



高等学校电子与通信工程类专业“十二五”规划教材

基于FPGA的现代 数字系统设计

主 编 刘桂华
副主编 罗 亮



西安电子科技大学出版社
<http://www.xduph.com>

高等学校电子与通信类专业“十二五”规划教材

基于 FPGA 的现代数字系统设计

主 编 刘桂华
副主编 罗 亮
参 编 魏东梅 龙惠民 李家会
赵海龙 何燕玲 秦明伟



西安电子科技大学出版社

内 容 简 介

本书以 Xilinx 公司 FPGA 的开发为重点, 主要内容包括现代数字系统设计技术概述、基于原理图的设计、基于 Verilog HDL 语言的设计、基于 IP Core 的设计、系统仿真、可编程逻辑器件原理、基于 FPGA 的系统级设计技术、在线逻辑分析技术和其它设计工具简介。

本书内容全面、新颖, 注重基础又兼顾前沿。编写风格上尽量体现学生易学、教师易教等特点。书中涉及的例子具有典型性和实用性, 大部分章后有实验项目供课程实践环节选做, 附录中还有相关的设计课题供学生在课程设计时选用。

本书可作为高等工科院校本、专科电子电气信息类专业的教材及信息类专业课程设计、综合设计的教学参考书, 也可作为参加电子设计竞赛者和 FPGA 开发应用人员的自学参考书。

图书在版编目(CIP)数据

基于 FPGA 的现代数字系统设计/刘桂华主编. —西安: 西安电子科技大学出版社, 2012.9

高等学校电子与通信类专业“十二五”规划教材

ISBN 978-7-5606-2814-1

I. ①基… II. ①刘… III. ①可编程逻辑器件—数字系统—系统设计—高等学校—教材
IV. ①T332.1

中国版本图书馆 CIP 数据核字(2012)第 111674 号

策 划 毛红兵

责任编辑 毛红兵

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfxb001@163.com

经 销 新华书店

印刷单位 陕西华沐印刷科技有限责任公司

版 次 2012 年 9 月第 1 版 2012 年 9 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印张 20.5

字 数 487 千字

印 数 1~3000 册

定 价 35.00 元

ISBN 978-7-5606-2814-1 / TP·1342

XDUP 3106001-1

*** 如有印装问题可调换 ***

本社图书封面为激光防伪覆膜, 谨防盗版。

前 言

随着微电子技术的飞速发展，传统的可编程逻辑器件正在向大容量、高性能、低成本的方向发展，以 FPGA 为代表的可编程逻辑器件应用日益广泛。在国内，越来越多的电子设计工程师迫切希望掌握 FPGA 设计工具，了解 FPGA/CPLD 器件结构特性，本书正是在这样的背景下编写的。

本书的内容安排以学生的认知规律作为指导原则，以 Xilinx 公司 ISE 10.1 集成软件的使用为线索，从设计和实现两个角度衔接各个知识点，并结合了作者多年来的理论和实践教学经验。全书共分为 9 章。第 1 章为现代数字系统设计技术概述；第 2 章全面介绍 Xilinx 公司的 ISE 10.1 系列软件基本开发工具、开发流程和方法；第 3 章详细介绍了 Verilog HDL 语言的门级、数据流、行为级建模、可综合设计及测试方法；第 4 章介绍了 IP Core 的种类选择及复用，并详细介绍了 Xilinx 公司常用 IP Core 的设计方法；第 5 章详细介绍了主流的 ModelSim 仿真软件在 FPGA 设计过程中的系统仿真验证及在 ISE 软件中的调用方法；第 6 章围绕 Xilinx 公司的 CPLD/FPGA，介绍了主流器件的结构、工作原理、配置模式及器件的选用；第 7 章系统介绍了基于 MicroBlaze 嵌入式处理器的 SOPC 系统设计，以及基于 FPGA 的 DSP 系统设计的硬、软件平台和设计方法；第 8 章介绍了 JTAG 边界扫描测试原理和基于 ChipScope Pro 软件的 FPGA 在线逻辑分析方法；第 9 章简要介绍了其它主要的工具设计软件，包括综合工具 Synplify Pro、仿真工具 Active HDL 以及集成软件 FPGA Advantage 工具的使用方法。

通过本书的学习，读者能够熟练掌握 Xilinx 公司的 ISE 开发软件和 Verilog HDL 语言，了解 Xilinx 公司的主流 FPGA 及其硬件特点，具备自主选择适当的 FPGA 器件及使用 ISE 软件进行数字系统的设计和调试的能力。本书图文并茂，突出了教材的实用性和代表性，大部分章节后安排有大量的设计实例和习题，在重要的章后还有相关的实验项目供课程实践环节选做，附录中提供了相关的设计课题供学生在课程设计时选用。本书适用于高等工科院校本、专科信息类专业学生，也可作为参加电子设计竞赛者和 FPGA 开发应用人员的自学参考书。

本书由刘桂华主编。第 1 章由刘桂华编写，第 2 章和第 7 章的前 3 节由罗亮编写，第 3 章由龙惠民编写，第 4 章和第 6 章由魏东梅、刘桂华编写，第 5 章由李家会编写，第 7 章第 4~6 节和第 8 章由赵海龙编写，第 9 章由魏东梅、李家会和何燕玲编写，附录由秦明伟编写。刘桂华和罗亮完成全书的统稿工作。

清华大学的孟宪元教授给本书提出了很多宝贵的修改意见，特在此表示衷心感谢。本书还得到 Xilinx 大学计划中国区经理谢凯年以及深圳依元素科技有限公司的大力支持和帮助。另外，一些研究生和本科生也参与了程序调试和绘图工作，在此表示衷心的感谢。

