

XILINX | 赛灵思大学计划推荐用书

E-ELEMENTS

依元素推荐用书

清華

开发者书库



FPGA Modern Digital System Design

Based on Xilinx Programmable Logic Device and Vivado Platform

# FPGA现代数字系统设计

基于Xilinx可编程逻辑器件  
与Vivado平台

孟宪元 钱伟康◎编著

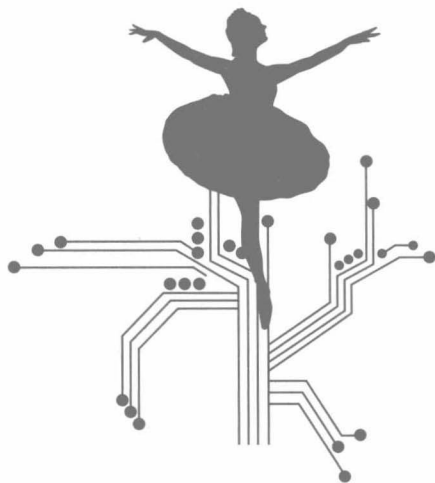
Meng Xianyuan Qian Weikang

清华大学出版社



清华

开发者书库



FPGA Modern Digital System Design

Based on Xilinx Programmable Logic Device and Vivado Platform

# FPGA现代数字系统设计

基于Xilinx可编程逻辑器件  
与Vivado平台

孟宪元 钱伟康◎编著

Meng Xianyuan Qian Weikang

清华大学出版社

北京

## 内 容 简 介

本书是以 Xilinx 公司全可编程 FPGA 和 SoC 为基础,针对最新的设计工具软件——Vivado 介绍 FPGA 设计理论与设计方法。

全书分为 8 章,包括现代数字系统设计技术概论、可编程逻辑器件、Verilog HDL 硬件描述语言、Vivado 设计流程、数字系统的设计与综合、基于 FPGA 的 DSP 系统设计、Zynq 嵌入式系统设计技术和 EGO1 综合性设计项目举例。各章都安排了针对性强的已验证过的设计实例,并附有 Verilog HDL 手册、EGO1 开发板资料,供师生在教学中选用。

本书可作为高等院校电子、通信、自动化、计算机等专业本科教学参考书,也可作为信息类专业研究生和数字系统设计人员的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

### 图书在版编目(CIP)数据

FPGA 现代数字系统设计:基于 Xilinx 可编程逻辑器件与 Vivado 平台/孟宪元,钱伟康编著. —北京:清华大学出版社,2019

(清华开发者书库)

ISBN 978-7-302-49913-8

I. ①F… II. ①孟… ②钱… III. ①可编程序逻辑器件—系统设计 IV. ①TP332.1

中国版本图书馆 CIP 数据核字(2018)第 055453 号

责任编辑:盛东亮

封面设计:李召霞

责任校对:时翠兰

责任印制:杨 艳

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质量反馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:三河市国英印务有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:32.25

字 数:785 千字

版 次:2019 年 4 月第 1 版

印 次:2019 年 4 月第 1 次印刷

定 价:79.00 元

产品编号:073614-01

# 前言

## PREFACE

由 Xilinx 公司发明的 FPGA 技术,按照摩尔定律已经历了 30 多年的发展历程,它的可编程特性使其成为电子产品设计和验证不可或缺的手段,在数字信号处理(DSP)系统和嵌入式系统等设计领域也得到日益广泛的应用。为了适应迅速发展的技术进步,培养出符合新时代要求的合格人才,近年来大学教育利用 FPGA 的可编程特性进行了广泛的探索,也取得了令人瞩目的成果,例如得到教育部认可的口袋实验板,以及贯穿式教育和工程实训等教改措施。

根据近期在多个大学进行数字系统和嵌入式系统工程实训的经历,结合国家对新型人才培养的要求,以及根据读者对利用 FPGA 设计现代数字系统的需求,选取目前流行的 FPGA 器件、设计工具和设计语言编写了此书。本书选用 Xilinx 公司 7 系列全可编程 FPGA 和 SoC,2017x 最新版本的 Vivado 设计工具,以及在许多大学得到广泛应用的依元素公司开发的 EGO1 开发板,作为学生的口袋实验板随身携带,不仅可以使设计项目在 FPGA 硬件上运行,也可以通过实验验证、理解和运用所学知识。

Verilog HDL 是设计者们喜爱的语言,与 VHDL 相比,更节省代码,更接近 C 语言,适合有 C 语言基础的读者学习,因此被业界广泛使用,也为本书所采用。

