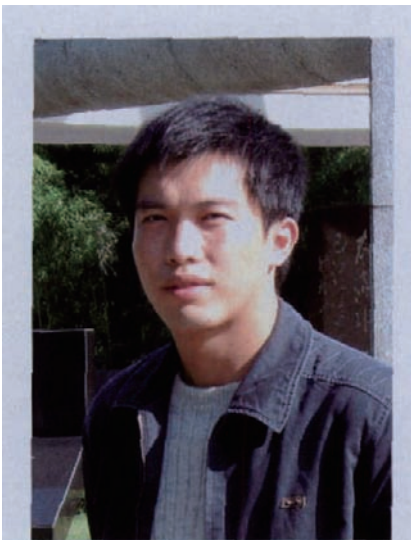


# VCSmx在大规模FPGA设计测试中的应用

Application of VCSmx in Large-Scale FPGA Design Testing

上海旋极信息技术有限公司 马 翔



马 翔

毕业于华东理工大学信息与工程学院, 现任上海旋极信息技术有限公司技术部工程师, 研究 FPGA 前端设计开发和测试, 主要负责 FPGA 软件测试工具和 SpaceWire 总线测试工具等工作。

随着 ASIC 设计的制造成本和复杂性的提高, IC 行业形成了成熟、先进、不断成长的 ASIC 设计方法和开发流程。FPGA 是我国航空现阶段专用集成电路的最佳实现路径, FPGA 的开发流程和设计流程与 ASIC 类似, 设计成果可以为将来 ASIC 开发重用<sup>[1]</sup>。

为了满足各种复杂的大规模 FPGA 设计的测试任务的需要, 同时保证测试质量, 提高测试效率, 必须拥有专业的测试技术和流程。软件仿真可调性强, 借助高性能高容量的测试软件和测试服务器, 可以有效地提高测试效率, 能够测试目前乃至将来一段时间大规模复杂 FPGA 设计, 保证了设计质量和开发的成功率。

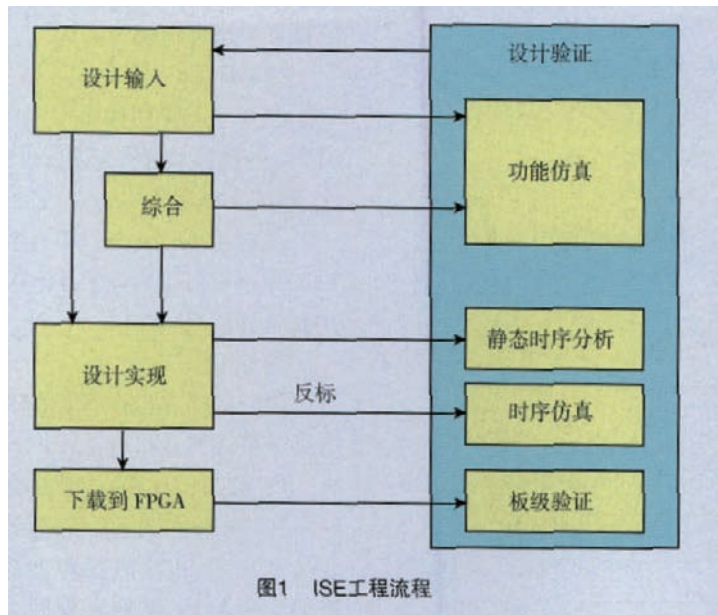
FPGA 设计正在向更复杂和更大规模的方向发展, 开发风险必然随之增加。在 FPGA 开发的各个阶段, 软件仿真测试能够较早地介入设计周期, 及时发现设计中存在的问题, 可以有效缩短设计周期, 提高设计效率。目前, FPGA 主要厂商都针对自己的 FPGA 芯片推出了集成开发环境。但对于专注 FPGA 软件测评的单位而言, 专业的第三方测试软件能够兼容各厂 FPGA 芯片, 有利于形成测试 FPGA 设计的统一标准。

## 专业测试 FPGA 设计的必要性

目前, FPGA 开发人员大多在 FPGA 厂商 IDE 上完成整个开发流程。厂商 IDE 具有容易上手、流程清晰、集成度高和比较可靠的特点。

图 1 是 ISE 的工程流程, 具体分为 5 个步骤: 输入、综合、实现、验证和下载。其中, 验证存在于整个设计流程, 静态时序分析目的是发现使芯片时序失效和对芯片性能起决定作用的电路关键路径<sup>[2]</sup>; 功能仿真和时序仿真测试设计的正确性, 考虑了布线延时的时序仿真在线宽为深亚微米尺寸和频率较高的设计中尤为重要。时序仿真和静态时序分析的结果是“所见即所得”的, 因此板级验证时可以不再关心 FPGA 内部信号。