编者希望通过本书能与广大读者交流 FPGA 设计方面的体会和感受，并期望为高等学校 FPGA/CPLD 设计人才的培养贡献绵薄之力，但是限于作者的水平，书中会有不妥之处，殷切期待读者的批评和指正。

编 者

2012 年 6 月

目 录

第 1 章 现代数字系统设计技术概述	1	3.1.2 Verilog HDL 的基础知识.....	56
1.1 概述	1	3.1.3 Verilog HDL 的描述层次.....	62
1.1.1 可编程逻辑器件 PLD 概述.....	1	3.2 门级建模.....	63
1.1.2 电子设计自动化技术概述	3	3.2.1 门的类型.....	63
1.2 现代数字系统的设计流程	6	3.2.2 实例数组(Array of Instances)	65
1.3 EDA 软件工具简介	8	3.2.3 应用举例.....	66
1.4 现代数字系统设计的发展趋势	14	3.2.4 门延迟.....	66
小结	18	3.3 数据流建模.....	68
习题	19	3.3.1 连续赋值语句.....	68
第 2 章 基于原理图的设计	20	3.3.2 表达式、运算符和操作数.....	68
2.1 Xilinx ISE Foundation 介绍.....	20	3.3.3 举例.....	72
2.1.1 安装 ISE Foundation	20	3.4 行为级建模.....	73
2.1.2 ISE Foundation 界面	22	3.4.1 顺序块和并行块语句.....	73
2.1.3 ISE Foundation 的集成工具.....	23	3.4.2 条件语句.....	75
2.2 基于原理图的设计流程	25	3.4.3 循环语句.....	78
2.2.1 创建工程	25	3.4.4 赋值语句.....	81
2.2.2 原理图绘制	28	3.4.5 结构化语句.....	82
2.2.3 逻辑综合	34	3.4.6 系统任务和函数.....	87
2.2.4 物理实现	36	3.4.7 编译预处理命令.....	92
2.2.5 仿真验证	41	3.4.8 有限状态机设计	95
2.2.6 硬件配置	45	3.5 Verilog HDL 的可综合设计	100
小结	47	3.6 Testbench 文件与设计.....	102
习题	48	3.6.1 测试平台的搭建.....	102
实验项目	48	3.6.2 Testbench 文件设计.....	104
实验一 七段译码原理电路的设计与仿真	48	3.7 Verilog HDL 在 ISE 软件中设计示例.....	108
实验二 二十五进制计数器设计与仿真	50	小结.....	113
实验三 排球比赛计分显示系统设计与		习题.....	114
仿真	51	实验项目.....	116
仿真	51	实验一 七段数码显示译码器设计.....	116
第 3 章 基于 Verilog HDL 语言的设计	52	实验二 含异步清零和同步使能的四位	
3.1 Verilog HDL 概述	52	加法计数器.....	116
3.1.1 几个简单的 Verilog HDL 例子	53	实验三 桶形移位器(Barrel Shifter)设计	117
		实验四 用状态机实现序列检测器的设计... ..	117

实验五 循环冗余校验(CRC)模块设计.....	118	实验二 8位十进制频率计设计.....	175
第4章 基于IP Core的设计	120	第6章 可编程逻辑器件原理	177
4.1 IP模块的种类与应用.....	120	6.1 可编程逻辑器件的分类及特点.....	177
4.2 IP Core的选择与复用.....	121	6.2 复杂可编程逻辑器件CPLD.....	178
4.2.1 IP Core的选择.....	121	6.2.1 简单低密度PLD的原理.....	178
4.2.2 IP Core的复用.....	123	6.2.2 CPLD的结构和工作原理.....	185
4.3 IP Core生成工具简介.....	124	6.3 现场可编程门阵列FPGA.....	192
4.4 常用IP Core的设计.....	126	6.3.1 FPGA的基本结构.....	193
4.4.1 可逆计数器的设计.....	126	6.3.2 基于查找表的FPGA的结构和 工作原理.....	194
4.4.2 存储器的设计.....	129	6.4 FPGA的配置.....	203
4.4.3 时钟的设计.....	140	6.4.1 FPGA在系统可配置原理.....	204
小结.....	147	6.4.2 FPGA配置方式.....	205
习题.....	147	6.4.3 配置流程.....	208
实验项目.....	147	6.5 可编程逻辑器件的选用.....	208
实验一 计数器的设计.....	147	6.6 Xilinx器件命名.....	210
实验二 正弦波形发生器的设计.....	148	6.6.1 CPLD器件命名.....	210
实验三 双口RAM的设计.....	148	6.6.2 FPGA器件命名.....	210
第5章 系统仿真	150	小结.....	211
5.1 ModelSim软件的安装及简介.....	150	习题.....	211
5.1.1 ModelSim软件的安装.....	150	第7章 基于FPGA的系统级设计 技术	212
5.1.2 ModelSim窗口简介.....	151	7.1 基于FPGA的嵌入式开发技术.....	212
5.1.3 ModelSim的使用方式.....	154	7.1.1 嵌入式系统与SOPC技术.....	212
5.2 在ISE中调用ModelSim.....	158	7.1.2 基于SOPC的嵌入式系统开发 流程.....	214
5.2.1 建立仿真环境.....	158	7.2 MicroBlaze嵌入式处理器.....	216
5.2.2 在ISE中调用ModelSim实现 功能仿真.....	162	7.3 嵌入式开发套件EDK.....	219
5.2.3 在ISE中调用ModelSim实现 时序仿真.....	163	7.3.1 EDK的组成.....	220
5.3 在ModelSim中进行仿真.....	166	7.3.2 EDK的任务流程、工具模块与 工程管理.....	221
5.4 基于IP Core的FIFO仿真实例.....	169	7.3.3 EDK的使用.....	224
5.4.1 建立FIFO IP Core的源文件.....	169	7.4 在SOPC系统中添加定制外设IP Core.....	237
5.4.2 建立Testbench文件.....	170	7.5 Xilinx DSP设计平台.....	249
5.4.3 在ModelSim中进行仿真.....	172	7.5.1 FPGA实现DSP的特点.....	249
小结.....	174	7.5.2 FPGA的DSP硬件资源.....	251
习题.....	174	7.5.3 FPGA实现DSP的软件工具.....	256
实验项目.....	175		
实验一 数控分频器设计与仿真.....	175		

