

DOI:10.19651/j.cnki.emt.2107199

基于 SystemVerilog 的图像采集 压缩卡芯片验证平台设计*

王凯^{1,2,3} 王骞^{1,2} 符云越^{1,2} 李拓^{1,2,3} 刘凯^{1,2,3}(1. 山东海量信息技术研究院 济南 250098; 2. 浪潮电子信息产业股份有限公司 高效能服务器和存储技术
国家重点实验室 济南 250101; 3. 山东浪潮人工智能研究院有限公司 济南 250101)

摘要: 验证平台对视频采集压缩卡芯片的开发设计有重要作用。针对传统的验证平台在代码覆盖率以及测试效率方面存在的不足,设计了一款基于 SystemVerilog 搭建的验证平台,该验证平台采用面向对象程序语言设计,其中,PCIe host(RP)端采用 Xilinx IP 建模链路层和物理层,保证了 PCIe 总线环境与真实主机板卡环境相同;外部验证环境采用 SystemVerilog 分层设计的方法,并采用类思想进行上层验证环境设计,使较多验证组件能够移植至同一接口协议的不同类 SoC;此外,在自动化验证阶段,通过仿真报告自动判断 case 状态,调整随机基准以及在覆盖率报告中追踪未覆盖模块路径,极大地改善了代码的边角覆盖情况,加速了回归收敛。从采集压缩仿真过程、验证自动化以及覆盖率 3 个方面对该验证平台进行了分析,结果表明,该验证平台可快速完成相似设计的验证模组横向移植,提高相似功能芯片的验证可靠性,节省人力,加快仿真进度,加速覆盖率收敛,缩短验证周期,增加流片成功率。

关键词: SystemVerilog;功能验证;自动化验证;覆盖率收敛

中图分类号: TN402;TP391.9 **文献标识码:** A **国家标准学科分类代码:** 510;520

Verification platform of image compression card chip based on SystemVerilog

Wang Kai^{1,2,3} Wang Qian^{1,2} Fu Yunyue^{1,2} Li Tuo^{1,2,3} Liu Kai^{1,2,3}(1. Shandong Massive Information Technology Research Institute, Jinan 250098, China; 2. State Key Laboratory of
High-end & Storage Technology, Inspur Electronic Information Industry Co., Ltd., Jinan 250101, China;
3. Shandong Inspur Artificial Intelligence Research Institute Co., Ltd., Jinan 250101, China)

Abstract: The verification platform plays an important role in the design of the video acquisition and compression card chip. Aiming at the shortcomings of traditional verification platforms in terms of code coverage and test efficiency, designed a SystemVerilog-based verification platform. The verification platform is designed with an object-oriented programming language, where the PCIe host (RP) adopts the Xilinx IP modeling link layer and the physical layer, ensuring that the PCIe bus environment is the same as the real host board card environment. The external verification environment adopts the SystemVerilog hierarchical design method, and adopts the class idea to design the upper-layer verification environment, so that more verification components can be transplanted to different types of SoC of the same interface protocol. Moreover, in the automated verification stage, the case state is automatically judged by the simulation report, adjusting the random benchmark and tracking the uncovered module paths in the coverage report, which greatly improves the corner coverage of the code and accelerates the regression convergence. The verification platform is analyzed from three aspects: acquisition and compression simulation process, verification automation, and coverage. The results show that the verification platform can quickly complete the horizontal verification module transplantation of similar design, improve the verification reliability of similar function chips, save manpower, accelerate the simulation progress, accelerate the coverage convergence, shorten the verification cycle and increase the success rate of flow chip.

Keywords: SystemVerilog;functional verification;automated verification;coverage convergence

0 引言

随着 SoC 和 FPGA 设计规模的不断增大,IP 核的独立

正确性成为了系统正常工作的基本保证^[1]。在芯片设计的整体流程中,验证占据了 70% 的工作量^[2]。采用 SystemVerilog 验证语言,基于事务级验证技术可在大程度

