



普通高等教育“十一五”国家级规划教材

国家精品课程教材

马建国 孟宪元 编著

FPGA

现代数字 系统设计



清华大学出版社



普通高等教育“十一五”国家级规划教材 国家精品课程教材

马建国 孟宪元 编著

FPGA

现代数字 系统设计

清华大学出版社
北京

内 容 简 介

由于电子产品设计技术趋向可编程的片上系统,本书以基础理论知识为纲,针对目前FPGA设计中的主流设计工具软件之一——Altera公司的Quartus II,介绍了FPGA设计理论与设计方法。

全书共8章,包括:现代数字系统设计概论、可编程逻辑器件、Verilog HDL语言、Quartus II软件、数字系统的高级设计与综合、基于FPGA的DSP设计、SOPC设计、设计实例。各章都安排了针对性强的思考题与练习题,并附有Verilog HDL手册、DE2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、计算机等专业本科生的教学参考书,也可以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/马建国,孟宪元编著. —北京:清华大学出版社,2010.4
ISBN 978-7-302-21539-4

I. ①F… II. ①马… ②孟… III. ①可程序逻辑器件—数字系统—系统设计
IV. ①TP332.1

中国版本图书馆CIP数据核字(2009)第217737号

责任编辑:陈志辉

责任校对:白 蕾

责任印制:何 芊

出版发行:清华大学出版社

<http://www.tup.com.cn>

社 总 机:010-62770175

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京市世界知识印刷厂

装 订 者:三河市溧源装订厂

经 销:全国新华书店

开 本:185×260 印 张:27.75 字 数:671千字

版 次:2010年4月第1版 印 次:2010年4月第1次印刷

印 数:1~3000

定 价:43.00元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:023849-01

前言

FOREWORD

花样繁多的电子产品给今天的生活增添了最靓丽的色彩,也对电子产品的设计者提出了很高的要求。因此,在大学里获得的电子系统设计知识成为毕业生求职的重要砝码。然而,面对眼花缭乱的设计工具和层出不穷的新型电子器件,大学的教学计划制定者们却伤透了脑筋:既要继承传统的基础理论和设计方法,又要加入最新的器件与设计工具,教学计划不堪重负,学生负担难以减轻。如何让初学者在最短的时间里,获得最必要和专门的设计知识,这是我们教育工作者的艰难任务。

根据长期在教学和科研第一线的体会,结合国家精品课程建设的经验,我们选取了目前主流之一的 Quartus II 设计工具和 Verilog HDL 设计语言。Altera 公司的 Quartus II 是我国高校普遍采用的 EDA 设计工具,其设计功能强大,界面友好,符合设计者使用习惯,更新换代快,支持最新的 FPGA 器件。更重要的,这与 Altera 公司的徐平波先生不遗余力地推广有很大关系。Altera 公司每年都组织有序的免费大学教师培训,为身处教学一线的青年教师提供了不断更新知识和广泛结识 EDA 设计与教育同行的机会,培养了一大批热爱 EDA 教学的教师;Altera 在全国范围内建立的数十个联合实验室构筑了良好的设计和开发平台,建立了良好的实验教学和研究的环境。Verilog HDL 语言是设计者们喜爱的语言,较 VHDL 语言节省代码,更接近 C 语言,适合有 C 语言基础者学习,并被业界广泛使用。

本书共分 8 章。第 1 章介绍了现代数字系统设计概论,内容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 章介绍了三代 FPGA 的结构原理、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 章介绍了 Verilog HDL 语言的基本语法和设计举例,是全书设计用语言的基础教程。第 4 章介绍了 Quartus II 软件的安装、设计、仿真、SignalTap II 测试工具和基于模块设计的 LogicLock 工具。第 5 章是比较深入的高级设计与综合技术,内容包括 Verilog HDL 的编程风格、综合优化、同步设计、高级综合与系统综合,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 章介绍了在 FPGA 内实现 DSP 的方法,在 FPGA 内构建 DSP 是现代 DSP 技术最重要的发展方向之一,本章介绍了在 FPGA 内构建 DSP 的基本方法和 DSP Builder 设计工具,给出了使用 DSP Builder 设计工具设计 FIR 数字滤波器的完整设计过程。第 7 章介绍了 SOPC 系统设计技术,内容包括 SOPC 系统结构、硬软件协同开发流程、Nios II 嵌入式处理器、Avalon 总线、外设组件、SOPC Builder