7.5.4 DSP Tools 软件	257	FPGA Advantage.....	298
7.6 FIR 滤波器设计	265	小结.....	302
7.6.1 产生 FIR 滤波器的系数	266	附录	303
7.6.2 输入 FIR 滤波器模块	268	附录 1 Spartan-3E Starter Kit Board 介绍	303
7.6.3 FIR 滤波器模型设计	268	附录 2 参考课题.....	304
7.6.4 在 Simulink 中仿真 FIR 滤波器	270	课题一 数字式竞赛抢答器.....	304
7.6.5 完善 FIR 滤波器设计	271	课题二 数字钟.....	305
小结	272	课题三 数字频率计.....	305
习题	272	课题四 拔河游戏机.....	306
实验项目	273	课题五 乒乓球比赛游戏机.....	306
实验一 利用 BSB 向导构建嵌入式基本 系统	273	课题六 交通信号灯控制器.....	307
实验二 基于 SOPC 的流水灯设计	273	课题七 电子密码锁.....	308
实验三 创建一个 12×8 的乘法累加器	274	课题八 彩灯控制器.....	308
实验四 FIR 滤波器设计	275	课题九 脉冲按键电话显示器.....	309
第 8 章 在线逻辑分析技术	277	课题十 简易电子琴.....	309
8.1 JTAG 边界扫描测试	277	课题十一 出租车自动计费器.....	309
8.2 在线逻辑分析仪 ChipScope Pro 概述	278	课题十二 洗衣机控制器.....	310
8.3 在线逻辑分析仪 ChipScope Pro 的使用 ...	280	课题十三 DDS 波形发生器.....	311
小结	287	课题十四 简单的时分复用系统.....	311
实验项目	287	课题十五 异步串口通信.....	312
实验一 移位相加八位硬件乘法器电路 设计	287	课题十六 64 K 数据适配器设计	313
实验二 DDS 设计	287	课题十七 简化 LAPS 协议实现.....	315
第 9 章 其它设计工具简介	289	课题十八 直接序列扩频通信系统.....	317
9.1 Synplicity 公司的 Synplify Pro.....	289	课题十九 设计一个 MAC FIR 滤波器	318
9.2 Aldec 公司的 Active HDL.....	295	课题二十 直接型数字下变频器.....	318
9.3 Mentor Graphics 公司的		参考文献.....	320

第 1 章 现代数字系统设计技术概述

20 世纪后半期，随着集成电路和计算机技术的飞速发展，在电子系统设计领域，可编程逻辑器件(如 CPLD、FPGA)的应用已得到广泛的普及，这些器件为数字系统的设计带来了极大的灵活性。国际上电子和计算机技术较先进的国家一直在积极探索新的电子电路设计方法，并在设计方法、工具等方面进行了彻底的变革，极大地改变了传统的数字系统设计方法、设计过程和设计观念，促进了 EDA 技术的迅速发展。

本章将简要介绍可编程逻辑器件和 EDA 技术，并详细介绍现代数字系统设计流程、EDA 工具软件和现代数字系统设计的发展趋势。

1.1 概 述

1.1.1 可编程逻辑器件 PLD 概述

随着科学技术的进步，电路系统的基本硬件已从电子管、晶体管、小规模集成电路 SSI、中规模集成电路 MSI，发展到了超大规模集成电路 VLSI 及巨大规模集成电路 GSI，数字集成电路已得到非常广泛的应用，而微处理器和专用集成电路 ASIC 的广泛应用提高了系统的可靠性与通用性，它已逐渐取代了通用全硬件 LSI 电路。ASIC 以其体积小、重量轻、功耗低、速度快、成本低、保密性好等特点脱颖而出，占据了较大的市场份额。