收稿日期: 2021-07-07

* 基金项目: 山东省重大科技创新工程项目(2019JZZY010103)资助

上提高功能验证的效率^[3]。较采用传统验证法验证显示输入模块时,难以对实际工作中的帧数据和故障进行有效的建模,并且对验证结果正确性分析非常复杂。此外,采用传统验证法验证效率较低,而且验证覆盖率难以保证。因此,为了提高验证覆盖率、缩短验证工作周期,采用更先进的验证方法非常有必要^[4]。图像处理硬件化研究不仅是芯片设计的重要前提更是加速研究中的重要处理手段,其对实现图像算法的实时处理具有重大的现实价值^[5]。仿真验证需要编写定向测试组件,难以建立高复用性测试平台,人工搭建测试平台耗时占整个功能验证周期的 20%~30%。要提高验证效率,必须对测试平台搭建进行优化和改进^[6]。

文献[4]中提出基于 UVM 的显示控制输入模块验证平台的设计,输入时序丰富,但缺少 VIP (intellectual property) 仿真模型,自动化程度较低。文献[6]提出基于编程接口的图像坏元修正验证平台,平台内缺少通用验证模组,且失败验证用例无法自调整,进入回归测试,降低了系统验证效率。本文研究了验证组件高度模组化、部分组件硬件化、回归自动化、code corner 自识别的实现方式,提出一种基于 SystemVerilog 的视频采集压缩卡芯片验证平台,适用于图像处理系统验证。经过实验分析,方案实现了提高验证效率、加速覆盖率收敛。

1 验证环境

验证环境分为 5 个层次:信号层、命令层、功能层、场景层、测试层;对类进行了划分,共有 5 个基类,分别是驱动类(driver class)、发生器类(generator class)、计分板类(scoreboard class)、监控器类(monitor class)、环境类(environment class)。并且提供了断言、覆盖率收集功能,可以更好的使测试用例覆盖代码边角(code corner),如图 1 所示。

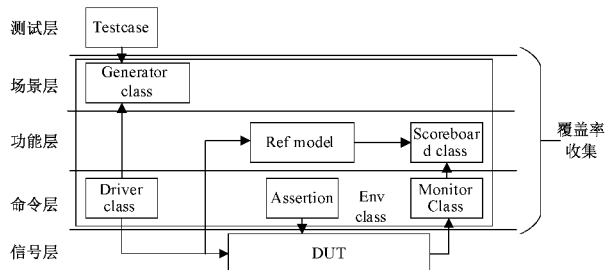


图 1 SystemVerilog 验证结构

2 待测设计 (DUT)

视频采集压缩卡芯片作为验证平台的被测设计,设计目标:1)支持 PCIe 4.0 协议;2)支持 10 M/100 M/1 000 M 以太网,支持 RGMII 接口、RMII 接口;3)支持 JPEG 压缩,压缩比至少 10:1;4)支持 VGA 传输标准。

2.1 采集压缩卡芯片整体架构

芯片的系统架构如图 2 所示,该视频采集压缩芯片采

用 SoC (system-on-chip) 设计技术。图像压缩模块采用 JPEG 处理标准,JPEG(joint photographic expert group)是 ISO 和 CCITT 联合制定的适用于连续色调、多级灰度、彩色/单色静止图像压缩的国际标准,有损压缩是 JPEG 标准的主要应用领域^[7]。总线模块采用了 AHB 总线作为片上总线,包含了 UART (universal asynchronous receiver/transmitter)、WDT (watch dog)、FMC (flash memory controller)、DDRC_0 (ddr controller 0)、GPIO (general purpose input output)、sub_clk&rst(clk&rst 集合)、PCIe-EP(peripheral component interconnect express_end point)、VGA(video graphics array)、JPEG、DMA (direct memory access)、EMAC(ethernet media access control)。

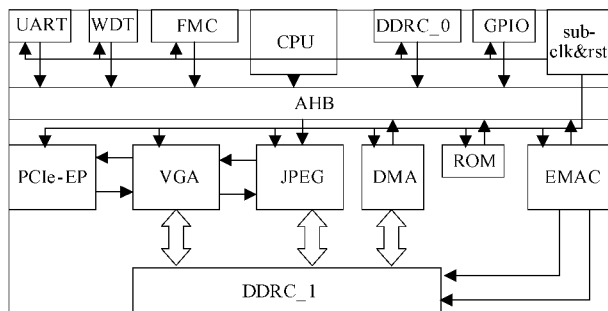


图 2 芯片架构

2.2 压缩卡芯片处理流程

采集压缩卡芯片工作流程分为 3 个阶段:自测阶段、启动阶段和使用阶段。

自测阶段:对系统中的硬件进行检测。在芯片上电初期,电压不稳,一直处于复位阶段。之后,CPU 会执行跳转指令来执行上电后的自检工作,自检工作的代码预存在 ROM 中,自检时会逐个硬件设备进行检测,硬件设备包括 CPU、存储器、时钟等,当检测到异常时,程序就会终止,并发出相应的警报。

