

工程设计与分析系列

ModelSim (第2版)

电子系统分析及仿真

于 斌 谢龙汉 编著

视频教学



- ★ ModelSim——全球使用最广泛的EDA仿真软件之一
- ★ ModelSim——支持Lattice、Altera和Xilinx器件仿真
- ★ 基础知识—实训实例—工程实例
- ★ 实例操作视频教学，轻松学习



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

工程设计与分析系列

ModelSim 电子系统分析及仿真

(第2版)

于 斌 谢龙汉 编著

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

ModelSim 是优秀的 HDL 仿真软件之一, 它能提供友好的仿真环境, 是业界唯一单内核支持 VHDL 和 Verilog 混合仿真的仿真器, 它采用直接优化的编译技术、Tcl/Tk 技术和单一内核仿真技术, 编译仿真速度快, 编译的代码与平台无关, 便于保护 IP 核, 个性化的图形界面和用户接口, 为用户加快调错提供强有力的手段, 是 FPGA/ASIC 设计的首选仿真软件。

本书以 ModelSim SE 10.1c 版软件为平台, 由浅入深、循序渐进地介绍 ModelSim 10.1c 软件各部分知识, 包括 ModelSim 10.1c 的基础知识、菜单命令、库和工程的建立与管理、Verilog/VHDL 文件编译仿真、采用多种方式分析仿真结果, 以及与多种软件联合仿真等知识。书中配有大量插图, 并结合实例详细地讲解使用 ModelSim 进行仿真操作的基本知识和方法技巧, 配书光盘中有本书实例操作的视频讲解, 读者能够轻松学习。

本书适合具有一定 HDL 基础的读者使用, 同时对相关领域的专业技术人员也有较高的参考价值, 也可作为大中专院校电子类相关专业和培训班的教材。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有, 侵权必究。

图书在版编目 (CIP) 数据

ModelSim 电子系统分析及仿真/于斌, 谢龙汉编著. —2 版. —北京: 电子工业出版社, 2014.2
(工程设计与分析系列)

ISBN 978-7-121-21920-7

I. ①M… II. ①于… ②谢… III. ①电子电路—计算机辅助设计—应用软件 IV. ①TN702

中国版本图书馆 CIP 数据核字 (2013) 第 276089 号

策划编辑: 许存权

责任编辑: 许存权 特约编辑: 马军令

印 刷: 北京京师印务有限公司

装 订: 北京京师印务有限公司

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1 092 1/16 印张: 24.25 字数: 610 千字

印 次: 2014 年 2 月第 1 次印刷

印 数: 3 500 册 定价: 59.00 元 (含光盘 1 张)

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zlt@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。



再版前言

ModelSim 是 Mentor Graphics 公司开发的 EDA 工具软件，是一款主要应用于 HDL 仿真的软件，为调试设计提供了强力的支持。

ModelSim 是一款强大的仿真软件，不仅支持对 HDL 的仿真，还支持 SystemC、C 语言等的调试和仿真，使得在整个的设计中可以采用更灵活的手段来完成设计功能。

在仿真的过程中，ModelSim 可以独立完成 HDL 代码的仿真，还可以结合 FPGA 开发软件对设计单元进行时序仿真，得到更加真实的仿真结果。多数的 FPGA 厂商都提供了与 ModelSim 的接口，使得设计者在器件的选择和结果的掌握上更加得心应手。

本书第 1 版在 2012 年出版以来，获得读者的广泛好评，已多次重印。并且，很多读者来信介绍他们具体应用 ModelSim 的情况，对本书提出了很多宝贵意见和建议。在此基础上，我们根据用户的建议、结合相关企业应用的需求和高校教学需求，再根据作者多年的教学和工作经验，进行了修订。第 2 版是在最新软件版本 ModelSim 10.1c 的基础上写作的，更新了大量内容，并且也更加贴合实际应用，相信可以更好地帮助读者深入应用 ModelSim。

本书在编写过程中，突出了以下特点：

1. 直观易懂性

全书以实例图解的形式介绍基础知识和实例操作，所有知识点和操作流程尽可能给出配套图片，直观易懂，使用户能够在最短的时间内获取最多的知识。

2. 可扩展性

全书以 ModelSim SE 10.1c 版软件为平台进行讲解，但是讲解过程中提供了命令行操作和菜单操作两种操作方法，命令行操作使得讲解的知识更具扩展性。


3. 实用性

全书采用了基础知识介绍和实例操作相结合的方法，互相补充，书中的实例都是具有实际意义的设计实例，并根据介绍内容的不同进行了选取，使读者能够更好地理解操作的过程，使读者在学完本书后能够快速地将知识应用于生产实践。

4. 结构清晰，讲解详尽

全书采用基础知识、综合实例的循序渐进的讲解方法，一步步地提高用户的仿真技能，而且每个知识点和实例都做了尽可能详细地讲解，使用户学习起来轻松自如。

5. 多媒体示范

本书的配套光盘中提供了所有实例的视频操作  **动画演示**，读者可以在观看视频时增强

对知识点的理解。同时，视频中操作的步骤严格按照书中实例的步骤进行，可以看到文字难以描述的过程。

本书分为 8 章，依次介绍 ModelSim 10.1c 基本知识、操作界面、工程和库、对不同语言的仿真、仿真分析方法、与其他软件的协同仿真、对不同公司器件的后仿真、ModelSim 的文件和脚本等内容。

