



5.1 Testbench 设计

5.1.1 Testbench 简介

Testbench 是一种验证工具。首先,大部分设计都需要输入输出。但是在软环境中没有激励输入,也不会对设计的输出正确性进行评估。那么需要有模拟实际环境的输入激励和输出校验的一种“虚拟平台”出现了。在这个平台上可以对设计从软件层面上进行分析和校验,这个就是 Testbench 的含义。

Testbench 包含两部分。

(1) 激励生成。这部分只用来生成输出而自己没有输入。生成的激励信号通过用户的设计输入端口进行互连。这里的激励,都是预先设想好的,比如根据某个协议或者某种通信方式传递。

(2) 输出校验。即接收设计的输入,然后通过校验,找出对应的问题。通俗地讲,就是利用 Testbench 把自己解脱出来,让软件来帮助自己找错误,并以打印、通知等方式来了解设计的正确性。

设计与验证框图如图 5-1 所示。

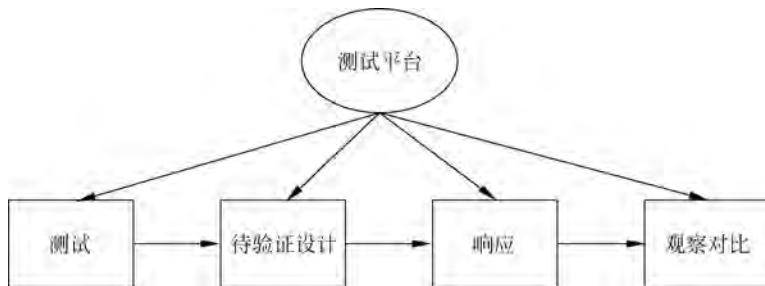


图 5-1 设计与验证

如图 5-2 所示,测试结果不仅可以通过观察、对比波形来验证,而且可以灵活地使用脚

本命令将有用的输出信息打印到终端或者产生文本进行观察,也可以写一段代码自动比较输出结果。总之,Testbench 的设计是多种多样的,它的语法也是很随意的,不像 RTL 级设计代码那么严格,很多高级的语法都可以在脚本中使用。因为它不需要实现到硬件中,是运行在 PC 上的一段脚本。但是,使用 Verilog 的验证脚本也有很多需要设计者留意的地方,它是一种基于硬件语言又服务于软件测试的语言,不过,只要掌握好了 Verilog 语言的关键点,是可以更好地利用它来满足设计验证。

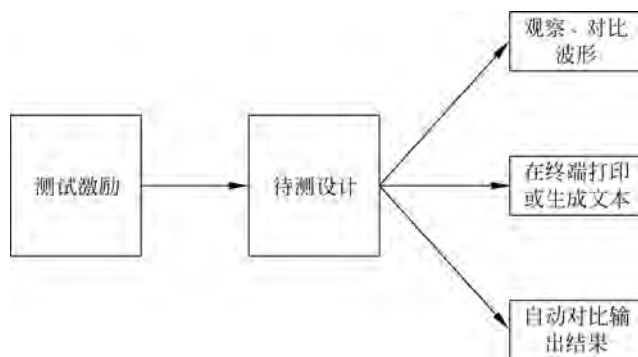


图 5-2 验证输出

5.1.2 Testbench 的搭建

Verilog 硬件描述语言在数字电路的设计中使用非常普遍,无论是哪种语言,仿真都是必不可少的。随着设计复杂度的提高,仿真工具的重要性也越来越凸显。在一些小的设计中,用 Testbench 来进行仿真是一个不错的选择。VHDL 与 Verilog 语言的语法规则不同,它们的 Testbench 的具体写法也不同,但是基本结构大体相似。在 VHDL 的仿真文件中应包含以下几点:实体和结构体声明、信号声明、顶层设计实例化、提供激励;Verilog 的仿真文件应包括模块声明、信号声明、顶层设计实例化、提供激励。Verilog 在设计中使用更普遍,这里以 Verilog 的仿真模型为例进行介绍,如图 5-3 所示。

```

module test_bench;
    // 端口声明语句
    // 输入reg, 输出wire
    initial
    begin
        // 产生时钟信号
    end
    initial
    begin
        // 提供激励源
    end
    // 例化语句, 例化测试块
endmodule
  