启动阶段:Startup.s 文件是系统的启动文件,主要包括堆和栈的初始化配置、中断向量表的配置以及将程序引导到 main() 函数等。具体的,进行 CPU 7 种工作模式的堆栈初始化并分配空间,关闭内存管理单元 MMU 和 TLB,关闭写缓冲和 Cache,设置中断向量基址及控制寄存器等相关寄存器,配置时钟,配置 L1 指令 Cache 及数据 Cache,配置 TLB,设置并打开 MMU 和 Cache,跳转到 __main 中执行。

使用阶段:在 CPU 完成各个模块的初始化后,PCIe 主端发送图像数据给视频压缩卡,压缩卡将数据存于一块 64 MB 的显存 (frame buffer, FB) 内,VGA 处理 frame buffer 内的图像数据,将其转为 DVI (display video interface) 格式,并通过 DVI 信道发送给 DAC,使得可以在压缩卡本地显示,同时 JPEG 模块会实时捕获 DVI 信道的图像数据并逐帧压缩(不影响 DVI 信道传输),并将压缩后的图像通过 EMAC 发送至远端。该视频采集压缩卡使用

阶段全过程可由本地 CPU 全程控制。

3 验证方案

3.1 验证需求

本文验证目标为视频采集压缩芯片功能的正确性,重点是对视频采集与压缩的正确性进行检查。根据对视频采集压缩芯片设计需求的分析,结合 SystemVerilog 中的类与接口的特点,对视频采集压缩芯片验证需求进行了如下定义。

1)支持 RS232 全双工传输;2)支持 PCIe4.0 协议;3)支持 JPEG 图片压缩功能,压缩比不低于 10 : 1;4)支持

VGA DVI 数字图像输出、支持二层显示框架(图像光标分离传输)、支持高性能 2D 绘图引擎;5)支持 EMAC 10 M/100 M/1 000 M 模式 RGMII,RMII 接口;6)Flash SPI 最大时钟频率 100 MHz,支持 256 MB 地址空间,支持 dual/quad IO 功能;7)支持 DDR4 接口,接口为 128 bit,工作频率 266 MHz,(接口速率 1 600 M~3 200 M,颗粒 16 bit,接口位宽 16 bit),峰值带宽 8.5 GB/s,支持 ECC 校验。