全书共 8 章。

第 1 章介绍现代数字系统设计概论,包括现代数字系统层次化的设计概念、多种描述方法和 IP、SoC 概念。本章使初学者对现代数字系统设计有一个整体的认识。

第 2 章介绍历代 FPGA 器件的结构特点、硬件资源和配置方法,为进一步的设计和 optimization 奠定必要的器件基础。

第 3 章介绍 Verilog HDL 的基本语法和设计实例,是全书的设计语言基础教程。

第 4 章介绍 Vivado 工具编程、仿真、综合和实现的设计流程,以及测试诊断工具和 IP 集成工具等。

第 5 章是比较深入的高级设计与综合技术,包括 Verilog HDL 的编程风格、综合优化、同步设计、高级综合与系统综合。本章是完成较复杂数字系统设计的重要设计知识与设计技巧。

第 6 章介绍在 FPGA 内实现 DSP 的方法,在 FPGA 内建 DSP 是现代 DSP 技术的最重要的发展方向。本章介绍在 FPGA 构建 DSP 的基本方法和 System Generator 设计工具,给出使用 System Generator 设计工具设计 FIR 数字滤波器的完整设计过程。

第 7 章介绍 Zynq 嵌入式系统设计技术,包括 Zynq 嵌入式处理器结构、AXI4 总线,利用 IP 模块在 Vivado 构建硬件,在 SDK 中编写软件,协同开发流程。

第 8 章给出三个综合设计实例。

为了方便读者,本书编写了 Verilog HDL 手册、EGO1 开发板资料和参考文献供查阅。

本书保留原书如下三个特点:

一是内容完整,包含设计理论、器件知识、设计语言、基本设计工具,还包括高级设计与综合技术、FPGA 内构建 DSP 的设计方法、SOPC 设计方法和综合设计实例。为初学者提供了完整的学习内容和丰富的参考资料。

二是注重读者的认识规律,由浅入深,循序渐进,既有深入的内容,又使初学者能很快入门;既有数字技术的理论知识,又有指导实践的实验实例。

三是书中涉及的所有程序均已经过调试,在教学过程中可以放心地使用和验证。

当然,现代数字系统设计涉及很广泛和深入的知识,不可能在一门课程中全部解决。我们希望能帮助初学者尽快入门,但更深入的研究和专门的设计知识需要在后续课程和设计实践中不断积累和完善。现代数字系统设计对理论和实践的综合要求都是比较高的,建议使用本教材的老师在介绍基本的设计基础后,尽量安排学生通过实验来发现和解决更多的问题,以提高实践能力。

本书是采用 Vivado 2017 版本的 FPGA 教材。选用本书作为教材的高校教师可以到网站 [http://e-elements.readthedocs.io/zh/ego1\\_v2.1/EGO1.html#id12](http://e-elements.readthedocs.io/zh/ego1_v2.1/EGO1.html#id12) 获取实验的详细资料及演示资料(需获准)。

本书由孟宪元和钱伟康教授共同完成编写和审稿。

由于 FPGA 技术发展迅速,设计工具的版本每年都有若干次更新,作者水平有限,编写时间仓促,书中的疏漏之处请读者予以指正。

感谢清华大学出版社对本书的出版给予的关心和支持!

孟宪元

2019年2月

# 目录

## CONTENTS

<b>第 1 章 现代数字系统设计概论</b> .....	1
1.1 概述 .....	1
1.2 数字系统的层次化结构 .....	4
1.2.1 开关电路级的基础——CMOS 反相器 .....	4
1.2.2 逻辑级的门电路.....	6
1.2.3 寄存器传输级的有限状态机.....	7
1.2.4 数字系统的系统级构成 .....	10
1.2.5 复杂系统的算法级设计 .....	11
1.3 数字系统设计的描述方法.....	13
1.3.1 原理图设计 .....	15
1.3.2 程序设计法 .....	15
1.3.3 IP 模块的使用 .....	16
1.3.4 基于模型的设计技术 .....	17
1.3.5 高层次综合——HLS 设计 .....	18
1.3.6 脚本设计技术 .....	19
1.4 IP 技术 .....	19
1.4.1 IP 知识产权模块 .....	19
1.4.2 IP 模块的种类与应用 .....	20
1.4.3 片上系统和 IP 核复用.....	20
1.5 全可编程 FPGA/SoC 实现智能化系统 .....	21
1.5.1 软件智能化和硬件最佳化 .....	22
1.5.2 在线可重构技术 .....	23
1.5.3 可配置加速堆栈 .....	25
本章小结 .....	26
习题 .....	26
<b>第 2 章 可编程逻辑器件</b> .....	27
2.1 概述.....	27
2.1.1 可编程逻辑器件概述 .....	27
2.1.2 可编程逻辑器件分类 .....	30
2.2 CPLD 的结构和工作原理 .....	32