在大规模 FPGA 开发过程中, 验证工作量随设计的速度、规模和复杂度的提高而增加, 验证在 FPGA 设计中的地位也越来越重要。图 2 是来自 Synopsys 报告, 2001 年验证的工作量是设计的 3 倍, 其预计、设计规模每扩大 10 倍, 仿真向量就要增加



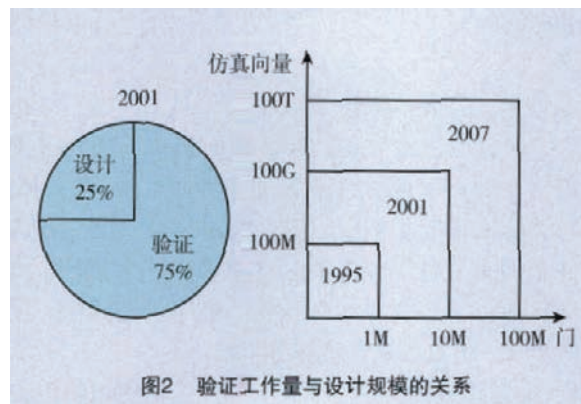
1000倍<sup>[3]</sup>。图3是Synopsys 2004年用户调查的数据,可以看到ASIC设计失败大多是由功能错误导致的<sup>[4]</sup>。由此可见,测试在ASIC和大规模FPGA开发中的作用是很重要的。

设计失败在测试方面的原因主要有测试不全面和欠缺系统测试。在开发流程中,开发人员常常也是测试人员,必然存在一定的思维定势和考虑不周全,容易漏测特殊条件、边界情况、临界状态等状况,而没有测试到的状况有可能是具有潜伏性和破坏性的故障。

目前厂商IDE较适合做单元测试和集成测试,对大规模FPGA设计的测试支持不足,设计规模和仿真时间都有限,仿真效率也不高,等待时间较长。由于FPGA可编程性强,因此许多系统测试直接在FPGA板上进行。但板上测试一般不够全面,出现问题也很难定位错误。另外,设计风格和代码规范的不规范和不统一也是一个重要的因素,很大程度上影响了设计的可读性、可测性和可重用性,导致大量重复劳动和设计风险。

开发和测试不同厂商的FPGA需要安装和熟悉不同的IDE,由于可

以得到厂商提供的单元库和技术库,因此可以选用先进的ASIC工具测试大规模FPGA设计,如Synopsys公司的VCSmx仿真软件可以作为通用FPGA测试工具使用。



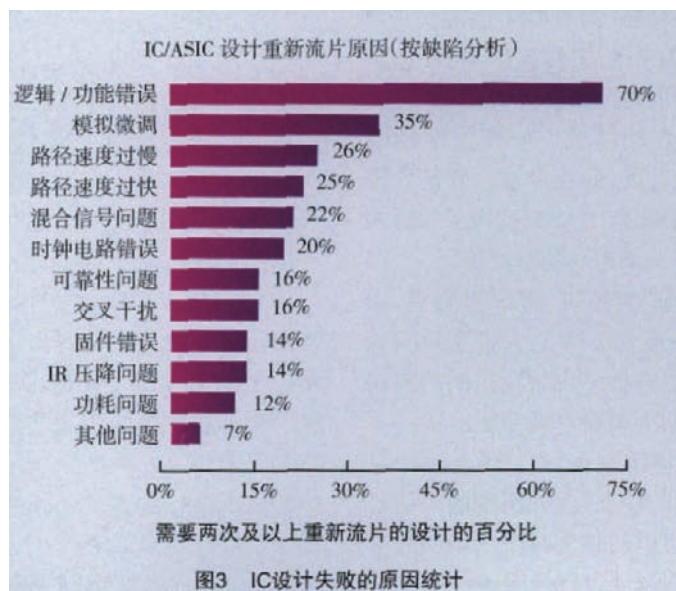
## 使用 VCSmx 对 FPGA 做测试

### 1 FPGA 测试流程

软件仿真测试FPGA设计的基本流程如图4所示。“检查文档”和“总结报告”是测试项目交接、交付的必要内容,保证设计文档和测试报告的规范性、完整性和一致性。仿真前的“代码规则检查”和“人工走查”很有必要,代码规范检查工具leda自动检查设计的代码编写风格、代码可综合性、时钟和复位规则等。leda还可以检查verilog设计文件的VCS可仿真性。

运行仿真前必须搭建仿真环境,根据测试目标编写测试平台和测试用例。VCSmx支持混合语言仿真,仿真文件和测试文件可以选择VHDL和Verilog的一种语言描述。图5是一个待测设计的框图,它在测试平台下实例化。待测设计是一个AHB从设备,控制IO、SRAM控制和UART收发是其中的一些模块。该设计已经通过了板级功能验证,但系统联调时间断性地出现错误。由于包含的模块较多,所以需要进一步收集出错情况。