3.2 验证用例列表

根据验证计划中的分析,可以得出如下基本验证用例,如表 1 所示。

表 1 基本测试用例

用例标识	验证用例描述。
case001	UART 串并转换验证。
case002	UART 全双工工作场景验证。
case003	UART 奇偶校验转换场景验证。
case004	系统启动及自测阶段打印验证。
case005	Flash controller 读写外部存储验证。
case006	Flash controller 寄存器及系统初始化验证。
case007	GPIO 通路验证。
case008	完成对 DDR 控制器初始化配置、DDR PHY 初始化配置、DDR SCL、write leveling 等功能的测试。DDR 配置为 DDR 数据读写的前提条件,需要全部通过后,才能进行下一步的数据读写测试。
case009	通过 ROM 中预存机器指令对 CPU 通用寄存器读写测试。
case010	PCIe single/burst8 模式 bar0~bar3 memory 空间访问通路验证。
case011	PCIe single/burst8 模式 bar4~bar5 memory io 空间访问通路验证。
case012	PCIe bar2~bar5 lock 之后,可读不可写;解除 lock,可读可写。
Case013	PCIe vender id、device id、revision id 读写验证。
Case014	PCIe 链路初始化与训练验证。
case015	通过 PCIe 正确读写 VGA 的功能寄存器。
case016	通过 PCIe 配置 VGA 为 8 bit RGB 模式;配置行场消隐和分辨率(640×480/800×600/1 280×1 024/1 920×1 080/1 920×1 200/1 280×800/64×64),frame buffer 起始地址;使能显示和色彩模式;配置调色板;VGA DVI 通道输出图像验证。
case017	通过 PCIe 配置 VGA 为 16 bit RGB 模式;配置行场消隐和分辨率(640×480/800×600/1 280×1 024/1 920×1 080/1 920×1 200/1 280×800/64×64),frame buffer 起始地址;使能显示和色彩模式;配置调色板;VGA DVI 通道输出图像验证。
case018	通过 PCIe 配置 VGA 为 32 bit RGB 模式;配置行场消隐和分辨率(640×480/800×600/1 280×1 024/1 920×1 080/1 920×1 200/1 280×800/64×64),frame buffer 起始地址;使能显示和色彩模式;配置调色板;VGA DVI 通道输出图像验证。
case019	JPEG buffer 模式;在 DDR 获取 VGA 处理过的缓存图像,压缩完成后,放到 EMAC 待发送缓存,对比压缩前后图像,确定压缩比。
case020	JPEG DVI 模式;在 VGA DVI 通道捕获图像数据并压缩,压缩完成后,放到 EMAC 待发送缓存内,对比压缩前后图像,确定压缩比。
case021	图像由 pic_case_generator 产生,经过 RP 应用层,通过 PCIe RP,发送给 VGA 的 frame buffer,VGA 图像处理过后,通过 DVI 通道发送给 DAC。
case022	EMAC phy 模型内部采用 fifo 将 phy 接收的数据进行缓存,fifo 非空后,产生 fifo 的读取信号,然后读取出的缓存数据再发往 SV_EMAC 模型,数据和数据有效信号对齐,发送数据时钟独立于接收时钟。由于 RGMII、RMII 接口的双沿采样,使用两倍于 phy 时钟的时钟频率对数据进行采样和读取,保证数据的正确接收和发送。

做出修改,使用断言反向追踪,设计关于断言的 IP corner&bus Assertion,在每一次总线完成读写时,追踪到全总线信号覆盖状态,完成随机测试后,将收集到的总线信号状态同源取或,确定总线缺少的 case 种类,完善 testcase,提高验证效率。

4)在 DUT 运行中,会因为裸机测试程序而产生中断,Interrupt class 是中断收集组件,DUT 有外部中断和内部中断,此组件都会收集,内部中断会通过组件的包含关系,反馈给驱动组件,驱动组件子类会基于实时的中断状态,修正驱动时序。

5)各个模块的接收组件在收到数据后,从协议和时序中将数据提取出来,组成 Package 组件类型,通过 mailbox 传递给各自模块的 Monitor 和 Scoreboard 进行监控和校对。

6)完成后,通过 Out2script 类输出最终结果同时发给脚本,记录并进行下一个 test case 的测试。

图像处理算法向数字电路移植的过程中,首先采用 C 或者 Python 等高级语言构建图像处理算法模型,然后使用 Verilog 或 SystemVerilog 代码对构建的模型进行描述和移植^[2]。JPEG Ref model 是用 SystemVerilog 编写的 JPEG 压缩数学算法的参考模型。Flash_model 使用的是镁光 N25Q Verilog 行为模型。DDR4_model 使用的是 SK hynix DDR4 Verilog-HDL 模型 2 lane,2Gx16。

4.2 视频采集与压缩验证模组

该部分验证平台较文献[10],实现了 SV 格式 PCIe 协议包及应用层验证模组设计,主要包含 PCIe 应用层 tx/rx 组包逻辑(User_app_tx/rx)、PCIe 应用层配置类(Pcie_app_config)、PCIe 应用层 com 接口类(Pcie_app_com)、EMAC 应用配置类(EMAC_app_config)、EMAC 网络协议类(EMAC_protocol_transfer)、EMAC 应用层网络接收类(EMAC_app_rx)、图像接收类、图像监控类、图像计分板类。该验证平台架构如图 6 所示。