2.2.1	简单可编程逻辑器件原理	32
2.2.2	CPLD 的结构和工作原理	38
2.3	FPGA 的结构和工作原理	41
2.3.1	SRAM-查找表类型	42
2.3.2	反熔丝多路开关类型	43
2.4	逻辑级 FPGA 的结构和工作原理	45
2.4.1	可编程逻辑	45
2.4.2	可编程互连线	50
2.4.3	可编程 I/O	52
2.5	系统级 FPGA 的结构和工作原理	53
2.5.1	片上存储器及接口	53
2.5.2	数字时钟管理	57
2.5.3	时钟资源	62
2.5.4	系统级 I/O	65
2.6	平台级 FPGA 的结构和工作原理	67
2.6.1	DSP 模块	67
2.6.2	高速串行接口	70
2.7	全可编程 FPGA 的特性和结构	73
2.7.1	采用统一的 7 系列架构	73
2.7.2	高性能和低功耗结合的工艺	74
2.8	ASIC 架构的 UltraScale 系列	76
2.8.1	UltraScale 架构	76
2.8.2	SSI 互连技术	79
2.9	FPGA 的配置	80
2.9.1	编程原理简介	81
2.9.2	编程模式	82
2.9.3	典型的配置电路	84
2.9.4	编程流程	85
2.9.5	部分重配置	86
	本章小结	88
	习题	88
<b>第 3 章</b>	<b>Verilog 硬件描述语言</b>	<b>90</b>
3.1	硬件描述语言概述	90
3.1.1	硬件描述语言特点	90
3.1.2	层次化设计	91
3.2	Verilog HDL 程序的基本结构	91
3.2.1	模块结构分析	93
3.2.2	模块的实例化	96
3.3	Verilog HDL 词法、数据类型和运算符	98

3.3.1	词法约定	98
3.3.2	数据类型	99
3.3.3	运算符	101
3.4	Verilog HDL 行为语句	104
3.4.1	赋值语句	105
3.4.2	顺序块和并行块语句	107
3.4.3	结构说明语句	108
3.4.4	条件语句	113
3.4.5	循环语句	116
3.4.6	系统任务和系统函数	118
3.4.7	编译预处理命令	123
3.4.8	Verilog HDL 可综合设计	125
3.5	Verilog HDL 设计举例	127
3.5.1	组合电路设计	127
3.5.2	时序电路设计	130
3.5.3	数字系统设计	131
3.5.4	数码管扫描显示电路	137
3.5.5	LED 通用异步收发电路设计	141
3.6	Testbench 文件与设计	149
	本章小结	155
	习题	156
<b>第 4 章</b>	<b>Vivado 设计工具</b>	<b>158</b>
4.1	Vivado 工具概述	158
4.1.1	单一的、共享的、可扩展的数据模型	159
4.1.2	标准化 XDC 约束文件——SDC	160
4.1.3	多维度分析布局器	161
4.1.4	IP 封装器、集成器和目录	163
4.1.5	Vivado HLS	164
4.1.6	其他特性	165
4.1.7	TCL 特性	167
4.1.8	Vivado 按键流程执行设计项目	168
4.2	Vivado 设计流程	175
4.2.1	创建工程	175
4.2.2	功能仿真	182
4.2.3	RTL 级分析	184
4.2.4	综合设计	186
4.2.5	分配引脚和时序	188
4.2.6	设计实现	200
4.2.7	生成 bit 文件	204



4.2.8	下载	204
4.3	产生 IP 集成器子系统设计	206
4.3.1	产生 IP 集成器模块设计	207
4.3.2	定制 IP	207
4.3.3	完成子系统设计	209
4.3.4	产生 IP 输出产品	210
4.3.5	例示 IP 到设计中	211
4.4	硬件诊断	211
4.4.1	设计诊断概述	212
4.4.2	Vivado 逻辑诊断 IP 核	213
4.4.3	HDL 例示法添加 ILA 核	215
4.4.4	系统内诊断 uart_led 设计	219
4.4.5	网表插入法添加诊断核	224
4.4.6	添加 VIO 诊断核	227
	本章小结	231
	习题	231
<b>第 5 章</b>	<b>数字系统的高级设计与综合</b>	<b>232</b>
5.1	Verilog 编程风格	232
5.1.1	逻辑推理	233
5.1.2	陷阱	237
5.1.3	设计组织	245
5.1.4	针对 Xilinx FPGA 的 HDL 编码	249
5.2	综合优化	251
5.2.1	速度与面积	252
5.2.2	资源共享	253
5.2.3	流水线、重新定时和寄存器平衡	255
5.2.4	有限状态机编译	259
5.3	数字系统的同步设计	261
5.3.1	同步设计基本原理	261
5.3.2	建立和保持时间	263
5.3.3	时序例外约束	267
5.3.4	同步设计中的异步问题	269
5.4	数字系统的综合	273
5.4.1	数字系统综合概述	273
5.4.2	系统级综合	275
5.4.3	高级综合	279
5.4.4	寄存器传输级综合	282
5.4.5	逻辑级综合	284
	本章小结	285