ASIC 是专门为某一领域或特定用户需要而设计的 LSI 或 VLSI 电路。数字系统中 ASIC 的分类如图 1.1 所示。

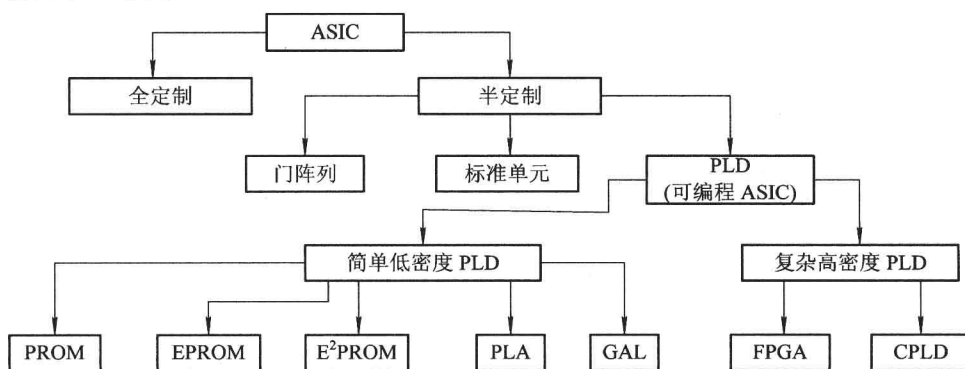


图 1.1 数字系统中 ASIC 的分类

1. 全定制 ASIC

全定制 ASIC 设计是基于晶体管级、手工设计版图的制造方法。设计人员从晶体管的版图尺寸、位置和互联线开始设计,以达到芯片面积利用率高、速度快、功耗低的最优化性能。设计者对于电路具有完全的控制权,各层掩模都是按照特定电路功能专门制造的,这种设计方式可以最大限度地实现电路性能的优化。全定制 ASIC 要求设计人员具有半导体材料和工艺技术知识,还具有完整的系统和电路设计的工程经验。全定制 ASIC 由于其设计周期很长,设计时间和成本非常高,市场风险非常大,因此多用于大批量的 ASIC 产品,例如微处理器、高压器件、A/D 转换器和传感器等专用芯片。

2. 半定制 ASIC

半定制 ASIC 是一种约束性设计方法,它是在芯片上制作一些具有通用性的单元元件或元件组的半成品硬件,用户仅需考虑电路逻辑功能和各功能模块之间的合理连接即可。这种方法简化了版图设计,提高了设计效率和性价比。对于产量规模不大的器件,可以直接采用这种方式进行生产。半定制 ASIC 按照逻辑实现的方式不同可以分为门阵列、标准单元和可编程逻辑器件 PLD(Programmable Logic Device)。

门阵列是在硅片上按照某种规范的方式制造出大量的标准门(晶体管阵列),但没有进行相互的连接。用户在设计时,根据电路的功能要求,将对应的逻辑关系表达为晶体管的互连关系,再将这种互连关系转换为连线版图,从而在门阵列基础上实现所设计的电路,它是较早使用的半定制 ASIC 设计方法。与全定制 ASIC 设计相比,这种方式涉及工艺少、造价低,适合于小批量的 ASIC 设计。门阵列设计的缺点是芯片面积利用率低,灵活性差,对设计限制过多。

标准单元是在外部尺度规范条件下对各种常用的逻辑功能单元(各种组合逻辑或时序逻辑单元)进行物理版图级的设计,形成标准单元,并创建版图单元库,包括 SSI 逻辑块、MSI 逻辑块、数据通道模块、微处理器以及 I/O 电路的专用单元阵列,供用户调用以设计不同的芯片。在标准单元设计中,所有的连线、接触点、过孔、通道已完全确定,设计者通常按照性能优化原则,根据特定的工艺条件,通过调整每个晶体管的宽度,可以在性能和面积上做到最大限度的优化。标准单元设计完毕后可以形成对应的工艺掩模文档,以便在以后的设计中重复使用。用标准单元设计 ASIC 比门阵列具有更加灵活的布图方式,可以根本解决布通率问题,是目前 ASIC 设计中应用广泛的设计方法之一。

门阵列法和标准单元法设计的 ASIC 共有的缺点是与 IC 设计工艺密切相关的。一旦工艺发生变化,则标准门或标准单元库要随之更新,这是一项十分繁重的工作。另外,需要投入大量的成本和时间,才能制作出全套的工艺掩模和相关的工艺检测系统,一旦产品检验不合格,设计需要修改,将导致巨大的损失。

可编程逻辑器件实质上是门阵列及标准单元技术的延伸和发展。可编程逻辑器件是一种半定制的逻辑芯片,但与门阵列和标准单元不同,芯片内的硬件资源和连线资源是由厂家预先定制好的,可编程逻辑器件的逻辑功能由用户通过 EDA 软件和编程器对其逻辑结构进行重新设定,它既具有硬件电路的工作速度又具有软件可编程的灵活性。可编程器件设计不需要制作任何掩模,基本不考虑布局布线问题,设计成本低。它在设计中主要考虑逻辑功能的实现,不需要考虑具体单元器件的实现,设计周期短。由于可编程器件的编程工

艺都可以反复写入和擦除，设计中存在任何问题可以马上进行修改，不需要付出硬件代价，所以设计的风险低。

PLD 从 20 世纪 70 年代发展到现在，已形成了许多类型的产品，其结构、工艺、集成度、速度和性能都在不断地改进和提高。最早期的可编程逻辑器件有可编程只读存储器(PROM)、紫外线可擦除只读存储器(EPROM)和电可擦除只读存储器(E²PROM)，其后出现了结构上稍复杂的可编程芯片，它能够完成各种数字逻辑功能，这一阶段的产品主要有可编程阵列逻辑(PAL)和通用阵列逻辑(GAL)。由于受到结构规模的限制，以上这些 PLD 只能完成简单的数字逻辑功能，称为简单低密度 PLD 器件。进入 90 年代后伴随着铜微处理器硅芯片技术的发展，可编程逻辑器件在体积与性能上得到了更良好的体现，出现了复杂高密度 PLD 器件，如 1984 年 Xilinx 公司发明的现场可编程门阵列 FPGA (Filed Programmable Gate Array)以及随后出现的复杂可编程逻辑器件 CPLD(Complex Programmable Logic Device)，它们直接面向用户，具有极大的灵活性和通用性、使用方便、开发效率高、成本低以及工作可靠性好等特点，因而很快得到普及和应用，发展非常迅速。