```

图 5-3 Verilog 的仿真模型

5.2 ModelSim 介绍及仿真

5.2.1 ModelSim 简介

Mentor 公司的 ModelSim 软件是业界最优秀的 HDL 语言仿真软件之一。它提供个性化图形界面和用户接口,编译仿真速度快,而且所编译的代码与平台无关,是 FPGA/ASIC 设计的首选仿真软件。

仿真的主要目的是验证功能是否与设想的一致。仿真分为功能仿真和时序仿真,功能仿真是不带芯片时间延迟的仿真方法,主要用来验证功能;时序仿真加入了时间延迟,可以考查在一定条件下功能是否符合设想。

5.2.2 ModelSim 仿真

如图 5-4 所示,打开 ModelSim 软件,新建一个 Library。

如图 5-5 所示,给新建的 Library 命名。

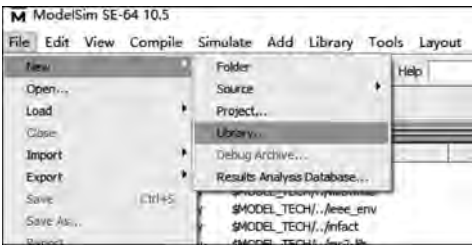


图 5-4 File→New→Library 菜单命令



图 5-5 给 Library 命名

如图 5-6 和图 5-7 所示,新建一个工程,并给工程命名。

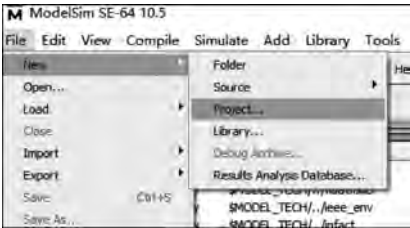


图 5-6 File→New→Project 菜单命令

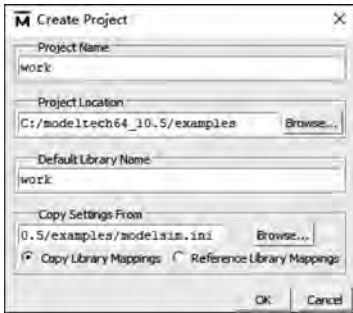


图 5-7 给工程命名

在如图 5-8 所示的界面,单击 Close 按钮。
如图 5-9 所示,新建一个 Verilog 文件。
如图 5-10 所示,进入主程序,下面以全加器为例介绍。编辑完成后,单击保存按钮。文件名要与 module 后面的名称相同,文件扩展名改为.v。

