

工程设计与分析系列

ModelSim

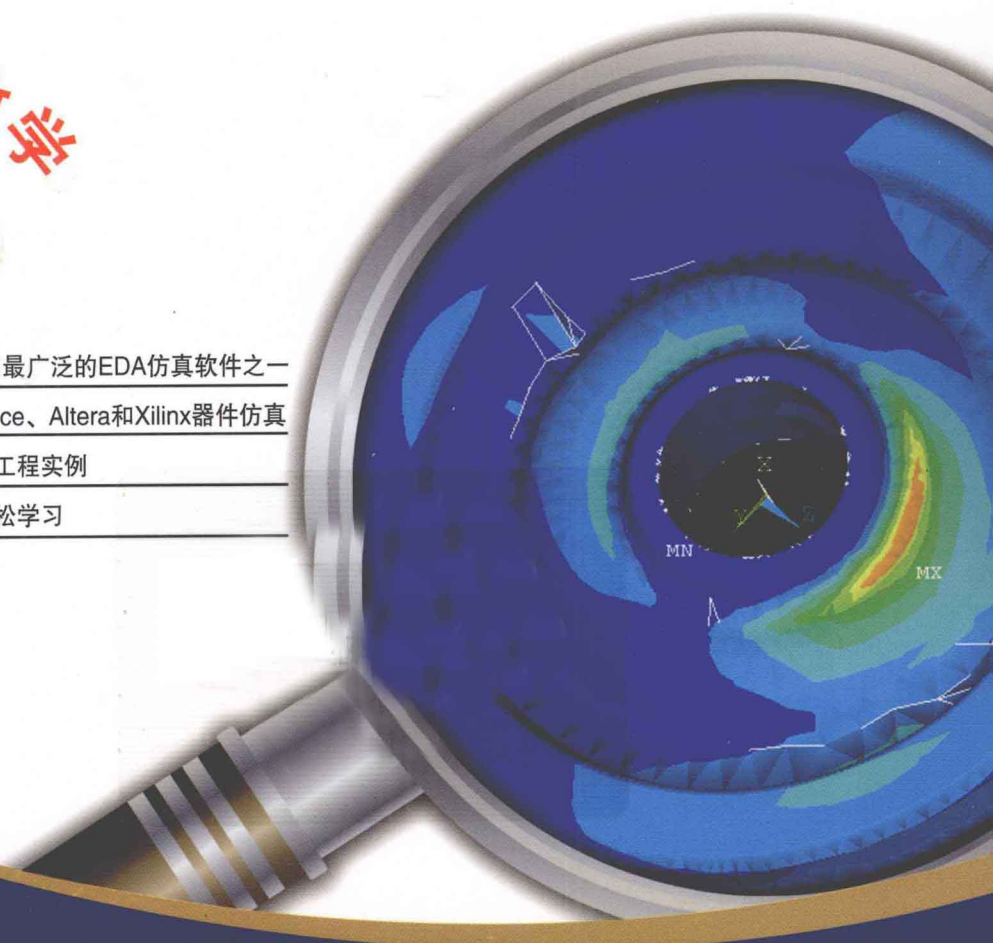
电子系统分析及仿真

于 斌 米秀杰 主编

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

于 斌 米秀杰 主编

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, 是业界唯一的单内核支持 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 6.1f 版软件为例, 由浅入深、循序渐进地介绍了 ModelSim 6.1f 软件各部分知识, 包括 ModelSim 6.1f 的基础知识、菜单命令、库和工程的建立与管理、Verilog/VHDL 文件的编译仿真及采用多种方式分析仿真结果等知识。书中配有大量的插图和详细的讲解, 并结合实例讲解使用 ModelSim 进行仿真操作的基本知识和方法技巧。

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

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, 米秀杰主编. —北京: 电子工业出版社, 2011.12

(工程设计与分析系列)

ISBN 978-7-121-14904-7

I. ①M… II. ①于…②米… III. ①VHDL 语言—仿真程序—研究 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2011) 第 218771 号

策划编辑: 许存权

责任编辑: 陈韦凯 特约编辑: 刘海霞

印 刷: 北京市顺义兴华印刷厂

装 订: 三河市双峰印刷装订有限公司

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: 787×1092 1/16 印张: 19 字数: 487 千字