第 1 章 概述。介绍 IC 设计的基本流程和 ModelSim 不同版本的特点和功能，并给出一个简单的实例，快速地掌握使用 ModelSim 进行基本仿真的流程。

第 2 章 操作界面。介绍 ModelSim 的基本操作界面，包括菜单栏中各命令的基本功能和主界面中工作区、命令区、MDI 区的功能，并介绍仿真中经常使用到的窗口。

第 3 章 工程和库。介绍工程和库的相关知识，给出详细的工程管理方法和库的建立导入方式，并给出了实例。

第 4 章 ModelSim 对不同语言的仿真。介绍使用 ModelSim 对 Verilog 语言和 VHDL 的仿真方法，并分别配以实例进行讲解。给出了对 SystemC 的仿真方法和三种语言混合仿真需要注意的事项，并给出了与 Verilog 混合仿真的实例。

第 5 章 利用 ModelSim 进行仿真分析。介绍如何使用 ModelSim 观察仿真结果，进行仿真分析，主要包括 WLF 文件、创建波形激励、波形分析、存储器查看、数据流窗口、原理图窗口、性能分析、信号探测和利用 JobSpy 进行批处理使用等内容，在章末配有多个实例演示这些功能。

第 6 章 ModelSim 的协同仿真。介绍如何使用其他软件工具与 ModelSim 进行系统仿真，弥补 ModelSim 的不足，主要介绍了使用 Debussy 和 Matlab 与 ModelSim 进行仿真的配置方法和步骤，并给出了实例。

第 7 章 ModelSim 对不同公司器件的后仿真。介绍利用 FPGA 开发工具与 ModelSim 联合进行后仿真的过程，以 Altera、Xilinx、Lattice、Actel 四大业界主流厂商的开发工具为例，并结合实例演示。

第 8 章 ModelSim 的文件和脚本。介绍前 7 章中涉及的文件类型，包括 SDF 文件、VCD 文件、Tcl 文件、DO 文件等，这些文件都是在仿真中有重要作用的文件类型，在本章中统一进行讲解，并配以实例演示。

全书主要由哈尔滨理工大学于斌编写，参与本书编写和光盘开发的人员还有谢龙汉、林伟、魏艳光、林木议、王悦阳、林伟洁、林树财、郑晓、吴苗、李翔、莫衍、朱小远、唐培培、耿煜、尚涛、邓奕、张桂东、鲁力、刘文超、刘新东、米秀杰等。由于时间仓促，书中难免有疏漏之处，请读者谅解。如有任何意见和建议，读者可通过电子邮件 yubin@hrbust.edu.cn 与我们交流。

编者



目 录

第1章 概述	1	2.3 工具栏	42
1.1 IC设计与ModelSim	2	2.4 标签区	42
1.1.1 IC设计基本流程	2	2.5 命令窗口	43
1.1.2 ModelSim概述	3	2.6 MDI窗口	44
1.2 ModelSim应用基本流程	5	2.6.1 源文件窗口	44
1.3 ModelSim基本仿真流程	5	2.6.2 波形窗口	45
1.3.1 创建一个工作库	6	2.6.3 列表窗口	46
1.3.2 编译设计文件	7	2.6.4 数据流窗口	46
1.3.3 运行仿真	8	2.6.5 属性窗口	47
1.3.4 查看结果	9	2.6.6 进程窗口	48
1.4 ModelSim工程仿真流程	9	2.6.7 对象窗口	48
1.4.1 创建工程及工程库	10	2.6.8 存储器窗口	48
1.4.2 创建新文件	11	2.6.9 原理图窗口	49
1.4.3 加载设计文件	12	2.6.10 观察窗口	50
1.4.4 编译源文件	13	2.7 界面的设置	50
1.4.5 运行仿真和查看结果	14	2.7.1 定制用户界面	50
1.4.6 工程调试	14	2.7.2 设置界面参数	52
第2章 操作界面	16	第3章 工程和库	54
2.1 整体界面	17	3.1 ModelSim工程	55
2.2 菜单栏	17	3.1.1 删除原有工程	55
2.2.1 File菜单	18	3.1.2 开始一个新工程	55
2.2.2 Edit菜单	25	3.1.3 工程标签	57
2.2.3 View菜单	27	3.1.4 工程编译	58
2.2.4 Compile菜单	28	3.1.5 仿真环境配置	61
2.2.5 Simulate菜单	30	3.1.6 工程文件组织	63
2.2.6 Add菜单	33	3.1.7 工程及文件属性设置	64
2.2.7 Tools菜单	33	实例3-1 工程文件管理	69
2.2.8 Layout菜单	39	3.2 ModelSim库	73
2.2.9 Bookmarks菜单	39	3.2.1 概述	73
2.2.10 Window菜单	40	3.2.2 库的创建及管理	74
2.2.11 Help菜单	41	3.2.3 资源库管理	76

