

基于 VMM 的 SOC 可重用验证平台设计

肖庚亮

(华南理工大学电子与信息学院, 广州 510641)

摘 要 当前集成电路设计规模空前增长,对验证提出了巨大挑战,搭建组件化和可重用的验证平台已逐渐被业界采用。以一款接入交换机芯片验证为例,介绍了应用 VMM 方法学搭建基于 subenv 的组件化和可重用的验证平台的方法和原则。项目实践表明此方法可以极大地提高验证效率。

关键词 验证方法学 子环境 重用性 组件化

中图法分类号 TN407; **文献标志码** B

当前集成电路设计规模越来越大,复杂度也急剧增长。对验证提出了巨大挑战,验证已成为芯片设计的瓶颈,验证工作可占整个芯片设计 70% 的工作量。目前验证方法主要有仿真验证和 FPGA 验证。FPGA 验证在此不作讨论。仿真验证就是利用专门的 EDA 工具对用 RTL 描述的电路(DUT)施加各种激励,通过检查 DUT 的响应,来判断其功能的正确性。仿真验证可分为两个阶段:前仿和后仿。本文要讨论的是前仿,也称为功能仿真。目前业界大多采用的前仿真流程可分为单元仿真(UT),集成仿真(IT),系统仿真(ST)三个阶段。UT 是指对单一模块内部功能的仿真;IT 是对多个相关模块组成的子系统仿真;ST 则是对整个芯片进行仿真。当前芯片设计规模的庞大,单元模块数动辄几十上百甚至上千,而且还嵌入了 IP,同一模块也可能例化多个,因此决定各阶段搭建的平台必须具有可重用性,以减少大量的重复工作。芯片的各个 UT 环境作为 IT 环境的组件,各 IT 环境又作为整个 ST 环境的组件,这样层层复用的思想已被业界广泛应用。

1 VMM 验证平台

VMM(Verification Methodology Manual)是一种基于 SystemVerilog 的验证方法学,同时也是包含了搭建一个验证环境需要的基本类(class)所组成的类库的验证平台,包括激励、事务(transaction)、通道(channel),记分板(scoreboard),基本环境等组件。这些组件都对应 vmm. sv 库中一个基本类。基于 SystemVerilog 具有面向对象的特点,在搭建验证环境时需要根据实际项目验证环境的需求调用 VMM 库中的基本类,然后作相应的扩展(extend)实现所需要的功能。

VMM 类库中常用的基本类有 vmm_data, vmm_channel, vmm_xactor, vmm_env 等。vmm_data 是所有事务描述符和数据模型的基础,提供仿真激励基本的操作。vmm_channel 提供各种组件接口机制,可以在各个组件间传递扩展自 vmm_data 任意数据类型,在 VMM 验证平台 <vmm_data>_channel 操作可以直接实现各组件间传递数据激励。vmm_xactor 是所有事务处理的基础,包括总线功能模型(BFM)、监视器、发生器,提供了一套标准的控制机制。vmm_env 是实现验证环境的基本类,采用九步顺序执行机制 gen_cfg、build、reset_dut、cfg_dut、start、

2009 年 8 月 31 日收到

作者简介:肖庚亮(1983—),男,汉族,江西萍乡人,硕士研究生,研究方向:SOC 验证。E-mail:rubenshaw126@126.com。

wait_for_end、stop、cleanup、report,所有事务都在 vmm_env 生成例化并且将实例按照验证模型连接,在 vmm_env 框架中运行。本文所述的重用机制, vmm_subenv 起重要作用,它扩展自 vmm_env 类,具有 vmm_env 类的属性,是环境组件化的基础,其实体可例化在上层环境中,以实现重用。

2 基于 VMM 的验证架构

从图 1 可以看出 VMM 推荐的层次化验证平台从底至上被分成信号层、命令层、功能层、场景层以及测试层。其中的验证组件包括发生器、交易器、驱动器、监视器、检查器、记分板以及断言。

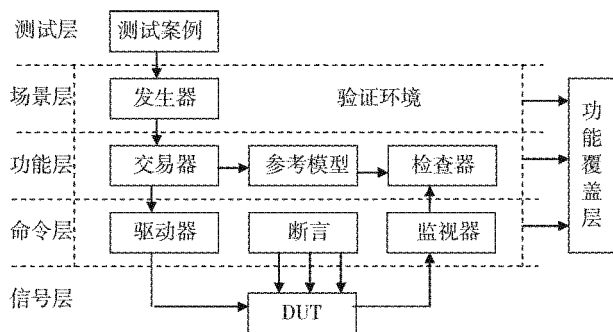


图 1 基于 VMM 的验证架构

3 基于 VMM 可重用验证环境实例

本实例为接入交换机芯片。该芯片是基于存储转发原理实现,具有 24 个 FE 口和 1 个 utopia level 2 口作为接入口,上行具有 2 个支持 1 G 速率的 GE 口,支持 Ethnet 与 ADSL 接入,支持以太网协议,PPP 协议和 vlan,其架构如图 2。