近十余年来，FPGA/CPLD 作为可编程逻辑器件的一个重要分支，其在结构、密度、功能、速度、性能等方面都取得了飞速的发展，如出现了集成度超过千万门、时钟频率超过千兆赫、数据传输位数达到每秒几十亿次的可编程逻辑器件。利用 CPLD 和 FPGA 来进行专用集成电路设计是目前最为流行的方式之一。如今电子设计工程师只需一台计算机、一套与器件相应的开发软件和 FPGA/CPLD 芯片就能在实验室或家中通过对 FPGA/CPLD 编程实现各种复杂的专门用途的数字集成电路，即所谓的可编程 ASIC。

但是，由于 FPGA/CPLD 的硬件资源和连线资源是厂家预先定制好的，设计者对于可编程 ASIC 电路设计的控制权有限，从而使得全定制或标准单元设计的 ASIC 在性能、速度和单位成本方面不具有竞争性。此外，也不可能用可编程 ASIC 去取代通用产品，如 CPU、存储器、A/D 和 D/A 等的应用。

为了避免设计的风险，在开发新的系统时通常采用 FPGA/CPLD 进行初步设计以验证系统设计的正确性，这已经成为一种标准的方法。在设计过程中，往往先利用 EDA 工具完成软件仿真，再利用可编程 ASIC 器件 FPGA/CPLD 进行硬件仿真，在可编程 ASIC 器件实现设计后，通过版图设计、芯片测试、制版和流片转成 ASIC 电路。

目前，为了降低单位成本，可以在可编程逻辑器件实现设计后，用特殊的方法转换成 ASIC 电路，如 Altera 公司的部分 FPGA 器件在设计成功后可以通过 HardCopy 技术转换成对应的门阵列 ASIC 产品。

1.1.2 电子设计自动化技术概述

现代数字系统设计领域中的电子设计自动化 EDA(Electronic Design Automation)技术是随着计算机辅助设计技术的提高和可编程专用集成电路 FPGA/CPLD 规模的扩大而产生，并不断完善的。由于可编程专用集成电路可以通过软件编程来对器件的硬件结构和工作方式进行重构，这一切极大地改变了传统的电子系统设计方法、设计过程，乃至设计观念。

EDA 技术融合电子技术、集成电路制造技术、计算机技术和智能化技术等，以计算机为工作平台，以相关的 EDA 软件为开发工具，以大规模可编程逻辑器件为设计载体，以硬

件描述语言(Hardware Description Language)为系统逻辑描述的主要方式,自动完成系统算法和电路设计。EDA 技术已有 30 多年的发展历程,大致可分为 20 世纪 70 年代的计算机辅助设计(CAD)阶段、80 年代的计算机辅助工程(CAE)阶段和 90 年代后的电子系统设计自动化(EDA)阶段。

20 世纪 70 年代,随着中小规模集成电路的开发应用,传统的手工制图设计印刷电路板和集成电路的方法已无法满足设计精度和效率的要求,因此工程师们开始进行二维平面图形的计算机辅助设计,以便解脱繁杂、机械的版图设计和 PCB 布局布线工作,这就是计算机辅助设计(CAD)阶段。

到了 20 世纪 80 年代,可编程逻辑器件进入商业应用,相应的辅助设计软件也投入使用。80 年代末硬件方面出现了 FPGA 和 CPLD, CAE 和 CAD 技术的应用也更加广泛,这些技术在原理图输入、自动布局布线和 PCB 分析,以及逻辑设计、逻辑仿真、布尔方程综合和化简等方面担当了重要角色。

20 世纪 90 年代后,随着科学技术的发展,电子产品的更新换代进一步加快,大规模和超大规模可编程逻辑器件 FPGA/CPLD 得到广泛应用,使电子系统设计发生了质的变化。如今,现代数字系统设计技术已进入到一个全新的阶段,出现了以高级语言描述、系统级仿真和综合技术为特征的电子设计自动化(EDA)技术。利用这一技术,设计者可以在 EDA 软件平台上使用硬件描述语言完成设计,EDA 工具自动地完成将软件方式描述的电子系统转换到硬件系统所需的逻辑编译、逻辑综合及优化、布局布线、逻辑仿真,直至对于特定 FPGA/CPLD 目标芯片的编程下载等工作。EDA 技术的出现,极大地提高了电路设计的效率和可靠性,减轻了设计者的劳动强度。设计师们摆脱了大量的辅助设计工作,而把精力集中于创造性的方案与概念构思上,从而极大地提高了设计效率,缩短了产品的研制周期。

利用 EDA 技术进行电子系统的设计,具有以下几个特点:

1) 软件硬化,硬件软化

软件硬化是指所有的软件设计最后转化成硬件来实现,用软件方式设计的系统到硬件系统的转换是由 EDA 开发软件自动完成的;硬件软化是指硬件的设计使用软件编程的方式进行,尽管目标系统是硬件,但整个设计和修改过程如同完成软件设计一样方便和高效。

现代的 EDA 软件配置了多种能兼用和混合使用的逻辑描述输入工具,例如既支持功能完善的硬件描述语言如 VHDL、Verilog HDL 等作为文本输入,又支持逻辑电路图、工作波形图等作为图形输入,具有从系统的数学模型直到门级电路多层次描述系统硬件功能的能力,而且可以将高层次的行为描述与低层次的寄存器传输级 RTL(Register Transformation Level)描述和结构描述混合使用。EDA 系统还配置了高性能的综合和优化工具,设计人员只需将设计描述程序输入到计算机,设计综合工具便能自动将其转化为适当的物理硬件实现,从而提高了设计效率,缩短了设计周期。