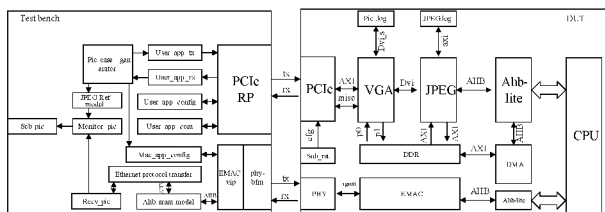


图 6 视频采集与压缩模块验证平台架构

PCIe 部分仿真流程如图 7 所示,各模组概述如下。

PCIe 应用层发送/接收组包(User_app_tx/rx)逻辑包含:PCIe 训练类,PHY 及链路层初始化类,IO/MEM、CPL、MSG 报文发生器等。

PCIe 应用层配置类(User_app_config)包含:初始化 PCIe RP 的内部消息及流控管理模块和电源管理模块类。

PCIe 应用层接口类(User_app_com)包含:解析 PCIe

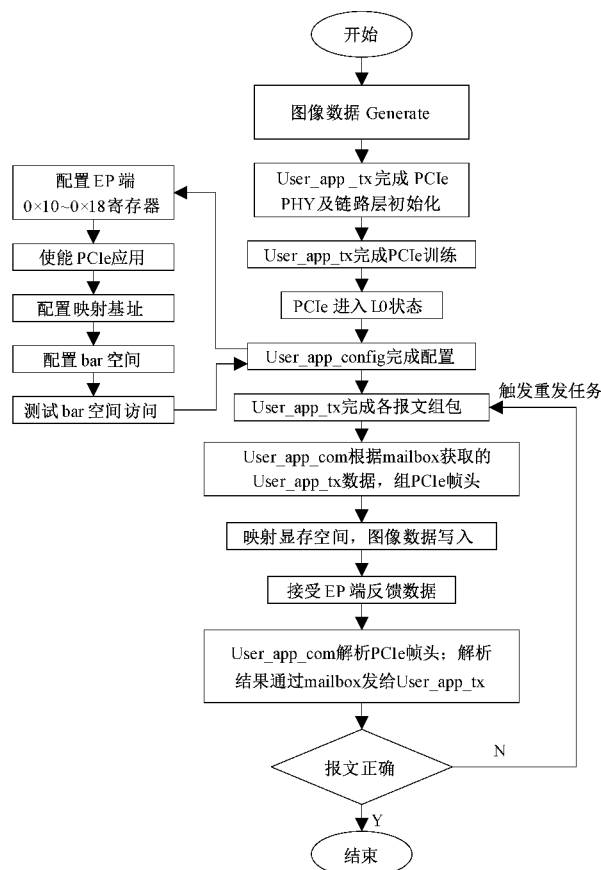


图 7 平台 PCIe 部分仿真流程

帧头数据并打印成 log,实时 display;接收 mailbox 中的 frame_store,将其中的 PCIe 协议中的 fmt、f_type、traffic_class、td、ep、attr、length 提取出来,组成 PCIe 帧,符合 PCIe 应用层时序,发送给 RP。

EMAC 部分仿真流程如图 8 所示,各模组概述如下。

EMAC 应用配置类(EMAC_app_config)包含:EMAC 初始化以及寄存器配置,包括 EMAC 通信速率选择,工作模式、描述符寄存器以及 DMA 寄存器等,并对接收描述符进行配置。

EMAC 数据接收类(EMAC_app_rx)类包含:Ahb_sram_model,用于接收描述符以及数据帧的保存。

EMAC 网络协议类(EMAC_protocol_transfer)包含:监测接收中断,并根据接收描述符信息对 Ahb_sram_model 中的以太网格式数据帧进行解包操作,以此得到包含帧头帧尾标志的完整图像数据,并将图像信息传至后续处理。

图像收集类:负责接收来自于 EMAC 中的图像数据,通过 EAV 和 SAV 这两个嵌入式控制字,分别表示有效视频的终点和起点^[11]。对行、场信息进行检测,分离出有效数据,并按照图像 id 将其分为分段存储;图像监控类:比对发送与接收的压缩后图像数据,将比对结果发送给图像计分板。图像计分板类:汇总比对数据,锁定出错图像 id,