本芯片为典型的 SOC 架构,主要分为两部分:业务处理子系统 (SPS) 和嵌入 CPU 子系统 (ECS);其中 ECS 主要由 ARM926 核、AHB 总线、APB 外设 (看门狗, UART, 定时器), AHB 总线与 APB 总线通过 AHB2APB 桥连接。SPS 主要由包交换引擎 (PSE)、各接口模块、缓存管理模块 (BM), CBUS 总线组成。PSE 通过 cbus2ahb 桥连接,以便 CPU 对 SPS 中的寄存器和表项进行配置和管理。由于本芯

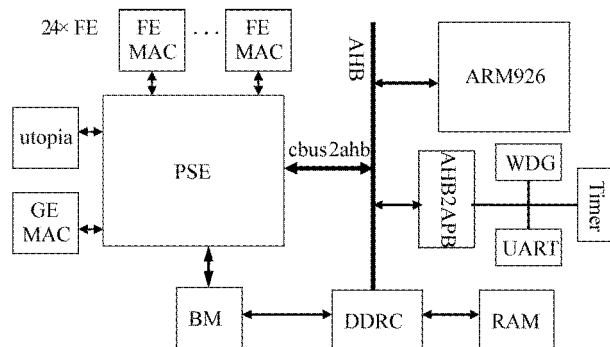


图 2 实例接入交换机芯片架构

片是基于存储转发原理,各端口进来的报文通过 BM 统一传送给 DDR 控制器 (DDRC) 再存入 DDRAM。其中 ECS 内部各模块都为成熟 IP,因此验证工作主要集中在 SPS 各模块。

3.1 UT 环境搭建步骤和方法

为了重用,UT 环境必须遵循以下原则。第一、各 UT 环境中各组件间传递的数据必须定义成统一的类;第二、UT 环境做成 subenv 的形式。对于第一点,以 utopia UT 环境为例,伪代码如下:

```
class top_pkt extend vmm_data;
c_atm_pkt atm_pkt; // c_atm_pkt, c_eth_pkt 分别为一种报文格式类
c_eth_pkt eth_pkt;
...

function new( type pkt_type )// 按 pkt_type 的类型构造
    If( pkt_type == ETH )
        eth_pkt = new();
    ...
Endfunction
...
endclass
```

这个统一的类就是 top_pkt, 本身只是个外壳, 其内部例化各 UT 环境中用到的数据格式, 以便使用同一种 channel 在各 UT 环境中进行无缝传输。各 UT 环境的搭建, 伪代码如下:

```
typedef utp_gen_key { UTP_GEN_ON, UTP_GEN_OFF }; // 定义组件开关
class c_otp_cfg
    utp_gen_key utp_gen; // utp_gen_key 为开关变量, 定义环境中各组件开关
    ...
```

```

endclass
class c_otp_subenv extend vmm_subenv;
c_otp_cfg otp_cfg;
c_otp_gen otp_gen; //定义环境中各组件,如激励,BFM,
Checker 等
...
endclass
class otp_env extend vmm_subenv;
c_otp_subenv m_otp_subenv;
...
endclass

```

c_otp_cfg 配置类定义环境中各组件开关,可以根据实际需要开/关各验证组件。各组件和配置类在 subenv 中例化,c_otp_subenv 类最终在 UT 环境中例化,总体架构如图 3。

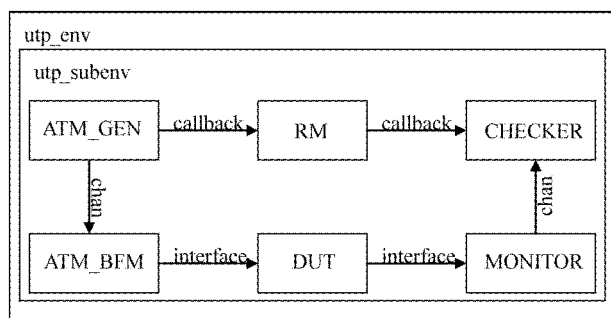


图 3 utopia 接口 UT 验证环境架构

3.2 IT 环境搭建步骤和方法

本例的 IT 环境以各 UT 环境为组件,各 UT 的 subenv 类都不需要改动而直接重用。IT 环境要做的只需将各 UT 的 subenv 类例化,DUT 有多个相同的模块也只要将同一个类例化多个,然后根据 DUT 内部各模块的连接关系将各 UT 的 RM 连接起来组成 IT 的 RM。本例中分为两个 IT,业务处理子系统 (SPS),ARM 子系统 (ECS)。ARM 子系统包括的 ARM 核和 amba3.0 总线是比较成熟的 IP,不在 IT 中独立验证。SPS 中具有 utp_subenv,fe_subenv,ge_subenv,pse_subenv,bm_subenv 等组件。各 subenv 都在 SPS 环境中例化,其内部组件重新进行合理连接。SPS IT 环境架构如图 4。