3.2.4 导入FPGA的库.....	77	过程.....	145
3.2.5 本节实例.....	78		
第4章 ModelSim 对不同语言的仿真	84	第5章 利用 ModelSim 进行仿真分析	148
4.1 VHDL 仿真.....	85	5.1 仿真概述.....	149
4.1.1 VHDL 文件编译.....	85	5.2 WLF 文件和虚拟对象.....	150
4.1.2 VHDL 设计优化.....	86	5.2.1 保存仿真状态.....	150
4.1.3 VHDL 设计仿真.....	91	5.2.2 Dataset 结构.....	153
4.1.4 还原点和仿真恢复.....	96	5.2.3 Dataset 管理.....	154
4.1.5 TEXTIO 的使用.....	97	5.2.4 虚拟对象.....	156
实例 4-1 VHDL 设计的仿真全过程.....	99	5.3 利用波形编辑器产生激励.....	159
4.2 Verilog 仿真.....	104	5.3.1 创建波形.....	159
4.2.1 Verilog 文件编译.....	104	5.3.2 编辑波形.....	165
4.2.2 Verilog 设计优化.....	105	5.3.3 导出激励文件并使用.....	168
4.2.3 Verilog 设计仿真.....	106	5.4 采用描述语言生成激励.....	170
4.2.4 还原点和仿真恢复.....	111	5.5 ModelSim 波形分析.....	175
4.2.5 单元库.....	111	5.5.1 波形窗口和列表窗口.....	175
4.2.6 系统任务和系统函数.....	112	5.5.2 时间标记.....	178
4.2.7 编译指令.....	114	5.5.3 窗口的缩放.....	178
实例 4-2 32 位浮点乘法器的 Verilog 仿真过程.....	115	5.5.4 在窗口中搜索.....	180
4.3 C 调试.....	121	5.5.5 窗口的格式编排.....	181
4.3.1 概述.....	121	5.5.6 波形和列表的保存.....	184
4.3.2 C 步进调试与调试设置.....	123	5.5.7 信号总线.....	186
4.4 SystemC 仿真.....	124	5.5.8 光标操作.....	186
4.4.1 概述.....	124	5.5.9 其他功能.....	187
4.4.2 SystemC 文件的编译和链接.....	125	5.5.10 波形比较.....	188
4.4.3 设计仿真和调试.....	130	5.6 存储器的查看和操作.....	194
4.4.4 常见错误.....	132	5.6.1 存储器的查看.....	194
4.5 混合语言仿真.....	134	5.6.2 存储数据的导出.....	196
4.5.1 编译过程与公共设计库.....	134	5.6.3 存储器初始化.....	197
4.5.2 映射数据类型.....	136	5.6.4 存储器调试.....	198
4.5.3 VHDL 调用 Verilog.....	139	5.7 数据流窗口的使用.....	199
4.5.4 Verilog 调用 VHDL.....	141	5.7.1 概述.....	199
4.5.5 SystemC 调用 Verilog.....	141	5.7.2 设计连通性分析.....	200
4.5.6 Verilog 调用 SystemC.....	142	5.7.3 信号追踪和查找.....	201
4.5.7 SystemC 调用 VHDL.....	143	5.7.4 设置和保存打印.....	203
4.5.8 VHDL 调用 SystemC.....	144	5.7.5 本节实例.....	204
实例 4-3 systemC 与 Verilog 混合仿真		5.8 原理图窗口的使用.....	208
		5.9 ModelSim 的剖析工具.....	211
		5.9.1 运行性能剖析和存储器剖析.....	211

5.9.2 查看性能剖析结果.....	212	ModelSim 进行时序仿真.....	310
5.9.3 查看存储器剖析报告.....	215	7.2 ModelSim 对 Xilinx 器件的	
5.9.4 保存结果.....	216	后仿真.....	317
5.10 覆盖率检测.....	217	7.2.1 ISE 简介.....	317
5.10.1 启用代码覆盖.....	217	7.2.2 后仿真流程.....	319
5.10.2 覆盖率的查看.....	222	实例 7-3 用 ISE 对全加器进行时序	
5.10.3 覆盖率检测的过滤.....	225	仿真.....	319
5.10.4 覆盖信息报告.....	227	实例 7-4 用 ISE 直接调用 ModelSim	
5.11 信号探测.....	230	进行时序仿真.....	327
5.12 采用 JobSpy 控制批处理仿真.....	232	7.3 ModelSim 对 Lattice 器件的	
5.12.1 JobSpy 功能与流程.....	233	后仿真.....	336
5.12.2 运行 JobSpy.....	233	7.3.1 Diamond 简介.....	336
5.13 综合实例.....	235	7.3.2 后仿真流程.....	337
实例 5-1 三分频时钟的分析.....	235	实例 7-5 用 Diamond 对全加器进行	
实例 5-2 同步 FIFO 的仿真分析.....	242	时序仿真.....	337
实例 5-3 基 2 的 SRT 除法器的仿真		实例 7-6 用 Diamond 完成布局绕线, 使用	
分析.....	248	ModelSim 进行时序仿真.....	343
第 6 章 ModelSim 的协同仿真.....	256	7.4 ModelSim 对 Actel 器件的后仿真	345
6.1 ModelSim 与 Debussy 的		实例 7-7 用 Libero IDE 调用 ModelSim	
协同仿真.....	257	进行时序仿真.....	346
6.1.1 Debussy 工具介绍.....	257	第 8 章 ModelSim 的文件和脚本.....	354
6.1.2 Debussy 配置方法.....	261	8.1 SDF 文件.....	355
实例 6-1 与 Debussy 的协同仿真.....	264	8.1.1 SDF 文件的指定和编译.....	355
6.2 ModelSim 与 Matlab 的协同		8.1.2 VHDL 的 SDF.....	357
仿真.....	271	8.1.3 Verilog 的 SDF.....	357
实例 6-2 与 Matlab 的协同仿真.....	274	8.1.4 SDF 文件信息.....	359
实例 6-3 与 Simulink 的协同仿真.....	278	8.2 VCD 文件.....	361
实例 6-4 使用 cosimWizard 进行		8.2.1 创建一个 VCD 文件.....	361
协同仿真.....	286	8.2.2 使用 VCD 作为激励.....	363
8.2.3 VCD 任务.....	364	8.2.4 端口驱动数据.....	365
第 7 章 ModelSim 对不同公司器件的		8.3 Tcl 和 DO 文件.....	367
后仿真.....	294	8.3.1 Tcl 命令.....	367
7.1 ModelSim 对 Altera 器件的		8.3.2 Tcl 语法.....	367
后仿真.....	295	8.3.3 ModelSim 的 Tcl 时序命令.....	368
7.1.1 Quartus II 简介.....	295	8.3.4 宏命令.....	369
7.1.2 后仿真流程.....	297	8.3.5 本节实例.....	371
实例 7-1 直接采用 Quartus II 调用			
ModelSim 进行仿真.....	298		
实例 7-2 先用 Quartus II 创建工程, 再用			