使用、Nios II 软件开发技术和设计实例。第 8 章介绍了三个数字系统综合设计实例,综合运用前述章节内容。

为了方便使用者,本书编写了 DE2 开发板资料、Verilog HDL 手册和英文缩略语供查阅。

本书有如下三个特点:

一是内容完整,既包含设计理论、器件知识、设计语言、基本设计工具,又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举例。为初学者提供了完整的学习内容和丰富的参考资料。

二是注重读者认知规律,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。

三是书中涉及的所有程序均经调试,在教学和工程中均可放心使用。

当然,现代数字系统设计中涉及很宽广和深入的知识,要在一门课程中全部掌握是不现实的。我们不奢望在本书解决 FPGA 设计中所遇到的所有问题,只希望让初学者尽快入门,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。数字系统设计是一门理论和实践综合要求都比较高的课程,建议使用本教材的老师在介绍了基本的设计基础后,尽量安排学生上机实验以发现和解决更多的问题。

本书由马建国教授和孟宪元教授担任主编,孟宪元参加了第 1 章、第 5 章的编写,罗亮参加了第 1 章、第 7 章、第 8 章的编写,刘桂华参加了第 2 章的编写,龙惠民、何燕玲参加了第 3 章的编写,熊丽英、权震华参加了第 4 章的编写,赵海龙参加了第 6 章的编写,权震华还编写了 Verilog HDL 手册、DE2 开发板资料,邢玲编写了英文缩略语表。本书由孟宪元、罗亮和马建国统稿。

本书的编写得到了教育部“国家十一五规划教材”基金的资助,得到了教育部“教育质量工程——国家精品课程”的资助,感谢 Altera 公司亚洲大学计划推广部徐平波先生和 Altera 公司联合实验室的支持,也感谢清华大学出版社对本书的出版给予的关心和支持!

微电子技术发展迅速,器件水平和设计工具更新换代速度很快,我们热忱希望与广大读者建立广泛的联系与合作,分享在现代数字系统设计中的体会与经验。

马建国 mjpg_my@263.net

孟宪元 mengxy@mail.tsinghua.edu.cn

2009 年 10 月

目录

CONTENTS

第 1 章 现代数字系统设计概论	1
1.1 概述	1
1.2 数字系统的层次化结构	2
1.2.1 开关电路级的基础——CMOS 反相器	2
1.2.2 逻辑级的门电路	4
1.2.3 寄存器传输级的有限状态机	5
1.2.4 数字系统的系统级构成	6
1.2.5 复杂系统的算法级设计	8
1.3 数字系统设计的描述方法	9
1.3.1 原理图设计	11
1.3.2 程序设计法	11
1.3.3 状态机设计	13
1.3.4 IP 模块使用	13
1.3.5 基于平台的设计方法	14
1.3.6 电子系统级——ESL 设计	15
1.4 IP 技术	16
1.4.1 IP 知识产权模块	16
1.4.2 IP 模块的种类与应用	17
1.4.3 片上系统和 IP 核复用	19
1.5 SOC 技术	21
1.5.1 SOC 基本概念	21
1.5.2 SOC 与 SOC 设计技术	21
1.5.3 SOC 芯片设计方法	23
1.5.4 SOC 平台设计方法	24
1.5.5 软/硬件协同设计方法	27
1.6 利用 FPGA 平台实现片上系统	31