习题	285
<b>第 6 章 FPGA DSP 系统设计</b>	<b>287</b>
6.1 DSP 基础	287
6.1.1 DSP 的基本概念	287
6.1.2 FPGA 实现 DSP 的特点	290
6.2 DSP 硬核的结构与使用	292
6.2.1 输入和输出端口	293
6.2.2 DSP48E1 模块的操作	293
6.2.3 输入端口逻辑电路	295
6.2.4 输出端口逻辑	296
6.3 FPGA 设计 DSP 技术	298
6.3.1 浮点数与定点数的表示与转换	298
6.3.2 采样周期的设置	301
6.3.3 System Generator 模块	302
6.3.4 Black Box 模块	304
6.3.5 ModelSim 模块	308
6.3.6 Gateway In 模块和 Gateway Out 模块	308
6.3.7 Concat 模块、Convert 模块、Reinterpret 模块和 Slice 模块	309
6.3.8 模块通用属性	310
6.4 DSP48 实现 MAC	311
6.4.1 利用 Xilinx Blockset 设计 $12 \times 8$ MAC	311
6.4.2 利用 Simulink 仿真 $12 \times 8$ MAC	313
6.4.3 利用 System Generator Block 产生代码	313
6.4.4 实现 $12 \times 8$ MAC 设计	315
6.4.5 硬件协同仿真校验设计	316
6.5 设计 FIR 滤波器	318
6.5.1 产生 FIR 滤波器的系数	318
6.5.2 输入 FIR 滤波器系数	319
6.5.3 在 Simulink 中仿真 FIR 滤波器	321
6.5.4 实现 FIR 滤波器	322
6.5.5 连接演示板,通过 Simulink 仿真设计	323
6.6 设计 MAC FIR 滤波器	324
6.6.1 分析系数	326
6.6.2 添加控制逻辑并参数化	327
6.6.3 添加双口 RAM	328
6.6.4 在数据端口添加填充位和去填充位	330
6.6.5 完成 MAC FIR 设计	333
6.6.6 用各种信源测试设计	335
6.6.7 执行硬件在环路校验	336

6.7	Vivado HLS .....	340
6.7.1	高级综合的调度和装配.....	340
6.7.2	数据通道+控制器架构.....	341
6.7.3	理解 Vivado HLS .....	343
6.7.4	高级综合的优化方法.....	348
	本章小结.....	350
	习题.....	350
<b>第7章</b>	<b>嵌入式系统 Zynq 设计 .....</b>	<b>352</b>
7.1	Zynq 概述 .....	352
7.2	Zynq 设计入门 .....	353
7.2.1	Vivado 工程创建 .....	354
7.2.2	由 Vivado 创建 Zynq 嵌入式系统 .....	354
7.2.3	SDK 应用程序编写 .....	363
7.3	Zynq 嵌入式系统调试方法 .....	368
7.3.1	Vivado 硬件调试 .....	369
7.3.2	使用 SDK 进行 Zynq 调试 .....	374
7.4	调试 Linux 应用 .....	383
7.4.1	产生 SDK 软件工作空间 .....	383
7.4.2	启动超级终端.....	384
7.4.3	添加和诊断软件应用.....	385
	本章小结.....	393
	习题.....	393
<b>第8章</b>	<b>综合设计实例.....</b>	<b>394</b>
8.1	实例一：基于 VGA 接口的设计实例 .....	394
8.1.1	设计任务.....	394
8.1.2	原理分析与系统方案.....	395
8.2	实例二：PS/2 键盘编解码演示系统 .....	416
8.2.1	设计任务.....	416
8.2.2	原理分析与系统方案.....	416
8.2.3	设计实现.....	419
8.3	实例三：实现 SOPC 系统 .....	461
8.3.1	设计任务和方案.....	461
8.3.2	实验步骤.....	462
8.3.3	实验调试设备.....	475
	本章小结.....	476
	习题.....	477
<b>附录 A</b>	<b>EGO1 用户手册 .....</b>	<b>478</b>
<b>附录 B</b>	<b>Verilog HDL(IEEE 1364-2001)关键词表及说明 .....</b>	<b>496</b>
	<b>参考文献.....</b>	<b>504</b>



在介绍如何利用 FPGA 技术设计现代数字系统之前,先回顾一下数字技术发展的过程。在遵循摩尔定律的进程中克服了一系列的技术瓶颈后,现今,数字化技术正在成为当代社会的主要发展方向,以数字技术为手段,可将图、文、声、像等信息进行数字化的存储、处理和传播,使数字技术的产品渗透到社会生活的各个领域,进入数字化的时代。

而数字系统设计这门课程正是促使数字技术发展的基础,并随着社会进入数字化时代,数字化应用的需求又推动嵌入式系统的发展以及软硬件协同设计技术的提升,数字化设计的发展历程正相当于现代信息技术在产品设计领域中的应用不断发展的过程。

## 1.1 概述

微电子技术的高速发展是信息技术发展最重要的动力,信息技术蓬勃发展又带来集成电路行业的飞跃性进步,在新世纪,集成电路(IC)设计和制造技术都将会有一个前所未有的发展。