印 次: 2011 年 12 月第 1 次印刷

册 数: 4000 册 定价: 46.00 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 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。



前 言

ModelSim 是 Mentor Graphics 公司开发的 EDA 工具软件，它是一款主要应用于 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，不仅仅支持对 HDL 的仿真，还可以支持 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 厂商都提供了和 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。

本书是结合作者多年的教学和工作经验的基础上编写的，在编写过程中，本书突出了以下特点：

1. 直观易懂性

全书以图解实例的形式介绍基础知识和实例操作，所有的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

采用了 ModelSim 6.1f 版本为例进行讲解，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有可扩展性。


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，书上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，能够更好的使读者理解操作的过程，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知识，而且每个知识点和实例都做了尽可能详细的讲解，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**，用户可以在观看录像中增强知识点的理解。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。

本书共分为 9 章，依次介绍了 ModelSim 6.1f 基本知识、操作界面、工程和库、对不同语言的仿真、仿真分析、与其他软件的协同仿真、对不同公司器件的后仿真、ModelSim 的其他功能、ModelSim 的文件和脚本等内容。

第 1 章概述。介绍了 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，包括菜单栏中各条命令的基本功能和主界面中工作区、命令区、MDI 区的功能，并介绍了仿真中经常使用到的窗口。

第3章工程和库。介绍了工程和库的相关知识，给出了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。并给出了对 SystemC 的仿真方法和三种语言混合仿真需要注意的事项。

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、数据流窗口使用等内容，在章末配有多个实例来演示这些功能。

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。介绍了利用 FPGA 开发工具与 ModelSim 联合进行后仿真的过程，以 Altera 和 Xilinx 两大主流厂商的开发工具为例，并结合实例演示。

第8章 ModelSim 的其他功能。介绍了 ModelSim 的零散功能，包括 C 调试、性能剖析、覆盖率检测、信号探测和利用 JobSpy 进行批处理。

第9章 ModelSim 的文件和脚本。介绍前8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，在最后一章中统一进行讲解。

本书由于斌、米秀杰主编，参与编写和光盘开发的人员有谢龙汉、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、刘新东等。由于时间仓促，书中难免有疏漏之处，请读者谅解。读者可通过电子邮件 yu_bin@vip.sina.com 与我们交流。

编者



目 录

第 1 章 概述	1	2.6.3 列表窗口	38
1.1 IC 设计与 ModelSim	1	2.6.4 数据流窗口	39
1.1.1 IC 设计基本流程	1	2.6.5 属性窗口	39
1.1.2 ModelSim 概述	3	2.6.6 进程窗口	40
1.2 ModelSim 应用基本流程	5	2.6.7 对象窗口	40
1.2.1 创建工程及工程库	5	2.6.8 存储器窗口	40
1.2.2 创建新文件	6	2.7 界面的设置	41
1.2.3 加载设计文件	6	2.7.1 定制用户界面	41
1.2.4 编译源文件	7	2.7.2 设置界面参数	43
1.2.5 运行仿真	8	第 3 章 工程和库	45
1.2.6 查看结果	9	3.1 ModelSim 工程	45
1.2.7 工程调试	9	3.1.1 删除原有工程	45
第 2 章 操作界面	11	3.1.2 开始一个新工程	46
2.1 整体界面	11	3.1.3 工程标签	47
2.2 菜单栏	12	3.1.4 工程编译	48
2.2.1 File 菜单	12	3.1.5 仿真环境配置	51
2.2.2 Edit 菜单	16	3.1.6 工程文件组织	53
2.2.3 View 菜单	19	3.1.7 工程及文件属性设置	54
2.2.4 Format 菜单	23	实例 3-1 工程文件管理	59
2.2.5 Compile 菜单	23	3.2 ModelSim 库	62
2.2.6 Simulate 菜单	25	3.2.1 概述	62
2.2.7 Add 菜单	27	3.2.2 库的创建及管理	63
2.2.8 Tools 菜单	28	3.2.3 资源库管理	65
2.2.9 Window 菜单	33	3.2.4 导入 FPGA 的库	65
2.2.10 Help 菜单	33	第 4 章 ModelSim 对不同语言的仿真	67
2.3 工具栏	34	4.1 VHDL 仿真	67
2.4 工作区	34	4.1.1 VHDL 文件编译	67
2.5 命令窗口	35	4.1.2 VHDL 设计优化	69
2.6 MDI 窗口	36	4.1.3 VHDL 设计仿真	74
2.6.1 源文件窗口	37	4.1.4 还原点和仿真恢复	79
2.6.2 波形窗口	37	4.1.5 TEXTIO 的使用	80