2) 自顶向下(top-down)的设计方法

传统的设计方法都是自底向上的,即首先确定可用的元器件,然后根据这些器件进行逻辑设计,完成各模块后进行连接,并形成系统,最后经调试、测量看整个系统是否达到规定的性能指标。这种“自下而上”的设计方法常常受到设计者的经验及市场器件情况等因素的限制,且没有明显的规律可循。另外,系统测试在系统硬件完成后进行,如果发现

系统设计需要修改,则需要重新制作电路板,重新购买器件,重新调试与修改设计。整个修改过程需要花费大量的时间与经费。再者,传统的电路设计方式是原理图设计方式,而原理图设计的电路对于复杂系统的设计、阅读、交流、修改、更新和保存都十分困难,不利于复杂系统的任务分解与综合。

基于 EDA 技术的所谓“自顶向下”的设计方法正好相反,它主要采用并行工程和“自顶向下”的设计方法,使开发者从一开始就要考虑到产品生成周期的诸多方面,包括质量、成本、开发时间及用户的需求等。该设计方法首先从系统设计入手,在顶层进行功能划分和结构设计,由于采用高级语言描述,因此能在系统级采用仿真手段验证设计的正确性,然后再逐级设计底层的结构,用 VHDL、Verilog HDL 等硬件描述语言对高层次的系统行为进行电路描述,最后再用逻辑综合优化工具生成具体的门级逻辑电路的网表,其对应的物理实现级可以是印刷电路板或专用集成电路。“自顶向下”设计方法的特点表现在以下几个方面:

- (1) 基于可编程逻辑器件 PLD 和 EDA 开发工具支撑。
- (2) 采用系统级、电路级和门级的逐级仿真技术,以便及早发现问题,进而修改设计方案。
- (3) 现代的电子应用系统正向模块化发展,或者说向软、硬核组合的方向发展。对于以往成功的设计成果稍作修改、组合就能投入再利用,从而产生全新的或派生的设计模块。
- (4) 由于采用的是结构化开发手段,所以可实现多人多任务的并行工作方式,使复杂系统的设计规模和效率大幅度提高。
- (5) 在选择器件的类型、规模、硬件结构等方面具有更大的自由度。

3) 集设计、仿真和测试于一体

现代的 EDA 软件平台集设计、仿真、测试于一体,配备了系统设计自动化的全部工具,这些工具包括:多种能兼容和混合使用的逻辑描述输入工具以及高性能的逻辑综合、优化和仿真测试工具。电子设计师可以从概念、算法、协议等开始设计电子系统,将电子产品从电路设计、性能分析到设计出 IC 版图或 PCB 版图的整个过程在计算机上自动处理完成。

EDA 仿真测试技术极大地提高了大规模系统电子设计的自动化程度。在设计各个阶段都能方便地进行仿真和测试。设计的输入、输出或中间变量之间的信号关系由计算机根据要求提供的设计方案,从各种不同层次的系统性能出发完成一系列准确的逻辑和时序仿真验证。该测试技术通过计算机就能对系统上的目标器件进行边界扫描测试。目前大部分 FPGA/CPLD 芯片都支持边界扫描技术。边界扫描测试技术标准是由 IEEE 组织联合测试行动组(JTAG)在 20 世纪 80 年代提出的,用来解决高密度引线器件和高密度电路板上的元件的测试问题。它只需要四根信号线就能够对电路板上所有支持边界扫描的芯片内部逻辑和边界管脚进行测试。

4) 在系统可现场编程,在线升级

编程是指把系统设计的程序化数据按一定的格式装入一个或多个可编程逻辑器件的编程存储单元,定义内部模块的逻辑功能以及它们的相互连接关系。早期的可编程逻辑器件需要将芯片从印制板上拆下,然后把它插在专用的编程器上进行编程,目前广泛采用的在系统可编程技术则克服了这一缺点。

所谓在系统可编程是指可编程逻辑器件不需要使用编程器,具有将器件插在系统内或电路板上仍然可以对其进行编程和再编程的能力。目前的 FPGA/CPLD 器件为设计者提供了

系统内可再编程或可再配置能力，使得系统内硬件的功能可以像软件一样易于修改，这就为设计者进行电子系统设计和开发提供了可实现的最新手段。采用这种技术对系统的设计、制造、测试和维护也产生了重大的影响，给样机设计、电路板调试、系统制造和系统升级带来革命性的变化。

5) 设计工作标准化，模块可移置共享

设计语言、EDA 的底层技术及其接口的标准化能很好地对设计结果进行交换、共享及重用。

EDA 设计工作的重要设计语言——硬件描述语言 HDL 已经逐步标准化。VHDL 在 1987 年被 IEEE 采纳为硬件描述语言标准(IEEE 1076-1987)，VHDL 同时也是军事标准(454)和 ANSI 标准。Verilog HDL 在 1995 年成为 IEEE 标准(IEEE 1364-1995)，2001 年发布了 IEEE 1364-2001。作为两大被国际 IEEE 组织认定的工业标准硬件描述语言，VHDL 和 Verilog HDL 为众多的 EDA 厂商支持，且移植性好。

数据格式的一致性通过标准来保证。EDA 的底层技术、EDA 软件之间的接口等则采用标准数据格式，这样各具特色的 EDA 工具都能被集成在易于管理的统一环境之下，并支持任务之间、项目之间、设计工程师之间的信息传输和工程数据共享，从而使 EDA 框架日趋标准化。并行设计工作和“自顶向下”设计方法也是构建电子系统集成设计环境或集成设计平台的基本规范。目前，主要的 EDA 系统都建立了框架结构，并且都遵循国际计算机辅助设计框架结构组织 CFI(CAD Framework International)的统一技术标准。因此，EDA 技术代表了当今数字系统设计技术的最新发展方向。