由于数字系统可被软件和硬件共同控制,因此数字系统远比模拟系统灵活。从产品开发时间和产品定义的角度来看,采用数字设计技术,将使开发产品的速度大大加快。同时,数字系统不容易受到干扰,没有信号失真、衰变等缺陷,可以采用标准化的逻辑部件来构成各种各样的数字系统,因此数字化产品备受欢迎。

一切数字化产品的核心,应该归功于基于半导体技术高度发展的专用集成电路(Application Specific Integrated Circuit,ASIC),归功于系统的单芯片集成技术——片上系统(System on Chip,SoC)。在半导体技术的推动下,数字系统的性能、功能、体积和功耗不仅得到显著改善,而且价格不断降低。数字系统的半导体技术含量不断增加。例如,硅芯片的价格将占据 DVD 播放机和机顶盒成本的 40%。如今,计算机、通信和其他功能之间的融合正以前所未有的速度向前发展,价格也逐渐向消费电子产品靠近。可以相信,在半导体技术的推动下,产品的功能、成本和开发时间将会有质的飞跃。

片上系统的出现和发展大大加速了人类社会的信息化进程,它已经成为信息产业乃至新世纪知识经济实现的关键技术基础之一。片上系统已在国际学术界和工业界受到广泛关注。片上系统的大量生产和应用,可以为工业界创造大量的商业机会,使工业界从容对待新世纪的技术挑战,同时也为研究领域对片上系统的设计方法学和测试方法学提出许多新的研究课题。

纵观信息产业的发展,一直遵从著名的“摩尔定律”,每18个月,单片集成电路的晶体管的数目就会翻一番,同样规格的芯片的成本便会降低一半。在过去的半个世纪中,这条定律一直有效,它表现为芯片制造技术发展迅速。在Intel公司联合创始人戈登·摩尔指出这个规律的1965年,每个芯片才容纳50个晶体管,到了1970年,每个芯片能够容纳1000个元器件,每个晶体管的价格降低90%。

1970年之后的30年时间里,工艺尺寸呈简单的几何比例缩小,从而使芯片上所有元器件越来越小,保证了器件规模稳速地增长,验证了摩尔定律的预测。

实际上,工艺尺寸按比例地缩减也经历了两个阶段,在工艺尺寸达到 $0.5\mu\text{m}$ 之前,工艺尺寸是按照电压值不变的“固定电压缩尺”进行的,由于电源电压保持5V不按比例缩减,尽管器件的工作速度因此得到提升,但是产生的最大问题是器件密度造成功耗的增加,因为功耗与电压 $V$ 的二次方以及工作频率 $f$ 和分布电容 $C$ 等因素成正比,即有关系式:

$$P = CV^2 f \quad (1.1)$$

所以,在工艺尺寸达到 $0.5\mu\text{m}$ 之后,为了解决功耗的问题,不得不按比例降低电源电压,进入“全电压缩尺”阶段,以芯片的内核电压为例, $0.35\mu\text{m}$ 工艺的电压为3.3V, $0.25\mu\text{m}$ 工艺的电压为2.5V,到90nm内核电压在1V左右。降低电压是减少功耗的有效措施但也受到限制,即过度降低电压将造成逻辑1和逻辑0之间的电压差别无法保持,以及抗噪能力减弱。

随着器件速度的提高,芯片内部电源电压的降低,满足各种应用要求的芯片之间的接口也进一步复杂化,需要为不同电源电压的集成电路之间提供兼容的I/O电平建立标准。

由联合电子器件工程协会(JEDEC)制定的标准接口电平包括小信号振幅和全信号振幅的变换。短线串联端接逻辑(SSTL)、高速收发逻辑(HSTL)和注射收发逻辑(GTL)都规定相对于参考电压的小幅度电压振幅。低压CMOS(LVCMOS)和低压晶体管-晶体管逻辑(LVTTL)利用无端接电源到地电压转换,在包括LVTTL和ECL(发射极耦合逻辑)在内的某些逻辑系列的信号电平最初是对双极型或biCMOS输出电路定义的,CMOS输出电路可以适应这些接口要求的兼容性。

在工艺线宽按比例缩减期间,集成电路除了ASIC之外,20世纪80年代还推出了现场可编程门阵列(FPGA)和复杂可编程逻辑器件(CPLD)的可编程器件。尤其是FPGA,它的可编程特性使得设计者可以在现场按照用户的需求设计数字系统,FPGA利用查找表实现逻辑功能,Fabless的代加工和依托SRAM工艺等实现可编程技术都是其重要的特性。而且早期的FPGA中,器件的规模受到集成的内部逻辑限制(Logic Limit),器件内部元件的密度不可能太大,仅包含基本的逻辑部件。而工艺线宽缩小后,能够集成的逻辑规模不断增加,要求的焊盘数量也显著增加,器件的规模变为受四周的外部焊盘限制(Pad Limit)。此时,被四周焊盘围起来的器件内部,在扩充逻辑部件的同时,还有空间可以集成块存储器、数字信号处理模块和微处理器等新的模块,它们的规模也随着工艺的变化而不断增加。