实例 4-1 VHDL 设计的仿真全过程	82	5.5 ModelSim 波形分析	149
4.2 Verilog 仿真	87	5.5.1 波形窗口和列表窗口	149
4.2.1 Verilog 文件编译	88	5.5.2 时间标记	151
4.2.2 Verilog 设计优化	89	5.5.3 窗口的缩放	151
4.2.3 Verilog 设计仿真	89	5.5.4 在窗口中搜索	153
4.2.4 还原点和仿真恢复	94	5.5.5 窗口的格式编排	154
4.2.5 单元库	94	5.5.6 波形和列表的保存	157
4.2.6 系统任务和系统函数	95	5.5.7 信号总线	158
4.2.7 编译指令	97	5.5.8 其他功能	159
实例 4-2 32 位浮点乘法器的 Verilog 仿真过程	98	5.5.9 波形比较	160
4.3 SystemC 仿真	104	5.6 存储器的查看和操作	165
4.3.1 概述	104	5.6.1 存储器的查看	166
4.3.2 SystemC 文件的编译和链接	105	5.6.2 存储数据的导出	167
4.3.3 设计仿真和调试	110	5.6.3 存储器初始化	168
4.3.4 常见错误	111	5.6.4 存储器调试	168
4.4 混合语言仿真	114	5.7 数据流窗口的使用	169
4.4.1 编译过程与公共设计库	114	5.7.1 概述	169
4.4.2 映射数据类型	116	5.7.2 设计连通性分析	171
4.4.3 VHDL 调用 Verilog	120	5.7.3 信号追踪和查找	172
4.4.4 Verilog 调用 VHDL	122	5.7.4 设置和保存打印	173
4.4.5 SystemC 调用 Verilog	122	5.8 综合实例	174
4.4.6 Verilog 调用 SystemC	123	实例 5-1 三分频时钟的分析	175
4.4.7 SystemC 调用 VHDL	124	实例 5-2 同步 FIFO 的仿真分析	181
4.4.8 VHDL 调用 SystemC	125	实例 5-3 基 2 的 SRT 除法器的仿真 分析	186
第 5 章 利用 ModelSim 进行仿真 分析	127	第 6 章 ModelSim 的协同仿真	194
5.1 仿真概述	127	6.1 ModelSim 与 Debussy 的协同仿真	194
5.2 WLF 文件和虚拟对象	128	6.1.1 Debussy 工具介绍	194
5.2.1 保存仿真状态	129	6.1.2 Debussy 配置方法	199
5.2.2 Dataset 结构	131	实例 6-1 与 Debussy 的协同仿真	202
5.2.3 Dataset 管理	132	6.2 ModelSim 与 Matlab 的协同仿真	209
5.2.4 虚拟对象	134	6.2.1 简介	209
5.3 利用波形编辑器产生激励	137	实例 6-2 与 Matlab 的协同仿真	211
5.3.1 创建波形	137	实例 6-3 与 Simulink 的协同仿真	214
5.3.2 编辑波形	141	第 7 章 ModelSim 对不同公司器件的后 仿真	221
5.3.3 导出激励文件并使用	144	7.1 ModelSim 对 Altera 器件的后仿真	221
5.4 采用描述语言生成激励	146	7.1.1 Quartus II 简介	221