如图 5-11 所示,再新建一个测试文件,步骤同上面新建的主程序文件,文件扩展名改为.vt。

如图 5-12 所示,添加文件,再编译文件。先右击左边



图 5-8 Add items to the Project 界面

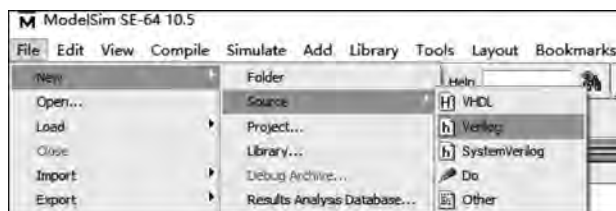


图 5-9 File→New→Source→Verilog 菜单命令

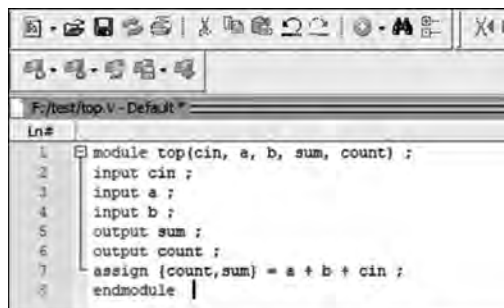


图 5-10 编写程序窗口

空白处,选择 Add to Project→Existing File 命令。

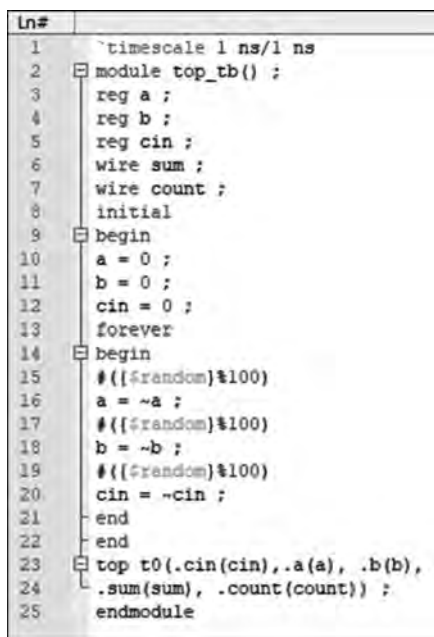


图 5-11 测试程序窗口

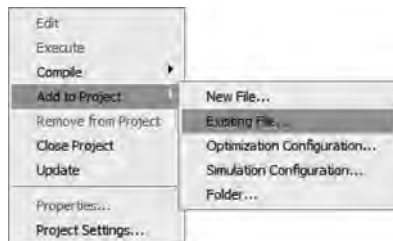


图 5-12 添加文件

如图 5-13 所示,选择刚刚新建的两个文件。按 Ctrl 键可以同时选择两个文件,单击“打开”按钮。



图 5-13 文件选择界面

如图 5-14 所示,选择菜单命令 Compile→Compile All,若出现两个“√”,则说明编译通过;若出现“×”,则说明文件编译出错,双击“×”,可以查看错误。

如图 5-15 所示,选择菜单命令 Simulate→Start Simulation,开始仿真。

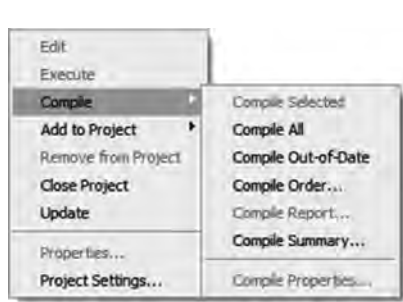


图 5-14 编译

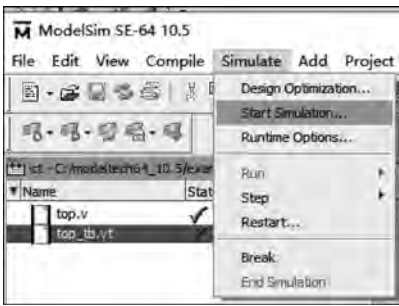


图 5-15 开始仿真

如图 5-16 所示,选择 work 库中的 top_tb,或者测试文件名称,一定不要选中左下角的 Enable optimization 复选框,否则不会出现波形。

如图 5-21 所示,由于前面没有创建文件夹,所以在这里会弹出询问是否创建文件夹的提示,单击 Yes 按钮,开始进行编译,编译成功界面如图 5-22 所示。至此,PDS 软件与 ModelSim 就可以进行联合仿真了。

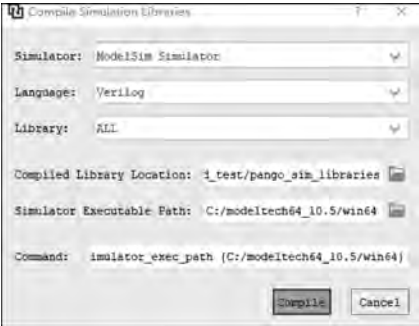


图 5-20 编译库设置

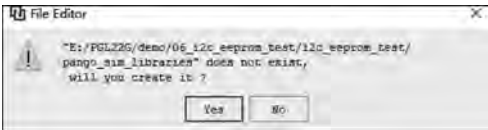


图 5-21 创建文件夹



图 5-22 编译完成