如图 4 所示,IT 中的 generator,channel,interface,BFM,checker 等组件都是重用自各 UT 相

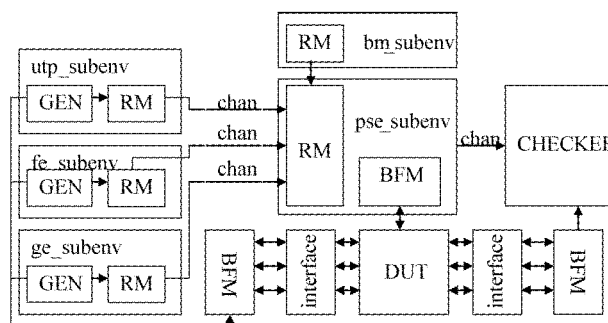


图 4 业务处理系统 (SPS) IT 验证环境架构

应组件,各 UT 的 RM 按实际芯片要求连接形成 IT 的 RM,除此之外只需例化整个芯片顶层到 IT 环境中。

3.3 验证过程及结果

UT 验证阶段:本例三个接口模块 FE/GE MAC, utopia 和 DDR3 都是采用成熟的 IP,在 UT 中不验证。UT 重点验证 PSE,BM 模块。以 BM 为例,BM 的 UT 环境和图 3 类似,编写激励 BM_GEN,它与 DUT (BM) 有两个接口,一个申请缓存接口,另一个为释放接口;BM_GEN 模拟生成大量的申请和释放请求,通过 REQ_BFM 和 RLS_BFM 转化为底层的时序信号来驱动 DUT。需要重点验证的是申请的地址最终是否得到释放以及 BM 的带宽。这两点可通过编写一个 CHECKER 来实现,在 CHECKER 中有队列存储申请和释放的地址和长度,同时记录仿真时间,到仿真结束时比较两队列的数据,统计申请和释放的带宽。

IT 验证阶段:由于 PSE 和 BM 在已在 UT 阶段中开发了 subenv 式的 UT 环境,在 IT 环境中可直接例化各组件,因此 IT 只需开发 GE/FEMAC, utopia 等接口模块的 generator,rm,bfm 及 subenv,节省大量的工作量。各模块子环境按图 4 连接组成 IT 环境。各模块接口之间挂有 monitor 用来收集数据,将收集到数据和相应的 RM 输出数据比较便可方便的定位 DUT 的问题。本例代码行数统计见表 1 及表 2。由统计数据可算出 IT 代码重用率达到 54%。

表 1 PSE 和 BM UT 环境代码统计表

模块	组件						
	GEN	BFM	RM	ENV	SUBENV	可重用	总计 重用率
PSE	981	3 728	6 453	1 045	1 721	11 902	13 928 85.45%
BM	437	675	843	562	714	2 232	3 231 69.08%

表 2 IT 环境代码统计表

模块	组件			
	BFM	RM	SUBENV	总计
GE MAC	963	1 235	1 081	3 279
FE MAC	852	1 105	917	2 874
utopia	873	1 389	1 142	3 404
DDRC	725	934	815	2 474

4 结 束 语

本文在大规模集成电路设计中验证成为设计的瓶颈的背景下,以一款接入交换机芯片为例,提出了

基于 VMM subenv 组件重用的验证方法,该方法在实际应用中表明,能够大幅减少工作量,提高验证效率。

参 考 文 献

1 Michael K, Pierre B. Reuse methodology manual for system on Chip designs . 3rd ed. Boston; Kluwer Academic Publishers, 2002; 239—263

2 Bergeron J, Cerny E, Hunter A, *et al.* Verification Methodology Manual for System Verilog. New York; Springer US, 2005; 231—254

3 Spears C. System verilog for verification. New York; Springer US, 2006; 138—192

4 徐英伟,刘 佳. SoC 功能验证的特点和方法. 微处理机, 2006 , 28(2) ; 11—13

5 杜 宁,郑建宏. 一种可重用的 SOC 验证平台. 微计算机信息, 2008 , 24(2) ; 56— 58

6 陈 辉,申 敏. 结合覆盖率驱动技术的 RVM 验证方法学在 SOC 验证中的应用. 微计算机信息, 2005 , 10-2; 120—121

Design of Reusable Testbench for SOC
Verification Based VMM

XIAO Geng-liang

(School of Electronic and Information Engineering, South China University of Technology, Guangzhou 510641, P. R. China)

[Abstract] With the rapid growth of the scale of integrated circuit, verification becomes a great challenge. Building componentized and reusable testbench is now more and more being used by industry. A kind of methodology of building componentized and reusable testbench with a example of switch IC are introduced. This kind of methodology is proved to be very usefull to improve verification efficiency in programs.

[Key words] VMM subenv reusability componentization