7.1.2 后仿真流程	224	8.3.2 覆盖率的查看	264
实例 7-1 直接采用 Quartus II 调用 ModelSim 进行仿真	224	8.3.3 覆盖率检测的设置	267
实例 7-2 先用 Quartus II 创建工程, 再用 ModelSim 进行时序仿真	235	8.3.4 覆盖信息报告	267
7.2 ModelSim 对 Xilinx 器件的后仿真	240	8.4 信号探测	270
7.2.1 ISE 简介	241	8.5 采用 JobSpy 控制批处理仿真	273
7.2.2 后仿真流程	242	8.5.1 JobSpy 功能与流程	273
实例 7-3 用 ISE 对全加器进行时序仿真	244	8.5.2 运行 JobSpy	274
第 8 章 ModelSim 的其他功能	252	第 9 章 ModelSim 的文件和脚本	277
8.1 C 调试	252	9.1 SDF 文件	277
8.1.1 概述	252	9.1.1 SDF 文件的指定和编译	277
8.1.2 C 步进调试与调试设置	254	9.1.2 VHDL 的 SDF	279
8.2 ModelSim 的剖析工具	255	9.1.3 Verilog 的 SDF	280
8.2.1 运行性能剖析和存储器剖析	255	9.1.4 SDF 文件信息	282
8.2.2 查看性能剖析结果	256	9.2 VCD 文件	284
8.2.3 查看存储器剖析报告	258	9.2.1 创建一个 VCD 文件	284
8.2.4 保存结果	259	9.2.2 使用 VCD 作为激励	286
8.3 覆盖率检测	260	9.2.3 VCD 任务	288
8.3.1 启用代码覆盖	260	9.2.4 端口驱动数据	289
		9.3 Tcl 和 DO 文件	291
		9.3.1 Tcl 命令	291
		9.3.2 Tcl 语法	292
		9.3.3 ModelSim 的 Tcl 时序命令	293
		9.3.4 宏命令	294



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

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, Synopsis 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 说明了基本的 IC 设计流程。

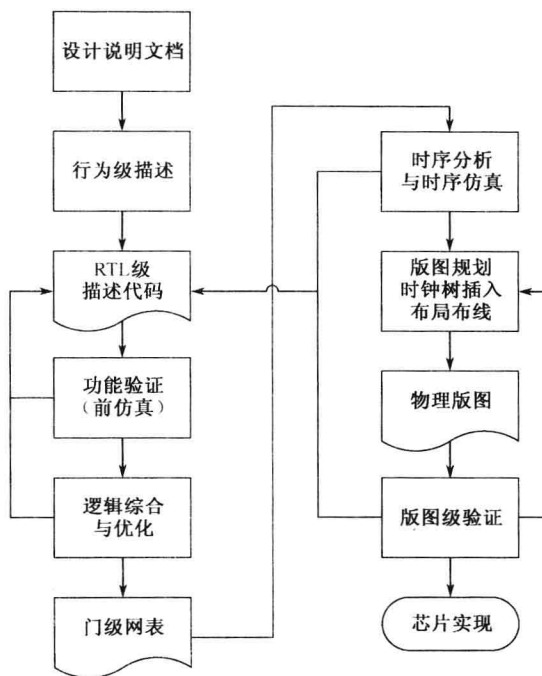


图 1-1 IC 设计基本流程

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,其中还包括了延迟信息。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 工具就有 Synopsis 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 开发的一款优秀的 HDL 仿真器。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 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,到今天已经更新到了 ModelSim 6.5。大的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的版本,如 ModelSim 6.1 版就有 6.1、6.1a、6.1b 直至 6.1f 共 7 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缺陷，类似于电脑中的系统更新包。

除去大版本和小版本，ModelSim 的每个版本都有 SE、LE、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 6.1f 版本，所有的实例和演示也均在此版本下编译进行。

表 1-1 ModelSim SE/LE/PE 的功能比较

Feature	ModelSim SE	ModelSim LE	ModelSim PE
Licensing-Floating License	yes	yes	option
Language Neutral License	option		
ASIC Sign-Off	yes		
32/64-Bit Cross-Compatibility	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 Performance Optimizations	yes		
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 98/NT/ME/2000/XP	Linux	Windows 98/NT/ME/2000/XP
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, 但是比较繁琐, 初接触也常常是一头雾水。这里我们介绍最基本的 ModelSim 应用流程, 并通过一个简单例子进行演示, 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将在后续的章节中进行介绍。

 **结果文件** —— 附带光盘 “Ch1\1-1” 文件夹

 **结果文件** —— 附带光盘 “AV1\1.avi”

1.2.1 创建工程及工程库

开始一个设计之前, 首先要在 ModelSim 中创建一个工程和对应的工程库。这里并不把所有的创建方式都列举出来, 仅采用新建工程的方式直接创建默认工程库, 这种方式比较简便。具体的操作按如下的步骤进行:

(1) 创建新工程

在 ModelSim 菜单栏中选择【File】→【New】→【Project】, 如图 1-2 所示。

(2) 输入工程名称

在弹出的对话框中输入工程名 (Project Name) 和库名称, 这里直接采用默认库 work, 输入的工程名称为 “quick”, 输入完毕后单击 OK 按钮完成, 如图 1-3 所示。

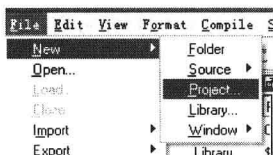


图 1-2 创建新工程

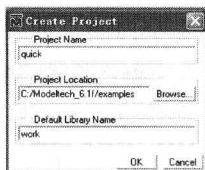


图 1-3 输入工程名称

(3) 创建工程完毕

在第二步中单击 OK 按钮后, 新的工程和工程库就被创建了。创建工程前, ModelSim 的 Workspace 窗口中只有 Library 一个标签, 当创建工程结束后, Workspace 窗口出现了新标签 Project。由于新建的工程中没有文件, 所以显示为空白区域, 如图 1-4 所示。至此, 工程和工程库创建完毕, 可以向工程中加载设计文件了。

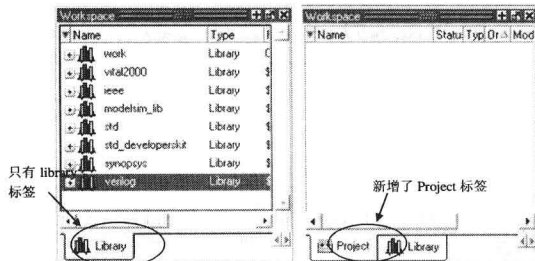


图 1-4 创建工程的前后对比

1.2.2 创建新文件

向工程中添加设计文件可以有两种方式：创建新文件和加载设计文件。创建新文件步骤如下：

(1) 创建新文件

在创建工程结束后，会弹出对话框，有多种添加方式可供选择。在本例中选择其中的 Create New File 选项，如图 1-5 所示。

(2) 输入文件名

在弹出的对话框中输入文件名称，在本例中输入 fulladd，将 Add file as type 选项选为 Verilog，单击 OK 按钮完成操作，如图 1-6 所示。

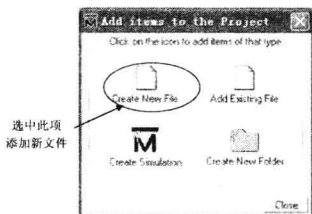


图 1-5 选择创建新文件

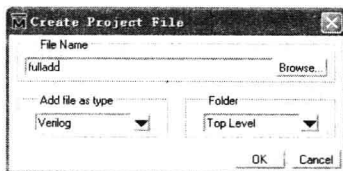


图 1-6 输入文件名

(3) 新建文件完毕

单击 OK 按钮后，就可以在 Project 窗口中看到新加入的文件“fulladd.v”，这时可以对该文件进行设计输入。双击文件，即可在编辑窗口看到文件内部的内容。由于是新建的文件，可以看到内部是空白的，如图 1-7 所示。

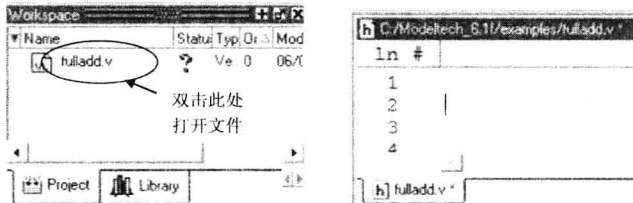


图 1-7 添加文件完成



第二步输入的文件名不能是“fulladd.v”，因为后缀的形式是在 Add file as type 中定义的！

1.2.3 加载设计文件

除了新建文件，还可以向工程中添加已有的设计文件。具体步骤如下：

(1) 选择添加已有文件

在创建工程结束弹出的对话框中，选择 Add Existing File 选项，如图 1-8 所示。

(2) 选择文件路径

选择添加已有文件后会有如图 1-9 所示的对话框，选择添加文件的目录。ModelSim 默认的路径是安装文件夹中的 example 目录。当然，也可以手动选择其他目录添加文件。这里将 example 目录下的“test.v”文件加载到工程中。