第1章 概述

IC (Integrated Circuit) 行业是一个具有广阔前景和巨大潜力的行业。随着工艺的发展和设计规模的不断扩大,EDA 软件在 IC 设计过程中扮演着越来越重要的角色。本书所介绍的 ModelSim 就是 EDA 软件的一种。通过本章的介绍,读者可以从整体上了解 IC 设计的流程及 ModelSim 的使用概况,并且可以通过一个简单的例子快速地掌握 ModelSim 的基本使用方法。



本章内容

- IC 设计流程
- ModelSim 的功能和作用
- ModelSim 基本流程



本章案例

- ModelSim 应用基本流程
- ModelSim 工程仿真流程

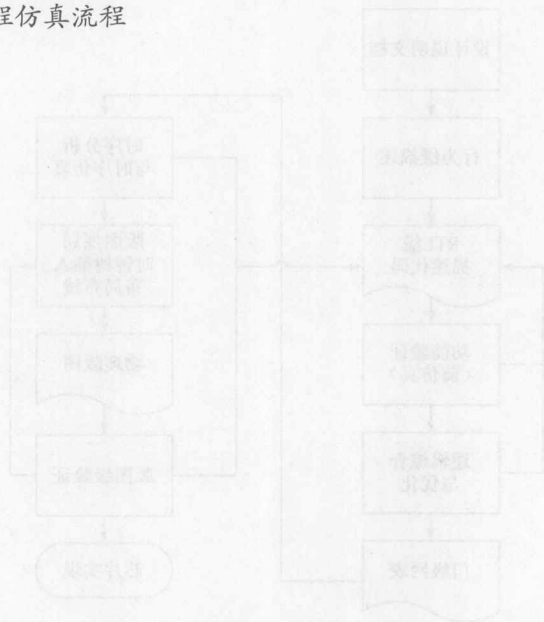


图 1-1 IC 设计基本流程

1.1 IC 设计与 ModelSim

ModelSim 是 Mentor Graphics 公司开发的 EDA 工具软件, 主要针对 IC 设计的仿真阶段, 即对采用 Verilog HDL (硬件描述语言) 或 VHDL (可视硬件描述语言) 描述的设计进行验证。细分整个 IC 设计流程, 这个阶段属于数字 IC 设计的仿真验证部分。

1.1.1 IC 设计基本流程

IC 设计流程包括两大类: 正向设计流程 (Top-Down) 和反向设计流程 (Bottom-Up)。正向设计流程指的是从最顶层的功能设计开始, 根据顶层功能的需要, 细化并完成各个子功能, 直至达到最底层的功能模块为止。反向设计正好相反, 设计者最先得到的是一些底层的功能模块, 采用这些底层的模块搭建出一个高级的功能, 按照这种方式继续直至顶层的设计。

IC 行业的最初阶段, EDA 工具软件功能并不强大, 所以两种方法都被采用。随着 EDA 工具的功能逐渐增强, Top-Down 的设计流程得到了很好的支持并逐步成为主流的 IC 设计方法。这种方法也符合设计者的思维过程: 当拿到一个设计项目时, 设计者首先想到的是整体电路需要达到哪些性能指标, 进而采用高级语言尝试设计的可行性, 再经过 RTL 级、电路级直至物理级逐渐细化设计, 最终完成整个项目。

由于 EDA 厂商的工具软件不尽相同, 每家厂商为了推销自己的产品, 都制定了一套采用自己公司或合作公司旗下软件的设计流程。例如, Synopsys 公司、Mentor 公司等, 都有一整套推荐流程, 这些公司推荐的流程都可以在各自公司的主页上找到, 这里不占用篇幅进行说明。尽管各家公司的推荐流程不同, 但是整个 IC 设计的基本流程是确定的。图 1-1 说明了 IC 设计的基本流程。

