考文献:

[1] Chang H, Kundert K. Verification of complex analog and RF IC designs [J]. Proceedings of the IEEE, 2007, 95(3):622-639.
[2] 杨华中,汪蕙,侯焕焕. 十亿晶体管级芯片系统集成所面

临的挑战与机遇[J]. 半导体情报, 2001, 38(1):2-4.
[3] Liang C, Zhong G, Huang S, et al. UVM-AMS based sub-system verification of wireless power receiver SoC [C]// Solid-State and Integrated Circuit Technology (ICSICT), 2014 12th IEEE International Conference on. China, Guilin, IEEE, 2014: 1-3.
[4] 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.
[5] Janick Bergeron. Writing testbench using system verilog[M]. Berlin: Synopsys, Inc, Springer, 2008.
[6] Srikanth Vijayaraghavan, Meyyappan Ramanathan. SystemVerilog Assertions 应用指南[M]. 陈俊杰,译. 北京:清华大学出版社, 2006.
[7] Accellera. Verilog-AMS language reference manual [M]. Napa, Accellera Systems Initiative Inc. 2014.

作者简介:

耿睿男, (1992-), 硕士研究生. 研究方向为混合信号设计验证. E-mail: mrgengrui@163. com.
王小力男, (1956-), 教授, 博士生导师. 研究方向为高性能 VLSI 与优化设计、集成化芯片系统设计与基础研究、光纤通讯与光波分复用(WDM)技术.