到了2000年,再单纯地做几何比例缩减不能解决功耗等问题,但是静态功耗与工艺的特性关系比较密切,所以与工艺有关的各种技术手段的发明使得器件的发展继续遵从摩尔定律。在工艺达到90nm时,采用了“应变硅”技术,纯硅在发生原子间力的应变后晶体结构

线性扩展,提高了功耗的容限;在 45nm 的工艺时,增加每个晶体管电容的分层堆积在硅上的新材料得到应用;到 22nm 工艺时,三栅极晶体管的出现保证了功耗和性能适应尺寸的缩小。

工艺线宽的不断缩小,产生的最大问题是器件本身的功耗和散热。

在降低功耗的措施上,考虑按性能要求一再提升工作频率不是出路,改为保持一定的时钟频率,限制微处理器执行计算机指令的速度;并将集成电路分成多个核,可降低每个核的功率和发热,出现了同构或异构的多核处理器芯片。

一个异构多处理系统由不同类型的多个单核或多核处理器构成,异构多核处理系统最简单的形式是由一个多核处理器和 GPU 组成。然而,现代科技让一颗芯片上的异构多处理系统包含以下模块:

- (1) 多核应用处理器(Multicore Applications Processors);
- (2) 多核图形处理器(Multicore Graphics Processors);
- (3) 多核实时处理器(Multicore Real-Time Processors);
- (4) 平台级管理单元(Platform Management Unit);
- (5) 配置和安全系统(Configuration and Security Unit);
- (6) 在 FPGA 可编程逻辑上实现特定多核处理器。

为什么多核处理器要与 FPGA 集成在一起来构成多处理器片上系统(MPSoC)的一部分?因为添加了 FPGA 之后,相当于给用户提供了一个可以定制的广义的处理器系统,以适应更广泛的应用要求,特别是需要利用硬件来实现的高速处理功能,利用此多核异构处理平台可实现自适应智能计算的多种应用。

20 世纪 90 年代,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,取得了巨大成功。在电子技术设计领域,可编程逻辑器件(如 CPLD、FPGA)的应用已得到广泛普及,这些器件为数字系统的设计带来了极大的灵活性。可编程逻辑器件可以通过软件编程对其硬件结构和工作方式进行重构,从而使得硬件的设计可以如同软件设计那样方便快捷。这一切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,促进了现代数字系统设计技术的迅速发展,实现了电子设计的自动化(Electronic Design Automation, EDA),它是在 20 世纪 90 年代初从计算机辅助设计(Computer Auxiliary Design, CAD)、计算机辅助制造(Computer Auxiliary Manufacture, CAM)、计算机辅助测试(Computer Auxiliary Test, CAT)和计算机辅助工程(Computer Auxiliary Engineering, CAE)的概念发展而来的。EDA 技术就是以计算机为工具,设计者在 EDA 软件平台上,用硬件描述语言(Hardware Description Language, HDL)完成设计文件,然后由计算机自动地完成逻辑编译、化简、分割、综合、优化、布局、布线和仿真,直至对于特定目标芯片的适配编译、逻辑映射和编程下载等工作。利用 EDA 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,可以通过计算机完成大量工作,并将电子产品从电路设计、性能分析到 IC 或 PCB 设计的整个过程通过计算机自动处理完成。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。

掌握科学的设计方法,恰当地选择设计工具是现代电子工程师最基本的素质。本章首先对基本数字系统的设计方法作初步介绍。

## 1.2 数字系统的层次化结构

为了进行复杂的数字系统设计,人们常采用分层次的方法,将系统设计的技术要求分别在行为域、结构域和物理域来考虑和描述,把问题由大化小、由复杂变简单,以便控制复杂度,减少每次处理的数量,也便于采用模块化设计。

在行为域,强调的是行为,它说明电路的功能,即电路输入/输出(I/O)的关系,但与该行为的实现无关,也可以说“如何实现”在行为域中被隐蔽起来了;在结构域,则对组成电路的各部件及部件间的拓扑连接关系进行描述,给出互连功能部件的层次关系;而在物理域,则要提供生产和制造物理实体所需要的信息,例如几何布局或拓扑约束等,即空间的物理布局和物理特性,没有任何功能部件的概念。如图 1-1 所示,常采用电子设计 Y 图表示各域相互之间的关系,对行为域、结构域、物理域的抽象层次一般按照结构描述从低至高分为五级:开关电路级、逻辑门级、寄存器传输级、硬件模块级(强调算法综合,又称算法级)和处理机系统级。