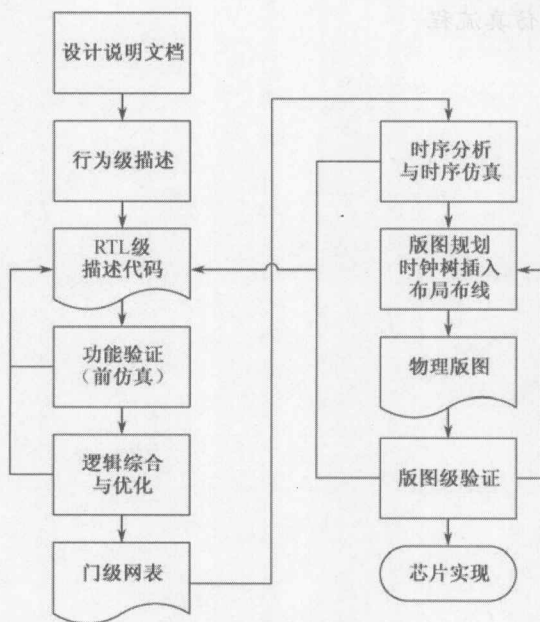


图 1-1 IC 设计的基本流程

设计的最开始阶段一定是设计文档的编写。这个设计说明文档主要包含了设计要实现的具体功能和期待实现的详细性能指标,包括电路整体结构、输入/输出(I/O)接口、最低工作频率、可扩展性等参数要求。完成设计说明文档后,需要用行为级描述待设计的电路。行为级描述可以采用高级语言,如 C/C++等,也可以采用 HDL 来编写。这个阶段的描述代码并不要求可综合,只需要搭建出一个满足设计说明的行为模型即可。

行为级描述之后是 RTL 级描述。这一阶段一般采用 VHDL 或 Verilog HDL 来实现。对于比较大的设计,一般是在行为级描述时采用 C/C++搭建模型,在 RTL 级描述阶段,逐一的对行为模型中的子程序进行代码转换,用 HDL 代码取代原有的 C/C++代码,再利用仿真工具的接口,将转换成 HDL 代码的子程序加载到行为模型中,验证转换是否成功,并依次转换行为模型中的所有子程序,最终完成从行为级到 RTL 级的 HDL 代码描述。这样做的好处是减少了调试的工作量,一个子程序转换出现错误,只需要更改当前转换的子程序即可,避免了同时出现多个待修改子程序的杂乱局面。

RTL 模型的正确与否,是通过功能验证来确定的,这一阶段也称前仿真。前仿真的最大特点就是没有加入实际电路中的延迟信息,所以,前仿真的结果与实际电路结果还是有很大差异的。不过在前仿真过程中,设计者只关心 RTL 模型是否能完成预期的功能,所以称为功能验证。

当 RTL 模型通过功能验证后,就进入逻辑综合与优化阶段。这个阶段主要是由 EDA 工具来完成,设计者可以给综合工具指定一些性能参数、工艺库等,使综合出来的电路符合自己的要求。

综合生成的文件是门级网表。这个网表文件包含了综合之后的电路信息,其中还包含了延迟信息。将这些延迟信息反标注到 RTL 模型当中,进行时序分析。主要检测的是建立时间(Setup Time)和保持时间(Hold Time)。其中建立时间的违例和保持时间较大的违例必须要修正,可以采用修正 RTL 模型或修改综合参数来完成。对于较小的保持时间违例,可以放到后续步骤中修正。对包含延迟信息的 RTL 模型进行仿真验证的过程称为时序仿真,时序仿真的结果更加逼近实际电路。

设计通过时序分析后,就可以进行版图规划与布局布线。这个阶段是把综合后的电路按一定的规则进行排布,设计者也可以添加一些参数对版图的大小和速度等性能进行约束。布局布线的结果是生成一个物理版图,再对这个版图进行仿真验证,如果不符合要求,那就需要向上查找出错点,重新布局布线或修改 RTL 模型。如果版图验证符合要求,这个设计就可以送到工艺生产线上,进行实际芯片的生产。

当然,上述流程只是一个基本的过程,其中很多步骤都是可以展开成很多细小的步骤,也有一些步骤(如形式验证)在这个流程中并没有体现。不过这个流程图可以包含基本的 IC 步骤,对于初学者已经足够了。另外,各公司推荐流程不同的原因是采用了不同的 EDA 软件来完成以上的 IC 基本流程。如前仿真阶段,可用于 HDL 仿真的 EDA 工具有 Synopsys 公司的 VCS、Cadence 公司的 Verilog-XL、明导公司的 ModelSim 等。

1.1.2 ModelSim 概述

ModelSim 是由 Mentor Graphics 公司开发的一款优秀的 HDL 工具软件。它能够提供最友好的调试环境,是唯一的单内核支持 VHDL 和 Verilog 混合仿真的仿真器。它具有如下主

要特点:

- RTL 级和门级优化, 本地编译结构, 编译仿真速度快。
- 单内核 VHDL 和 Verilog 混合仿真。
- 源代码模版和助手, 项目管理。
- 集成了性能分析、波形比较、代码覆盖等功能。
- 数据流 ChaseX。
- Signal Spy。
- C 和 Tcl/Tk 接口, C 调试。

ModelSim 具有多个版本。首先是大的版本, 从 ModelSim 4.7 开始, 不断更新, 在 2012 年推出了 10.0 版本, 随后又更新到了 ModelSim 10.1 版本。大版本更新主要是增加功能和改善性能, 不同版本之间的相比较, 最显而易见的就是菜单栏中的功能列表都会有不同幅度的变化。在大版本基础上还有小版本, 小版本以小写英文字母作为区分, 例如, ModelSim 10.1 版就有 10.1a、10.1b、10.1c 共三个不同的版本。这些版本主要是为大版本打上一些补丁, 弥补原有版本的部分 bug, 类似于计算机中的系统更新包。