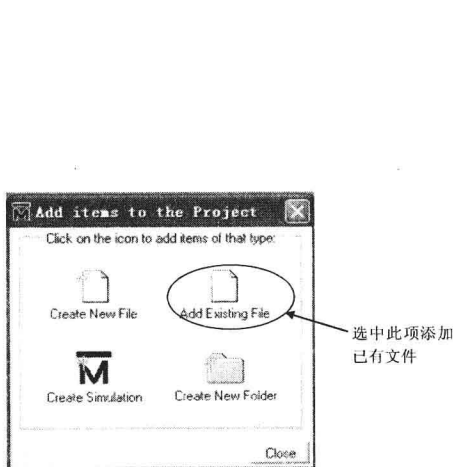


图 1-8 选择添加已有文件

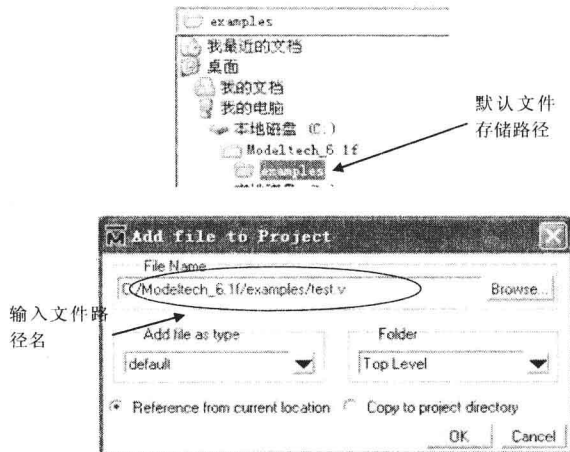


图 1-9 添加已有文件路径名

(3) 加载完成

单击 OK 按钮后，可以看到 Project 窗口又加入了一个“test.v”文件，如图 1-10 所示。细心的读者会发现加入的两个文件，它们对应的 Status 栏都是“? ”。这是该设计文件还没有被编译的标志，接下来就要编译文件。

1.2.4 编译源文件

编译过程是仿真器在检查被编译文件是否有语法错误。没有被编译的文件是不可以进行仿真的。编译的方式这里先简单介绍两种。

(1) 利用菜单选项编译

在 Project 标签中选择一个文件，右击鼠标会出现菜单，选择其中的 Compile 选项，会出现一系列的编译方式。最常用的是前两个，即编译选中文件 Compile Selected 和编译所有文件 Compile All，如图 1-11 所示。

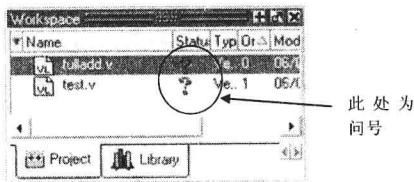


图 1-10 成功添加已有文件

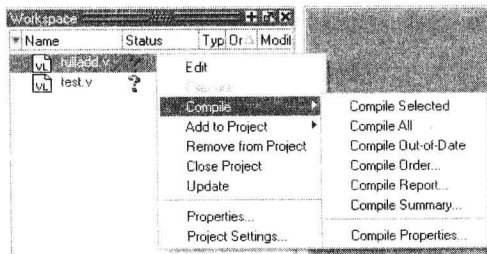


图 1-11 利用菜单编译

(2) 利用快捷工具栏编译

在 ModelSim 菜单栏下方有一排快捷工具栏，其中有编译按钮，可以直接单击对文件进行编译，如图 1-12 所示。共有两个编译按钮，左侧的是编译选中文件，右侧的是编译所有文件。

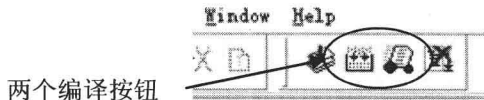


图 1-12 利用快捷工具编译

编译通过后原有的问号会变成对号，同时在命令窗口中会出现提示：“Compile of XXX was successful”，如图 1-13 所示。

1.2.5 运行仿真

编译通过的文件就可以进行仿真了。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 1-14 所示。左侧的按钮是开始仿真，右侧的按钮是停止仿真。