传统设计方法和 EDA 设计方法的主要区别如表 1.1 所示。

表 1.1 传统设计方法和 EDA 设计方法的不同

传统设计方法	EDA 设计方法
自底向上	自顶向下
手动设计	自动设计
硬、软件分离	打破硬、软件屏障
原理图方式设计	原理图、VHDL 语言等多种设计方式
系统功能固定	系统功能易变
不易仿真	易仿真
难测试修改	易测试修改
模块难移置共享	设计工作标准化，模块可移置共享
设计周期长	设计周期短

1.2 现代数字系统的设计流程

现代数字系统的设计流程是指利用 EDA 软件和编程工具对可编程逻辑器件进行开发的过程。在 EDA 软件平台上，利用硬件描述语言 HDL 等逻辑描述手段完成设计，然后结合多层次的仿真技术，在确保设计的可行性与正确性的前提下完成功能确认，接着利用 EDA

工具的逻辑综合功能,把功能描述转换成某一具体目标芯片的网表文件,输出给该器件厂商的布局布线适配器,进行逻辑化简及优化、逻辑映射及布局布线,再利用产生的仿真文件进行功能和时序等方面的验证,以确保实际系统的性能,最后,进行针对特定目标芯片的逻辑映射和编程下载等工作。整个过程包括设计准备、设计输入、设计处理和器件编程四个步骤以及相应的功能仿真、时序仿真和器件测试三个设计校验过程。现代数字系统的设计流程如图 1.2 所示。

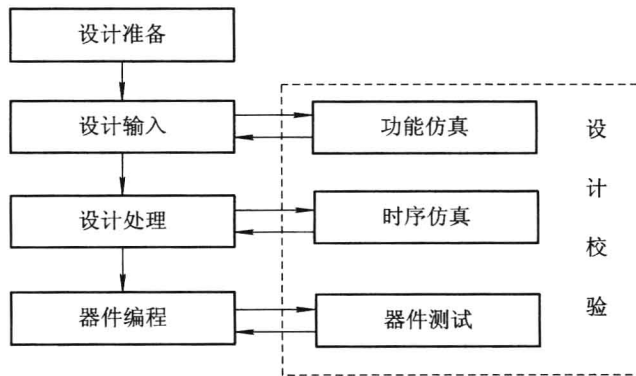


图 1.2 现代数字系统的设计流程

1. 设计准备

在设计之前,首先要进行方案论证、系统设计和器件选择等设计准备工作。设计者首先要根据任务要求,判明系统指标的可行性。系统的可行性要受到逻辑合理性、成本、开发条件、器件供应、设计员水平等方面的约束。若系统可行,则根据系统所完成的功能及复杂程度,对器件本身的资源和成本、工作速度及连线的可布性等方面进行权衡,选择合适的设计方案和合适的器件类型。

2. 设计输入

设计输入是设计者将所设计的系统或电路以 EDA 开发软件要求的某种形式表示出来,并送入计算机的过程。它根据 EDA 开发系统提供的一个电路逻辑的输入环境(如原理图、硬件描述语言(HDL)等形式)进行输入,这些方法可以单独构成,也可将多种手段组合来生成一个完整的设计。

输入软件在设计输入时,还会检查语法错误,并产生网表文件,供设计处理和设计校验使用。

3. 设计处理

设计处理是从设计输入文件到生成编程数据文件的编译过程,这是器件设计中的核心环节。设计处理是由编译软件自动完成的。设计处理的过程如下:

(1) 逻辑优化和综合。由软件化简逻辑,并把逻辑描述转变为最适合在器件中实现的形式。综合的目的是将多个模块化设计文件合并为一个网表文件,并使层次设计平面化。逻辑综合应施加合理的用户约束,以满足设计的要求。

(2) 映射。把设计分为多个适合用具体 PLD 器件内部逻辑资源实现的逻辑小块的形式。映射工作可以全部自动实现,也可以部分由用户控制,还可以全部由用户控制进行。

(3) 布局和布线。布局和布线工作是在设计检验通过以后由软件自动完成的, 它能以最优的方式对逻辑元件布局, 并准确地实现 PLD 器件内部逻辑元件间的互连。

(4) 生成编程数据文件。设计处理的最后一步是产生可供器件编程使用的数据文件。对 CPLD 器件而言, 产生熔丝图文件即 JDEC 文件, 对 FPGA 器件则生成位流数据文件。

4. 设计校验

设计校验过程是使用 EDA 开发软件对设计进行分析, 它包括功能仿真、时序仿真和器件测试。

功能仿真用于验证设计的逻辑功能, 通常是在设计输入完成之后, 选择具体器件进行编译之前进行的逻辑功能验证。功能仿真没有延时信息, 对于初步的逻辑功能检测非常方便。仿真结果将会生成报告文件和信号波形输出, 从中便可以观察到各个节点的信号变化。若发现错误, 则可返回设计输入中修改逻辑设计。

时序仿真是在选择了具体器件并完成布局、布线之后进行的快速时序检验, 可对设计性能作整体上的分析, 这也是与实际器件工作情况基本相同的仿真。由于不同器件的内部延时不一样, 不同的布局、布线方案也给延时造成不同的影响, 用户可以得到某一条或某一类路径的时延信息, 时序仿真也可给出所有路径的延时信息(又称延时仿真)。若设计的性能不能达到要求, 需找出影响性能的关键路径, 并返回延时信息, 修改约束文件, 对设计进行重新综合和布局布线, 如此重复多次直到满足设计要求为止。因此时序仿真对于分析时序关系, 评估设计的性能以及检查和消除竞争冒险等是非常有必要的。