除去大版本和小版本, ModelSim 的每个版本都有 SE、DE、PE 三个不同版本。这三个版本在功能上不尽相同, 表 1-1 列出了三个版本功能不同的部分, 相同的部分由于篇幅关系略去。简单来说 SE 版本是功能最完善的版本。本书中采用的是 ModelSim SE 10.1c 版本, 所有的实例和演示也均在此版本下编译进行, 操作系统使用 Windows 7 的 32 位操作系统。读者也可以下载学生版本来使用, 在 Mentor 公司的官网上简单注册即可下载并申请到许可文件, 基本功能都是可以使用的。

表 1-1 ModelSim SE/DE/PE 的功能比较

Feature	ModelSim SE	ModelSim DE	ModelSim PE
Licensing-Floating License	yes	yes	option
Language Neutral License	option		
ASIC Sign-Off	yes		
32/64-Bit Cross-Compatability	yes		
VHDL	option		option
Mixed Language	option		option
Analog/Mixed Signal(Advance MS Product)	option		
VHDL FLI	yes		
Waveform Editor	yes		
Dataflow Window	yes	yes	option
Source Annotation	yes	yes	option
C Debugger	yes	yes	option
Multiple Waveform Windows	yes		
Waveform Compare	yes	yes	option
JobSpy	yes		
User-Customizable GUI (via Tk)	yes		
Code Coverage (with Toggle Coverage)	yes	option	option
Coverage Viewer	yes	option	option
Verilog RTL & Gate	yes		
Performance Optimizations	yes		

续表

VHDL RTL & VITAL Performance Optimizations	yes		
Performance and Memory Profiler	yes	yes	option
Separate Elaboration	yes		
Integrated Sim Farm Support(via JobSpy)	yes		
Checkpoint & Restore	yes		
SWIFT Interface / SmartModels	yes		option
Synopsys Hardware Modeler Support	yes		
32-Bit OS Support	HP-UX, Linux, Solaris, Windows platform	Linux	Windows platform
64-Bit OS Support	X86-64 and Itanium-2 Linux		

1.2 ModelSim 应用基本流程

ModelSim 功能众多, 初学者刚一接触往往找不到头绪。软件自带的 User's Manual 虽然内容详尽, 但是比较烦琐, 初接触时也常常是“一头雾水”。这里暂且抛开烦琐的介绍, 以一个简单程序来演示 ModelSim 的基本操作流程, 使初学者很快能够熟悉 ModelSim 的基本功能, 更详细的软件说明和应用举例讲在后续的章节中进行介绍。

ModelSim 软件的仿真流程大致可以分为三种: Basic simulation flow (基本仿真流程)、Project flow (工程仿真流程)、Multiple library flow (多单元库仿真流程)。其中多单元库的仿真涉及到的注意点较多, 一般用于规模比较大的仿真, 故这里暂不介绍。前两种仿真流程操作比较简单, 这里用一个最基本的 Verilog HDL 程序来介绍仿真流程。这个 Verilog HDL 程序描述的是一个一位全加器, 相信本书的读者应该不会陌生。

1.3 ModelSim 基本仿真流程

Mentor Graphics 公司给出的 ModelSim 的基本仿真流程, 如图 1-2 所示, 概括为 4 步: 首先建立一个工作库, 然后编译设计文件, 再运行仿真, 最后根据仿真结果进行调试。根据此流程, 这里给出软件的具体操作步骤。

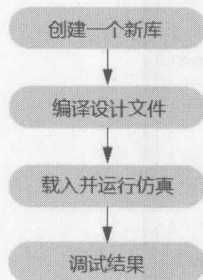


图 1-2 ModelSim 基本仿真流程



结果文件

附带光盘“Ch1\1-1”文件夹

视频教学



动画演示

附带光盘“AVT1-1.avi”

1.3.1 创建一个工作库

ModelSim 软件中，库是仿真的基础，所有仿真的程序实例都要编译入库，才能进行仿真，所以在进行仿真之前，必须先建立一个库文件。可按如下操作进行。

(1) 创建新的库

在 ModelSim 菜单栏中选中【File】→【New】→【Library】，如图 1-3 所示。

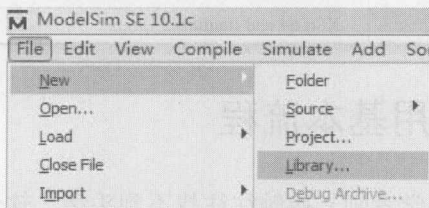


图 1-3 创建新的库

(2) 输入库名称

在弹出的对话框中输入库的名称，对话框如图 1-4 所示。对话框弹出时默认选中第三项 (a new library and a logical mapping to it)，下方两个可输入区域 Library Name 和 Library Physical Name，默认名称均为 work，这也是 ModelSim 中默认的库名称。由于是第一个例子，这里的选项都不进行设置，全部采用默认设置。

(3) 新的库建成

在第二步中单击 OK，即可完成库的创建。新的库会出现在 ModelSim 的库标签页中，如图 1-5 所示。最上方的 work (empty) 就是新建好的库，后面的 empty 表示此库是空的。可以看到，在 work 库下面还有很多库文件，这些都是 ModelSim 自带的库，可以在设计中使用。