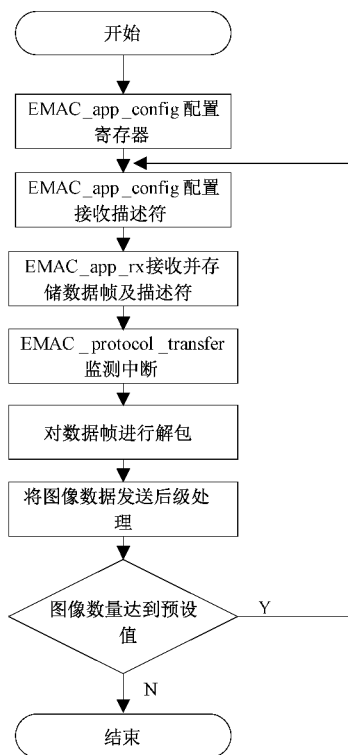


图 8 平台 EMAC 部分仿真流程

标记图像错误行列,打印错误数据与 ref model 产生的 golden 数据,打印 log;

由于需要验证 PCIe EP(end point)在 PCIe 体系结构中的功能,同时,设计该验证平台考虑到 FPGA 在做系统验证时的高效特性,因此 PCIe RP(root complex)模型直接使用 Xilinx 的 PCIe 4.0 RP IP,传输速率为 5 Gb/s,配置为单通道,修改该 IP 应用层逻辑,开放其 config 和 axi-s 接口,利用该 IP 硬件可综合的特性,实现在 FPGA 上同步验证。

4.3 自动化验证

该平台自动化验证采用的是 Tcl 加 CShell 脚本结合自动对比的方式。该验证平台在 monitor 中完成监控数据的收集,在 Scoreboard 记录对比结果和打印生成 log,利用 Tcl 脚本筛选 Scoreboard log 中的 testcase 编号等关键字信息以及仿真报告中关于时序和断言的检查结果,将过滤后的关键字信息保存成 txt 文件,CShell 脚本根据 testcase 关键字信息综合确定该验证用例是否通过。若成功,则继续下一个测试用例。若失败,SystemVerilog 环境再读取该 txt 文件,显示测试使用的随机种子(random seed),通过这个特定的种子就可以重新产生之前的激励^[12],开启 dump 波形 define。同时,CShell 脚本会 find 到该测试用例路径,将用例名及路径传递给 Tcl,Tcl 修改 SV 文件内 Generator 用例相关随机延时参数,放宽时序约束,再重跑一次 dump 波形,方便后续 debug,二次验证后,会继续下一个测试用例。进入回归测试阶段,在 Tcl 中输入覆盖率

报告,抓取为覆盖硬件逻辑的关键路径信息,在 case 中加入未覆盖逻辑的强相关赋值,皆可增强待设计的鲁棒性,亦可在回归阶段加速收敛。图像采集 testcase 和其余模块 testcase 采用的自动对比流程如图 9 所示。验证开始后,通过脚本控制,可以同时开启验证库中多个测试用例^[13]。

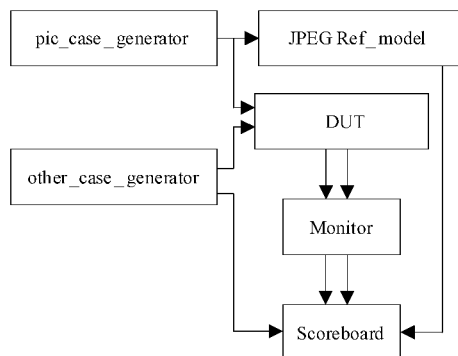


图 9 实时自动对比

5 实验过程

图像数据经过 PCIe 通道被采集压缩卡捕获。PCIe host 端将图像写入到 PCIe EP 的 bar0~bar1 空间,EP 将数据从 bar 空间提取转为 AXI 协议发送至 VGA 的 frame buffer,VGA 将再将图像数据格式从图像转为 DVI 格式,经过 display port 发给片外 DAC 模块用于本地监控显示;JPEG 从 DVI 通道或 frame buffer 内获取图像数据,并经过压缩,将其转为 JPEG 图片格式,按照 30 帧/s 的速率刷到 DDR 内,EMAC 内的 DMA 会将其读走,打包通过以太网口发送给远端。波形如图 10 所示。