测试平台模拟了与被测设计(DUT)对接的AHB bus、UART接口



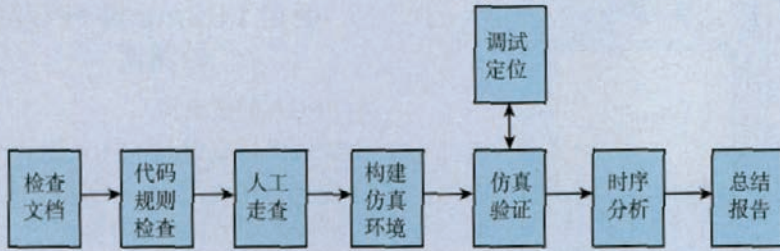


图4 软件仿真测试FPGA基本流程

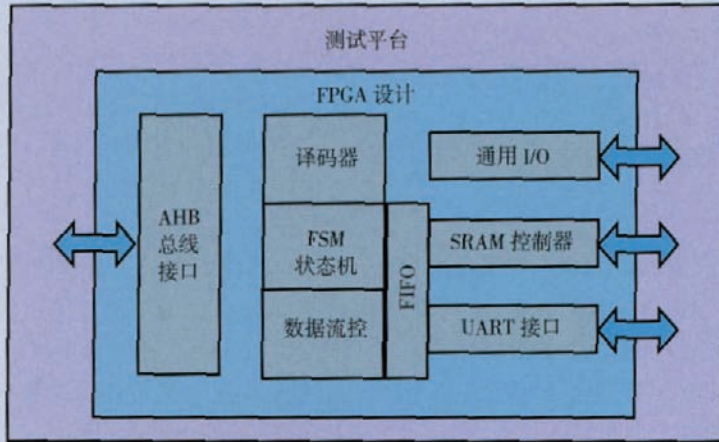


图5 待测设计和测试平台

和一片 SRAM,产生时钟信号和激励信号,监视 DUT 的输入和输出变化。获得数据后与预期结果比较,调用系统函数输出,或写到文件中保存。

测试用例包括激励的所有可能情况,测试平台依据设计要求检查 DUT 的输出。测试内容包括:

(1) 功能测试。检查正常激励下执行的功能,如 UART 正常收发数据、SRAM 控制信号波形和 IO 口是否在正确的时间翻转等。

(2) 性能测试。检查正常激励下的量化性能,如指令输入到信号输出之间的延时、UART 波特率的精确度和执行一条指令的时间等。

(3) 临界测试。测试极限值,如测试两条指令之间至少需要间隔的时间,可以选择一个较宽裕的间隔初值,逐渐缩小间隔直到功能出错。

(4) 容错性测试。检查设计对错误激励的容忍度,如出现输入信号有毛刺、错误的总线时序和错误的指令时,被测设计运行情况。

## 2 使用 VCSmx 仿真

仿真测试在 linux 环境下进行,项目文件夹下的文件部署如图 6 所示。post 下存放用于后仿真的 sdf 文件和网表文件。Makefile 文件中定义各种命令,它简化了每次键入命令的步骤。仿真的基本流程分 3 步:分析、编译和仿真<sup>[5]</sup>。

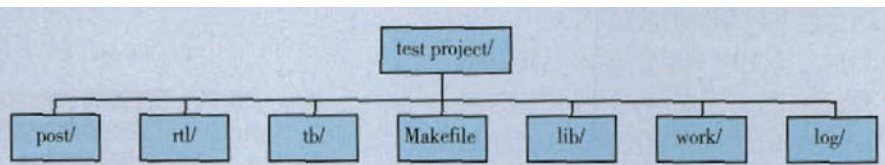


图6 测试项目的文件部署

(1) 分析。

先分析 Verilog 文件,再根据 VHDL 的依赖关系自底向上按顺序分析 VHDL 文件,分析命令如下,命令执行后产生的中间文件保存在 work 库中。

```
% vlogan ./lib/*.v ./post/*.v ./rtl/*.v
% vhdlan ./lib/*.vhd ./rtl/f1.vhd ./
```

rtl/f2.vhd ./rtl/f3.vhd

(2) 编译。

命令 vcs 使用中间文件生成目标码,再链接得到二进制可执行文件,默认文件名为 simv。

```
vcs -debug_all -v ${LIB_VLOG}+vhdllib+${LIB_VHDL}
${MODULE_NAME}
```

(3) 仿真。

执行文件 ./simv 开始仿真,直到仿真结束或终止进程。

仿真时, VCSmx 实时地显示 \$display 和 \$monitor 系统函数等输出的仿真日志,也实时保存波形文件。需要注意的是,波形文件的大小、信号数量和仿真时间有关。长时间仿真产生的波形文件可能占用分配的所有可用空间,导致仿真无法继续。因此,为了保证服务器正常运行,必须有限制地产生波形文件。