1.6.1	平台级 FPGA 的特点	32
1.6.2	在线可“重构”技术	34
小结	35
习题	35
第 2 章	可编程逻辑器件	36
2.1	概述	36
2.1.1	可编程逻辑器件概述	36
2.1.2	可编程逻辑器件分类	39
2.2	CPLD 的结构和工作原理	41
2.2.1	简单可编程逻辑器件原理	41
2.2.2	CPLD 的结构和工作原理	47
2.3	逻辑级 FPGA 的结构和工作原理	50
2.3.1	可编程逻辑单元	51
2.3.2	可编程布线通道	59
2.3.3	可编程逻辑 I/O 单元	61
2.4	系统级 FPGA 的结构和工作原理	62
2.4.1	片上块 RAM 及接口	62
2.4.2	数字时钟管理	65
2.4.3	系统级 I/O	72
2.5	平台级 FPGA 的结构和工作原理	74
2.5.1	乘法器	75
2.5.2	DSP 硬核	76
2.5.3	高速串行接口	78
2.6	FPGA 的配置	82
2.6.1	FPGA 配置方式	82
2.6.2	FPGA 配置流程	86
小结	87
习题	87
第 3 章	Verilog 硬件描述语言	88
3.1	硬件描述语言概述	88
3.1.1	硬件描述语言特点	89
3.1.2	层次化设计	89
3.2	Verilog HDL 程序的基本结构	90
3.2.1	模块结构分析	92
3.2.2	模块的实例化	94
3.3	Verilog HDL 词法、数据类型和运算符	96
3.3.1	词法约定	96

3.3.2	数据类型	98
3.3.3	运算符	99
3.4	Verilog HDL 行为语句	102
3.4.1	赋值语句	103
3.4.2	顺序块和并行块语句	105
3.4.3	结构说明语句	106
3.4.4	条件语句	110
3.4.5	循环语句	113
3.4.6	系统任务和系统函数	115
3.4.7	编译预处理命令	119
3.4.8	Verilog HDL 可综合设计	122
3.5	Verilog HDL 设计举例	123
3.5.1	组合电路设计	123
3.5.2	时序电路设计	126
3.5.3	数字系统设计	128
3.6	Testbench 文件与设计	133
小结	138
习题	139
第 4 章	Quartus II 开发软件	141
4.1	Quartus II 图形用户界面和安装	141
4.1.1	Quartus II 图形用户界面	142
4.1.2	Quartus II 软件安装要求	143
4.1.3	Quartus II 软件安装	143
4.2	Quartus II 软件的设计流程	146
4.3	设计输入	147
4.3.1	创建工程	148
4.3.2	建立文本编辑文件	151
4.3.3	建立图形设计文件	152
4.3.4	建立存储器初始化文件	159
4.4	项目的编译实现	159
4.4.1	编辑模块与界面	159
4.4.2	编译器选项设置	160
4.4.3	引脚分配	162
4.4.4	实现与报告分析	163
4.5	项目的仿真实验	165
4.5.1	编辑激励波形	165
4.5.2	设置仿真器	168
4.5.3	基于 ModelSim 的仿真	169

4.6	硬件下载与验证	173
4.7	数字系统设计示例	175
4.8	SignalTap II 实时测试工具	179
4.8.1	SignalTap II 实时测试流程	179
4.8.2	SignalTap II 工具的使用	180
4.9	LogicLock 技术	184
4.9.1	LogicLock 技术简介	184
4.9.2	LogicLock 技术特性	184
4.9.3	LogicLock 区域	185
4.9.4	LogicLock 设计实例	186
	小结	193
	习题	193
第 5 章	数字系统的高级设计与综合	194
5.1	Verilog 编程风格	194
5.1.1	逻辑推理	195
5.1.2	陷阱	198
5.1.3	设计组织	207
5.2	综合优化	211
5.2.1	速度与面积	211
5.2.2	资源共享	212
5.2.3	流水线、重新定时和寄存器平衡	215
5.2.4	有限状态机编译	219
5.3	数字系统的同步设计	221
5.3.1	同步设计基本原理	221
5.3.2	建立和保持时间	223
5.3.3	同步设计中的异步问题	226
5.3.4	时序约束	230
5.4	数字系统的综合	232
5.4.1	数字系统综合概述	232
5.4.2	系统级综合	234
5.4.3	高级综合	238
5.4.4	寄存器传输级(RTL)综合	241
5.4.5	逻辑级综合	243
	小结	244
	习题	244
第 6 章	FPGA DSP 系统设计	246
6.1	DSP 的基本概念	246