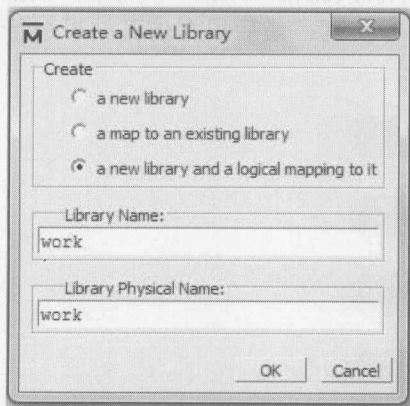


图 1-4 库的设置

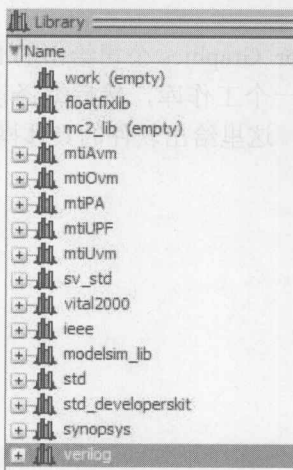


图 1-5 建好的库文件

库建好的同时，在 ModelSim 的下方窗口中也会出现对应的文字提示，如图 1-6 所示。这里读者可以先注意最上方的两行语句：vlib work 和 vmap work work，这是 ModelSim 的命

命令行操作语句，熟练使用这些语句可以大大加快仿真流程，这些都在后续章节中介绍。

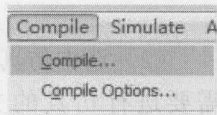
```
vlib work
vmap work work
# Copying C:\modeltech_10.1c\win32\..\modelsim.ini to modelsim.ini
# Modifying modelsim.ini
# ** Warning: Copied C:\modeltech_10.1c\win32\..\modelsim.ini to modelsim.ini.
# Updated modelsim.ini.
```

图 1-6 提示信息

1.3.2 编译设计文件

建好库后就可以编译文件，具体的步骤如下：

① 打开编译窗口。在菜单栏中选择【Compile】→【Compile】，



如图 1-7 所示。

② 选择编译文件。打开之后的编译文件窗口如图 1-8 所示。在查找范围下拉菜单中选中本设计的文件夹 1-1，可以看到包含两个文件“fulladd.v”和“test.v”，选中这两个文件，单击右下角的 Compile 按钮，即可编译仿真文件。需要说明的是，图 1-8 中最上方有一个下拉菜单 Library，这里显示的是 work，就是在第一步中建立的库文件。在实际操作过程中，如果不需要建立新的库，也可以跳过库文件的建立，直接进行设计文件的编译，在此界面中选中库名称即可。

③ 完成编译。单击 Compile 之后，刚才为空的 work 库就有了编译好的文件，如图 1-9 所示。



图 1-8 编译文件窗口

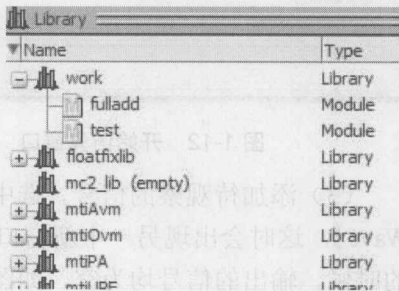


图 1-9 编译好的设计文件

与此同时，在 ModelSim 下方的窗口中也会有对应提示信息，如图 1-10 所示

```
vlog -reportprogress 300 -work work C:\modeltech_10.1c\examples\1-1\fulladd.v
# Model Technology ModelSim SE vlog 10.1c Compiler 2012.07 Jul 27 2012
# -- Compiling module fulladd
#
# Top level modules:
#   fulladd
vlog -reportprogress 300 -work work C:\modeltech_10.1c\examples\1-1\test.v
# Model Technology ModelSim SE vlog 10.1c Compiler 2012.07 Jul 27 2012
# -- Compiling module test
#
# Top level modules:
#   test
```

图 1-10 提示信息

1.3.3 运行仿真

编译通过的文件就可以进行仿真，仿真的具体步骤如下：

(1) 开始仿真。仿真的方式有很多，这里采用最简单的方式，单击快捷栏的仿真按钮开始仿真，如图 1-11 所示。左侧的按钮是开始仿真，右侧的按钮是停止仿真。

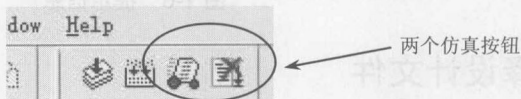


图 1-11 利用快捷工具栏仿真

(2) 选中仿真文件。开始仿真后会出现 Start Simulation 对话框，如图 1-12 所示。选中需要进行仿真的文件，在这里选中顶层模块 test，同时把下方【Optimization】区域的【Enable optimization】选项取消。单击 OK 后，在 Workspace 区域会出现新的标签 sim，同时在命令窗口还会有对应的提示信息，如图 1-13 所示。