(4) 使用 sdf 文件做时序仿真。

由于 FPGA 的特点,布局布线后产生下载文件时,能产生标注延时信息的 sdf 文件。目前主要 FPGA 厂商的 IDE 都能生成 sdf 文件。时序仿真更多地关心时序问题,如信号出现毛刺等。时序仿真的测试环境与功能仿真一样,编译命令:

```
vcs -v ${LIB_VLOG}
${MODULE_NAME} -sdf min: ${SDF_
```

MODULE}:\${SDF\_FILE}

其中, LIB\_VLOG 表示所使用的 FPGA 的库文件, -sdf min: 表示使用 sdf 文件中的最小工况,一般产生 sdf 文件内含有最大、典型和最小 3 种工况下的不同延时。

## 3 使用 VCSmx 调试

当仿真结果与预期结果有出入时,就需要对设计跟踪调试,逐步缩

小范围最终定位错误,才能正确提交问题单。如在这次测试中,测试平台向 AHB 接口发命令要求被测设计向输出口 cmdout[7:0] 中的一个位发送一个 160ms 脉宽的脉冲。在临界测试中,命令每隔 1s 发送一次,每次缩短间隔 1ms,新命令到来后执行新脉冲,但继承原脉冲的计数器。

通过 force/release 跟踪排查到一个间接控制 cmdout 输出的模块,该模块如图 7 所示。设计者在 160ms 到达时, qc=1, qc 与全局清零信号 reset 取或后得到 clr 信号。命令信号和 clr 同时有效时,优先执行 clr。因此命令和计时器同时到达时,命令将被忽略。

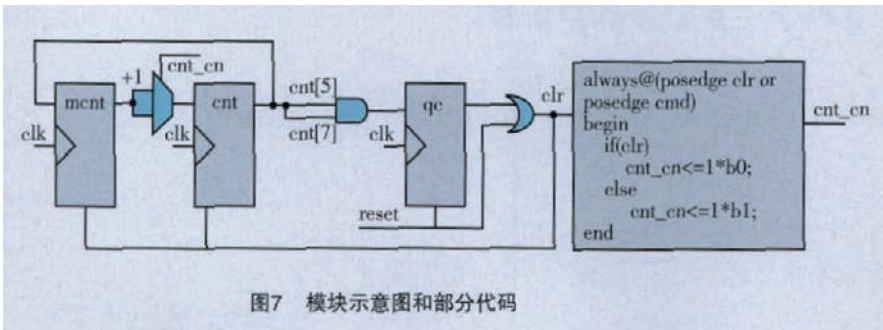


图7 模块示意图和部分代码

在 DVE 中新建 Wave 窗口,先把 AHB 接口到 cmdout 输出相关的信号加入窗口,在 DVE 下启动仿真。为了在 DVE 下重现故障场景,force 间隔变量到 161ms,再全速运行 160ms,即前一个 160ms 脉冲刚结束。

#### 4 使用 VCSmx 做代码覆盖率分析

覆盖率一般表示一个设计的验证进行到什么程度,也是决定功能验证是否完成的重要量化标准之一。常用的覆盖率有行覆盖率、条件覆盖率和 FSM 覆盖率。做覆盖率分析目

的是确保测试用例对设计的测试足够全面。设计运行时所有可能的状态都应该被测试到,不执行的代码可能是未来的故障,不能定位的内容可能影响代码可读性,应该考虑它们保留的必要性。虽然覆盖率的高低是衡量测试的一个标志,但没有必要追求 100% 的覆盖率。覆盖率分析后生成文本文件,作为总结报告依据或提交的附件。

## 结束语

为了满足各种复杂的大规模 FPGA 设计的测试任务的需要,同时保证测试质量,提高测试效率,必须拥有专业的测试技术和流程。软件仿真可调性强,借助高性能高容量的测试软件和测试服务器,可以有效地提高测试效率,并且能够测试目前乃至将来一段时间大规模复杂 FPGA 设计,保证了设计质量和开发的成功率。

(责编 岩石)





手柄类



水平调整件



合金拉手



手轮



把手



铰链

唯有更专业

才有高品质



压紧把手



拉紧把手



顶紧把手



**扬州芬尔机械配件有限公司**  
YANGZHOU FAIERR MECHANICAL FITTING CO.,LTD.

地址: 江苏省扬州市沙头镇施沙路8号 邮编: 225105  
电话: 0514-87533188 87533288 传真: 0514-87533288 87533088  
http: //www.faierr.com E-mail: sale@faierr.com.cn

广告索引号10-077