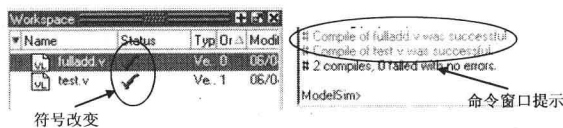


图 1-13 编译后的提示

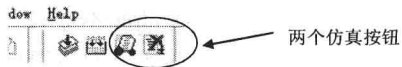


图 1-14 利用快捷工具栏仿真

(2) 选中设计文件

开始仿真后会出现 Start Simulation 对话框，如图 1-15 所示。选中需要进行仿真的文件，在这里选中顶层模块 test。单击 OK 按钮后，在 Workspace 区域会出现新的标签 sim，同时在命令窗口还会有载入文件的提示，如图 1-16 所示。

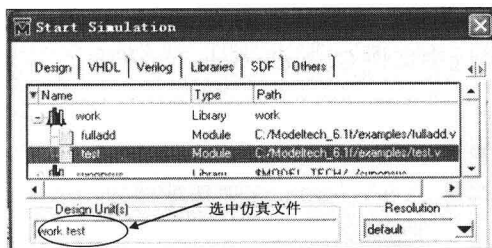


图 1-15 选中仿真文件

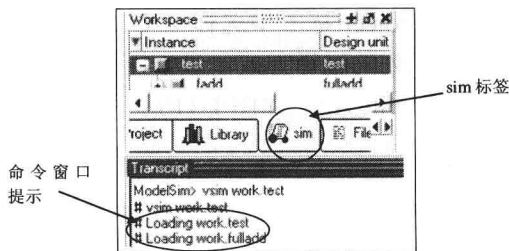


图 1-16 仿真标签及命令窗口提示

(3) 添加待观察的信号

选中 test 模块，在右键弹出的菜单中选择【Add】→【Add to Wave】，这时会出现另一个新窗口：wave-default。这里就是观察信号变化的区域，在仿真没有运行时，输出的信号均为空，如图 1-17 所示。

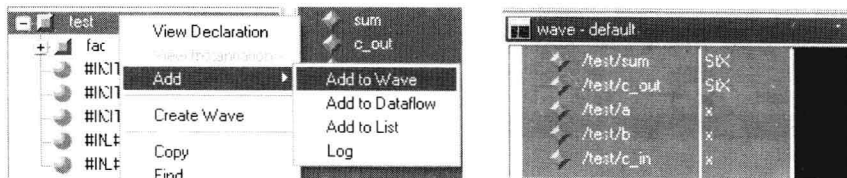


图 1-17 添加观察信号

(4) 运行仿真

快捷栏中也有运行按钮，如图 1-18 所示。共有三个运行按钮，从左到右依次为 Run、ContinueRun 和 Run -All。这里单击 Run-All 按钮进行仿真。



图 1-18 运行仿真

1.2.6 查看结果

在单击 Run-All 按钮后，可以在波形窗口（wave 窗口）观察输入/输出信号的变化，如图 1-19 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。

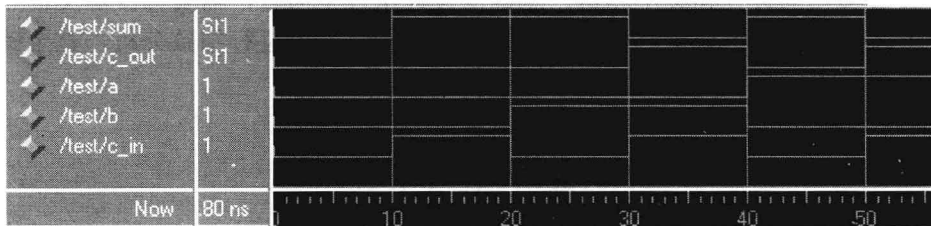


图 1-19 wave 窗口的输入/输出波形

1.2.7 工程调试

在实际的设计当中，错误是不可避免的，ModelSim 提供了丰富的错误提示类型，帮助设计者快速发现错误的位置和错误的类型。一般情况下调试过程如下：

(1) 编译错误提示

此时 Status 栏会显示一个红色的叉，表示编译不通过，即源文件中有错误。此时在命令窗口中会出现红色字体的提示，告知设计者哪个文件出现了几个错误，可能包含 error，也可能包含 warning，如图 1-20 所示。

(2) 查找错误原因和位置

双击命令窗口中的提示，就会弹出一个对话框的提示，会显示出在文件的第几行出现了哪种错误。在图 1-21 中提示，在文件的第七行出现了语法错误。这时，文件的第七行会以醒目的颜色标出来，方便设计者查找。