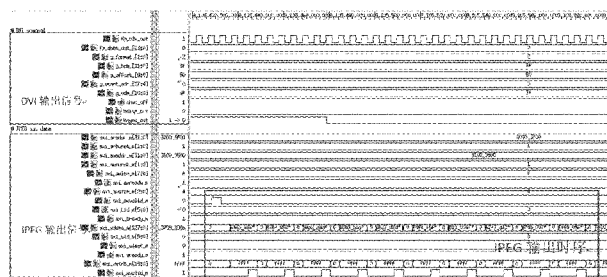


图 10 采集压缩过程波形

传送至 EMAC model 的图像与原图像经过 Ref model 压缩的 golden 比对一致,表明该分辨率下,采集压缩过程正确。

6 验证结果与分析

功能覆盖率是衡量设计特征已经被验证平台测试的一个指标^[14]。DUT 在验证平台内,经过模块级仿真、系统级仿真以及回归测试,所有验证用例功能验证均正确,功能覆盖率达到 100%。如图 11 所示,各模块覆盖率均为 100%。

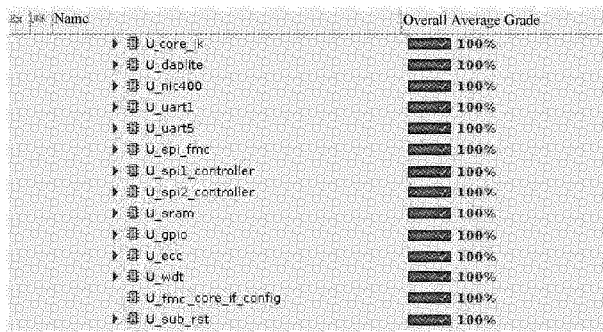


图 11 功能覆盖率

为统计设计平台的验证效率,本文引入式(1)的效率计算公式计算千行代码消耗的人时数^[13]。由表 2~4 的验证结果统计,可以看出,仿真验证阶段,项目时间周期要求为 180 d,UVN 环境平台总耗时约 181 d,验证平台总耗时约为 147 d;按每天工作 10 小时,其中 70%的时间用于测试,按照式(1)计算,UVN 平台测试效率约为 50.68 人·小时/千行,平台测试效率约为 41.16 人·小时/千行。相较于原版 UVN 搭建的环境,方案平台在基础测试覆盖率和随机测试覆盖率方面,均有提高;并且在完成耗时方面,相同人力情况下,节省时间约 20%,加快了验证进度。

$$\eta = \frac{a \text{ 人} \times b \text{ 小时} / \text{天} \times \text{效率} \times c \text{ 天}}{d \text{ 千行}} \quad (1)$$

表 2 芯片及团队规模

代码量/万行	完成时间/d	团队成员数量/人
40	180	16

表 3 UVM 环境平台耗时记录

测试模块	基础测试 覆盖率/ %	完成 耗时/ h	随机测试 覆盖率/ %	完成 耗时/ h	总覆 盖率/ %
UART	86.16	15d15	64.08	12d11	100.00
EMAC	84.24	18d23	59.24	14d09	100.00
GPIO	79.38	23d07	57.60	18d01	100.00
WDT	80.98	18d21	74.03	13d04	100.00
FMC	86.81	25d13	67.75	18d00	100.00
DDRC	86.42	27d15	72.73	20d06	100.00
ROM	82.76	24d08	71.03	20d23	100.00
SUB RST& CLK	90.09	22d14	75.37	19d01	100.00
DMA	84.52	31d05	75.21	23d09	100.00
JPEG	81.94	34d14	63.74	28d20	100.00
VGA	83.67	31d23	62.52	27d21	100.00
PCIe ep	85.06	33d19	69.64	26d22	100.00
ALL CHIP	83.84	65d21	71.68	53d08	100.00

表 4 设计平台耗时记录

测试模块	基础测试 覆盖率/ %	完成 耗时/ h	随机测试 覆盖率/ %	完成 耗时/ h	总覆 盖率/ %
UART	93.12	12d13	81.42	10d03	100.00
EMAC	92.43	15d23	73.75	12d04	100.00
GPIO	89.74	19d11	79.43	15d15	100.00
WDT	94.02	14d10	82.04	11d21	100.00
FMC	96.46	19d03	77.21	16d17	100.00
DDRC	91.68	20d09	82.20	18d23	100.00
ROM	92.17	18d13	78.91	17d17	100.00
SUB RST& CLK	98.55	16d10	83.46	15d19	100.00
DMA	93.00	25d05	80.90	21d21	100.00
JPEG	90.74	29d08	74.49	23d19	100.00
VGA	88.11	27d14	75.25	22d23	100.00
PCIe ep	91.73	28d05	82.63	21d21	100.00
ALL CHIP	93.37	53d21	81.08	42d11	100.00