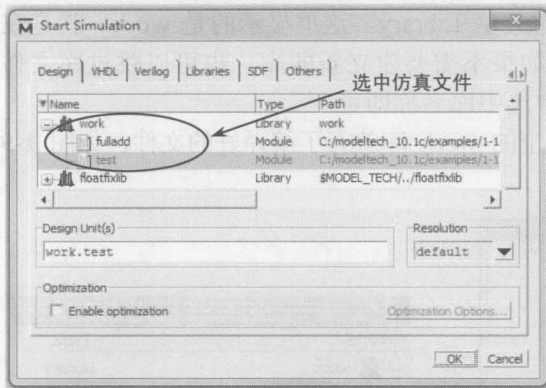


图 1-12 开始仿真窗口

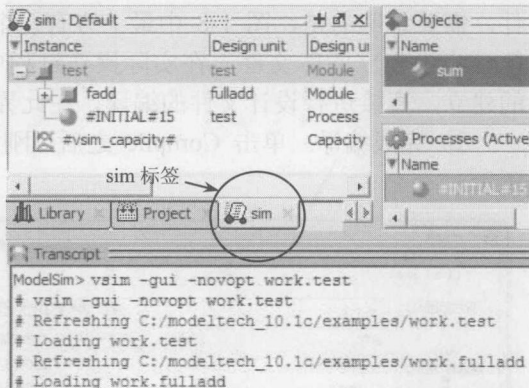


图 1-13 仿真标签及命令窗口提示

(3) 添加待观察的信号。选中 test 模块，单击右键，在右键弹出的菜单中选中【Add to Wave】，这时会出现另一个新窗口：wave。这里就是观察信号变化的区域，在仿真没有运行的时候，输出的信号均为空，如图 1-14 所示。

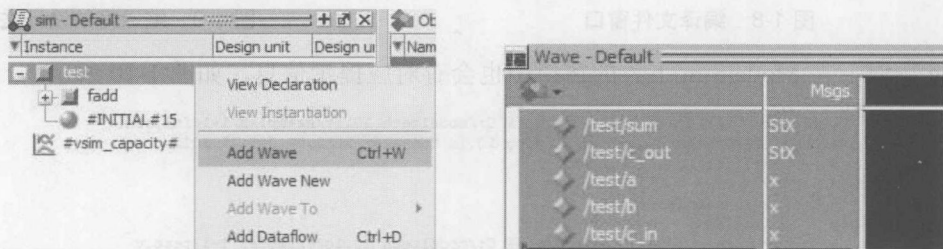


图 1-14 添加观察信号

(4) 运行仿真。快捷栏中也有运行仿真按钮，如图 1-15 所示。共有四个运行按钮，从左到右依次为 Run、ContinueRun、Run-All 和 Break。这里单击 Run-All 进行仿真。

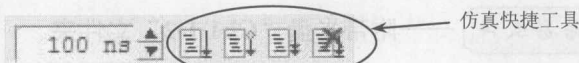


图 1-15 仿真工具按钮

1.3.4 查看结果

在单击 Run-All 后，可以在波形窗口（wave 窗口）观察输入与输出信号的变化，如图 1-16 所示。如果在设计中有一些系统函数（如 \$display）等，在命令窗口还会看到相应的提示。在本例中命令窗口没有输出。

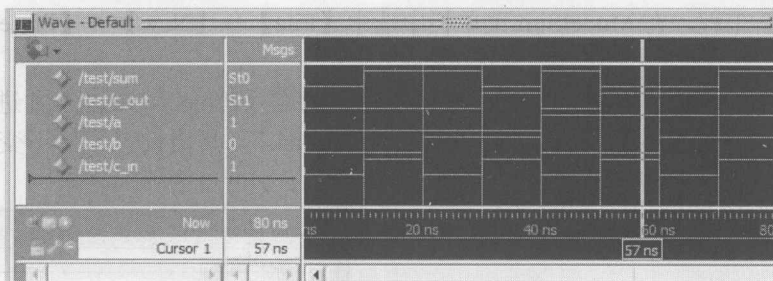


图 1-16 wave 窗口的输入与输出波形

至此，ModelSim 的基本仿真流程就结束了，根据最后的仿真波形，可以验证程序是否正确。以光标处为例，波形的高电平处为信号 1，低电平处为信号 0，输入的信号 a 为 1，b 为 0，c_in 为 1，在全加器中可知输出的结果应该为 10，对比上方的信号，c_out 为 1，sum 为 0，结果正确。

1.4 ModelSim 工程仿真流程

ModelSim 的工程仿真流程如图 1-17 所示，概括为 5 步：首先建立一个工程，然后向工程中添加设计文件，接下来编译设计文件，之后运行仿真，再进行调试。

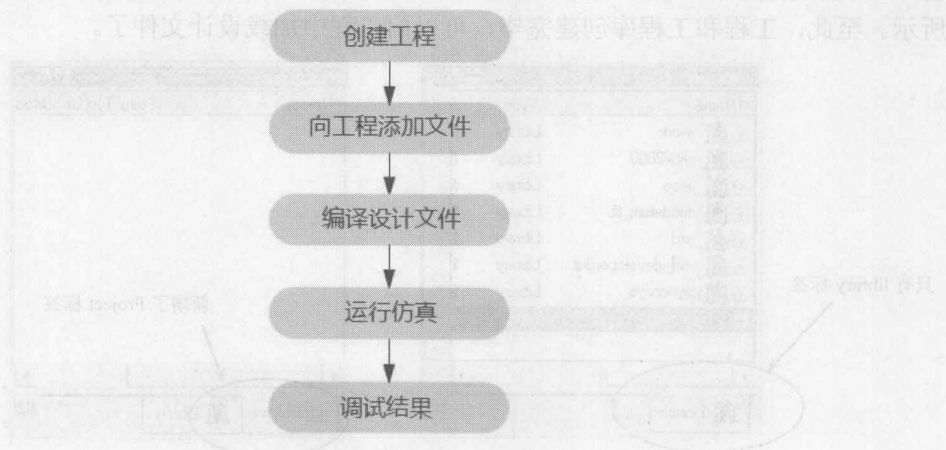


图 1-17 ModelSim 工程仿真流程