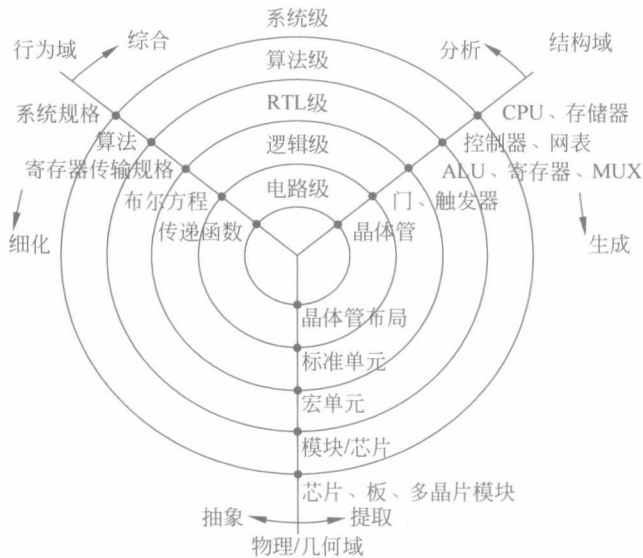


图 1-1 电子设计 Y 图

在实际应用时,可根据系统的复杂程度适当取舍。此时,各个域之间,又通过综合(Synthesis)与分析(Analysis)、抽象(Abstraction)与细化(Refinement)、生成(Generation)与提取(Extraction)分别实现行为域与结构域、物理域与行为域、结构域与物理域之间的转换。

本节主要依照层次结构分析数字系统的构成。

### 1.2.1 开关电路级的基础——CMOS 反相器

按照当前的半导体工艺技术,无论多么复杂的数字系统,其设计基础都是 CMOS 反相器。图 1-2(a)、图 1-2(b)、图 1-2(c)分别给出了 CMOS 反相器的电路图、输入/输出电压的特性曲线图和 CMOS 掩模图。这三张图实际上就是从结构域、行为域和物理域来描述反相器。

结构域的电路图说明反相器由上拉管和下拉管两部分组成,当输入电压从地电平跳到

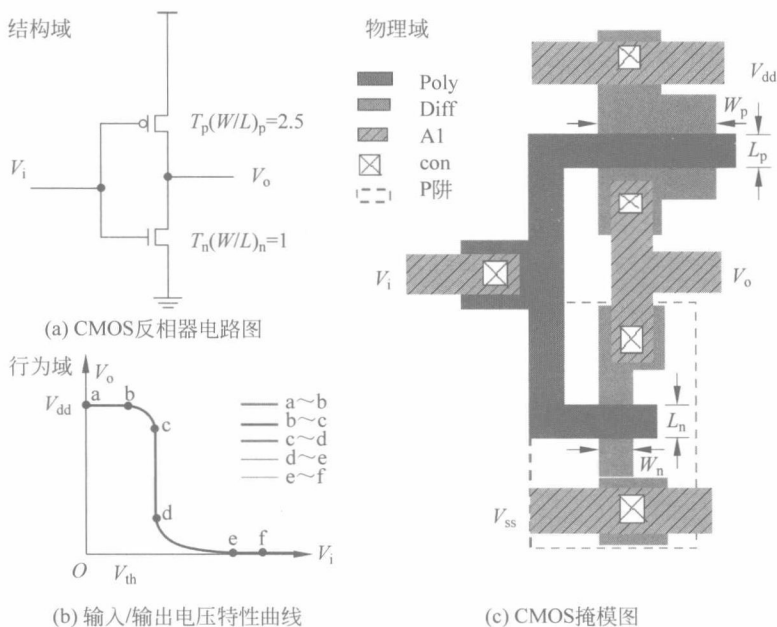


图 1-2 CMOS 反相器在各个域中的表示

电源电压时,下拉管导通;输出端分布电容上的电压要通过下拉管放电,从电源电压降为地电压,是输入信号的反相。当输入电压从电源电压跳到地电平时,上拉管导通,输出端分布电容上的电压要通过上拉管充电,从地电压上升到电源电压,也是输入信号的反相。反相器的晶体管工作于完全截止和充分导电两个极端状态,相当于开关的通断,所以又称开关管。

根据 CMOS 反相器的输入/输出特性可以得出行为域中反相器的工作特性。CMOS 反相器有以下优点:

- (1) 传输特性理想,过渡区比较陡;
- (2) 逻辑摆幅大,输出高电平  $V_{oh} = V_{dd}$ ,输出低电平  $V_{ol} = 0$ 。

物理域中上拉管和下拉管的宽度和长度,是指工艺在实现反相器时 n 管与 p 管扩散区的沟道宽度和长度,它们是反相器设计的主要参数,决定反向器的直流特性和交流特性。

CMOS 反相器的直流特性包括 nMOS 管的  $(W/L)_n$ 、pMOS 管的  $(W/L)_p$ 、功耗及直流输入与输出特性。

CMOS 反相器具有如下交流特性:

- (1) 输出电容  $C_{out} = C_{GDn} + C_{GDp} + C_{DBn} + C_{DBp} + C_{line} + C_{in}$ ;
- (2) 开关时间  $\tau_n = C_{out} / \beta_n (V_{dd} - V_{Tn})$ ,  $\tau_p = C_{out} / \beta_p (V_{dd} - |V_{Tp}|)$ ;
- (3) 一般阈值电平  $V_{th}$  位于电源  $V_{dd}$  的中点,即  $V_{th} = V_{dd} / 2$ ,因此噪声容限很大;
- (4) 只在状态转换为 b~e 段时两管才同时导通,才有电流通过,因此功耗很小;
- (5) CMOS 反相器是利用 p、n 管交替通、断来输出高、低电压的,而不像单管那样为保证  $V_{ol}$  足够低而确定 p、n 管的尺寸。

关于 CMOS 反相器的阈值电平  $V_{th}$ ,为了获得良好的噪声容限,应要求  $V_{th} = V_{dd} / 2$ ,假设  $\beta_n = \beta_p$  且  $V_{th} = |V_{tp}|$ ,则有  $V_{th} = V_{dd} / 2$ 。所以,为了满足  $\beta_n = \beta_p$ ,即为了提高电路的工作速度,一般取  $L_p = L_n = L_{min}$ ,则



$$W_p/W_n = \mu_n/\mu_p$$

即 p 管要比 n 管的栅极宽度大  $\mu_n/\mu_p$  倍。

### 1.2.2 逻辑级的门电路

CMOS 反相器是数字系统设计的基础,因为逻辑级的电路可以通过改造反相器来实现,而逻辑电路的设计也是在反相器设计的基础上完成的。如图 1-3 所示,下拉管串联形成与门,下拉管并联形成或门,下拉管的参数为不同的等效值  $\beta_{\text{eff}}$ 。

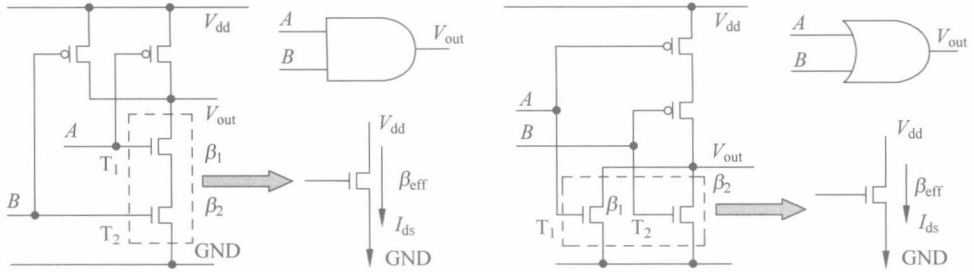


图 1-3 反相器下拉管串/并联构成与/或门

在一个组合逻辑电路中,为了使各种组合门电路之间能够很好地匹配,各个逻辑门的驱动能力都要与标准反相器相当。也就是说,在最坏工作条件下,各个逻辑门的驱动能力要与标准反相器的特性相同。

组合逻辑电路的设计可以被视为对标准反相器的改造。由于电子和空穴的迁移率有  $\mu_n \approx 2.5\mu_p$  的关系,所以 p 沟电阻约为 n 沟电阻的 2.5 倍。

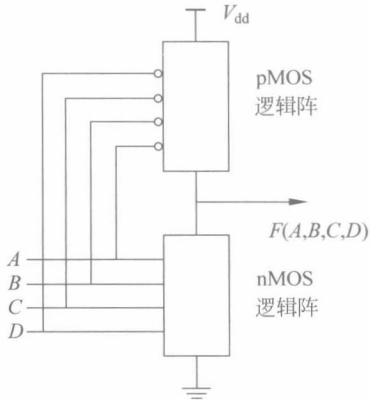


图 1-4 CMOS 组合逻辑

如图 1-4 所示,CMOS 组合逻辑形成规则如下:

- (1) nMOS 晶体管串联,实现 AND 运算;
- (2) nMOS 晶体管并联,实现 OR 运算;
- (3) 并联 nMOS 的分支,OR 各个分支的函数;
- (4) 逻辑函数串联是函数 AND 在一起;
- (5) 输出是 nMOS 逻辑的补;
- (6) pMOS 电路是 nMOS 电路准确的对偶。

在构成 CMOS 组合逻辑时,串联支路的影响必须考虑,因为 MOS 管的等效电阻也是串联的,使得时间常数加大,而影响性能,所以设计中要加宽沟道宽度,减少相应的等效电阻值。因此,复杂逻辑门电路的设计程序如下:

(1) 对电路估计输出分布电容  $C_{\text{out}}$ ,设计一个满足瞬态响应要求的反相器,分别计算  $(W/L)_{n \text{ inv}}$  和  $(W/L)_{p \text{ inv}}$ ;

(2) 构造 nMOS 逻辑块,考虑最大可能的串联晶体管数  $m$ ,选择每个器件是相同的,

$$(W/L)_n = m(W/L)_{n \text{ inv}}$$

(3) 构造 pMOS 逻辑块,考虑最大可能的串联晶体管数  $k$ ,选择每个器件是相同的,

$$(W/L)_p = k(W/L)_{p \text{ inv}}$$