7 结 论

本文设计了一种基于 SystemVerilog 的视频采集压缩卡验证平台。相较于传统验证平台,它实现了自设计验证组件的高度模块化,平台可移植至同一接口协议的不同类 SoC 设计,完成快速移植,快速布置;实现了部分组件硬件化,在需要极多验证时间的系统验证阶段,可同时在 FPGA 完成仿真验证。该平台采用自设计自动化验证流程,提高了平台验证效率,在验证回归阶段,有助于覆盖代码的边角情况。实际检验表明,该验证平台可提高验证可靠性,加速回归收敛,节省资源,缩短验证周期,增加流片成功率。

参考文献

- [1] CHEN W, RAY S, BHADRA J, et al. Challenges and trends in modern SoC design verification [J]. IEEE Design & Test, 2017, 34(5): 7-12.
- [2] 赵陆,文建平,莫为,等. 图像处理算法 IP 核的异构验证框架[J]. 液晶与显示, 2021, 36(7): 1042-1050.
- [3] 李厚博. 基于 SystemVerilog 的图像处理单元模块验证的研究[D]. 西安:西安电子科技大学, 2018.
- [4] 吕品. 基于 UVM 的显示控制输入模块验证平台的设计与实现[D]. 西安:西安电子科技大学, 2016.
- [5] 罗军. 图像处理快速算法研究与硬件化[D]. 武汉:武汉大学, 2014.
- [6] 李艳龙,杨琪,王雪峰. 基于 SV-DPI 的图像坏元修正 FPGA 自动化验证[J]. 红外技术, 2020, 42(12): 1192-1197.
- [7] 吴乐明,刘浩,况奇刚,等. 归类精度保持的图像测试集

- 压缩方法[J]. 电子测量与仪器学报, 2018, 32(10): 154-160.
- [8] 李鑫. 基于 UVM 的片段处理单元验证平台与随机化验证方法[D]. 西安: 西安电子科技大学, 2016.
- [9] 王鹏, 刘万和, 刘锐, 等. 基于 SystemVerilog 可重用测试平台的实现[J]. 电子技术应用, 2015, 41(2): 61-64.
- [10] 周奇, 宣学雷, 贺光辉. 应用于 FPGA 的 PCIe 接口设计与验证[J]. 微电子学与计算机, 2019, 36(7): 17-21.
- [11] 何雯, 董威, 苟辉. 基于 FPGA 的高速视频采集系统的设计[J]. 国外电子测量技术, 2016, 35(5): 83-87.
- [12] 李雨. 基于 UVM 的 SMBus 总线接口的验证[D]. 西安: 西安电子科技大学, 2019.
- [13] 乔鹏丽, 吕英杰. 基于 SystemVerilog 的数字基带通信芯片的验证[J]. 南开大学学报(自然科学版), 2020, 53(6): 1-4.
- [14] 钱一文, 景为平, 蒋斌. 基于 UVM 的 CPU 卡芯片验证平台[J]. 微电子学与计算机, 2016, 33(6): 37-40.
- [15] 朱伟杰, 阳徽, 费亚男, 等. FPGA 功能验证自动化技术研究与实践[J]. 航天控制, 2017, 35(2): 72-77, 98.

作者简介

王凯, 本科, 主要研究方向为集成电路设计、IC 验证技术。

E-mail: 745579593@qq.com

王骞, 硕士, 主要研究方向为集成电路设计、IC 验证技术。

E-mail: wang.qianlc@inspur.com

符云越, 本科, 主要研究方向为集成电路设计、IC 验证技术。

E-mail: fuyunyuc@inspur.com

李拓, 硕士, 主要研究方向为集成电路设计、IC 验证技术。

E-mail: lituo@inspur.com

刘凯, 硕士, 主要研究方向为集成电路设计、IC 验证技术。

E-mail: liukaibj@inspur.com