直接进行功能仿真的优点是设计耗时短, 对硬件库和综合器没有任何要求, 尤其对于规模比较大的设计项目, 综合和布局布线在计算机运行耗时可观, 若每次修改都进行时序仿真, 显然会降低设计开发效率。通常的做法是: 首先进行功能仿真, 待确认设计文件满足设计要求的逻辑功能后再进行综合、布局布线和时序仿真, 把握设计项目在实际器件的工作情况。

5. 器件编程

编程是把系统设计的下载或配置文件, 通过编程电缆按一定的格式装入一个或多个 PLD 的编程存储单元, 用于定义 PLD 内部模块的逻辑功能以及它们的相互连接关系, 以便进行硬件调试和器件测试。

器件编程需要满足一定的条件, 如编程电压、编程时序和编程算法等。随着 PLD 集成度的不断提高, PLD 的编程日益复杂, PLD 的编程必须在开发系统的支持下才能完成。

器件在编程完毕之后, 对于具有边界扫描测试能力和在系统编程能力的器件来说, 系统测试起来就更加方便, 它可通过下载电缆下载测试数据来探测芯片的内部逻辑以诊断设计, 并能随时修改设计重新编程。

1.3 EDA 软件工具简介

EDA 软件在现代数字系统设计中占据了极其重要的位置。EDA 的核心是利用计算机完成电子设计全程自动化, 因此, 基于计算机环境的 EDA 软件的支持是必不可少的。

由于 EDA 的整个流程涉及不同的技术环节, 每一环节中必须有对应的软件包或专用

EDA 工具独立处理, 因此单个 EDA 工具往往只涉及 EDA 设计流程中的某一步骤。这里就以 EDA 设计流程中涉及的主要软件为 EDA 工具分类, 并加以简要介绍。EDA 软件工具大致可以分为如下五个模块:

- 模块一: 设计输入编辑器。
- 模块二: HDL 综合器。
- 模块三: 仿真器。
- 模块四: 适配器(或称布局、布线器)。
- 模块五: 下载器(或称编程器)。

现在也有将五个模块集成在一起的 EDA 开发软件, 如 Xilinx 公司的 ISE、Altera 公司的 Quartus II 软件。

1. 设计输入编辑器

优秀的 EDA 软件平台不仅集成了多种输入编辑器的设计输入表达方式, 如状态图输入方式、波形输入方式以及 HDL 的文本输入方式, 而且还提供了不同的设计平台之间的信息交流接口和一定数量的功能模块库供设计人员直接选用。设计者可以根据功能模块的具体情况灵活选用。下面介绍几种设计输入编辑器中较常用和较成熟的设计输入方式。

1) 原理图输入

原理图输入是 EDA 工具软件提供的最基本的设计方法。该方法是选用 EDA 软件提供的器件库资源, 并利用电路作图的方法, 进行相关的电气连接而构成相应的系统或满足某些特定功能的系统或新元件。这种方式大多用在对系统及各部分电路很熟悉的情况, 或在系统对时间特性要求较高的场合, 它的主要优点是容易实现仿真, 便于信号的观察和电路的调整。原理图设计方法直观、易学。但当系统功能较复杂时, 原理图输入方式效率低, 它适应于不太复杂的小系统和复杂系统的综合设计(与其它设计方法进行联合设计)。Xilinx ISE 设计软件中原理图编辑器的窗口如图 1.3 所示。

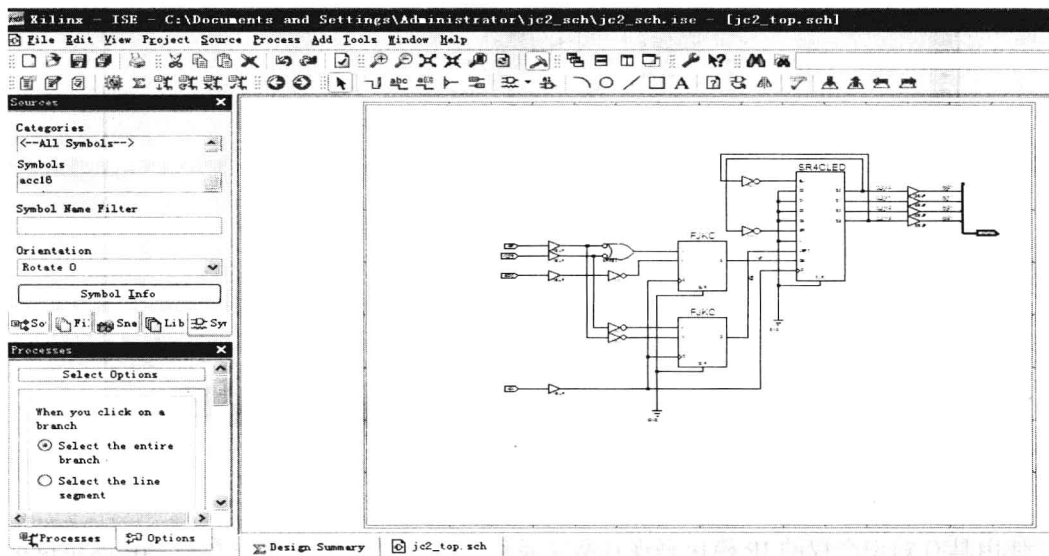


图 1.3 原理图编辑器的窗口