6.2	FPGA 实现 DSP 的特点	248
6.3	DSP 硬核的结构与使用	250
6.4	基于模型的设计方法	257
6.5	DSP Builder 设计工具	261
6.5.1	DSP Builder 的安装	261
6.5.2	DSP Builder 支持的芯片和设计流程	262
6.5.3	DSP Builder 设计规则	263
6.5.4	DSP Builder 设计实例	266
6.5.5	利用 Black Box 设计定制模块	276
6.6	FIR 滤波器设计	279
6.6.1	滤波器系数的确定	279
6.6.2	建立 FIR 滤波器模型	281
6.6.3	Simulink 中仿真	284
6.6.4	硬件测试	285
	小结	289
	习题	289
第 7 章	可编程片上系统 SOPC 设计	291
7.1	SOPC 系统结构	291
7.2	SOPC 软硬件协同开发流程	294
7.2.1	嵌入式系统的特性	294
7.2.2	SOPC 设计流程	296
7.3	Nios II 处理器软核	298
7.3.1	Nios II 处理器简介	298
7.3.2	Nios II 处理器的架构	299
7.3.3	Nios II 处理器用户自定义指令	300
7.4	Avalon 总线	302
7.4.1	Avalon 总线简介	302
7.4.2	术语介绍	305
7.4.3	Avalon 总线信号	308
7.4.4	Avalon 总线传输模式	309
7.5	外设组件	310
7.5.1	常用组件简介	310
7.5.2	用户自定义外设组件	312
7.6	SOPC Builder 的使用	314
7.6.1	SOPC Builder 工具简介	314
7.6.2	SOPC Builder 设计流程	316
7.7	Nios II 系统软件架构	321
7.7.1	Nios II 处理器编程模型	321

7.7.2	软件架构	323
7.7.3	Nios II 系统的引导	327
7.8	Nios II 软件开发平台	328
7.8.1	Nios II 软件开发工具	328
7.8.2	Nios II 软件开发流程	330
小结	337
习题	337
第 8 章	综合设计实例	338
8.1	实例一:PS/2 键盘编解码演示系统	338
8.1.1	设计任务	338
8.1.2	原理分析与系统方案	339
8.1.3	设计实现	350
8.1.4	系统配置与测试	366
8.2	实例二:基于 VGA 接口的乒乓游戏系统	367
8.2.1	设计任务	367
8.2.2	原理分析与系统方案	367
8.2.3	设计实现	372
8.2.4	系统配置与测试	384
8.3	实例三:实现乐曲播放的 SOPC 系统	385
8.3.1	设计任务和方案	385
8.3.2	定制 PWM 外设组件	385
8.3.3	构建 SOPC 硬件系统	396
8.3.4	软件程序设计	402
8.3.5	硬件实现与测试	406
小结	409
习题	409
附录 A	Altera DE2 开发板说明	410
附录 B	Verilog HDL(IEEE 1364—2001)关键词表及说明	418
附录 C	英文缩写词	426
参考文献	432

第 1 章

现代数字系统设计概论

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进步,在 21 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。

纵观信息产业的发展,著名的“摩尔定律”指出,每 18 个月,单片集成电路的晶体管的数目就会翻一番。在过去 40 多年中,这条定律一直有效,它表现为芯片制造技术发展迅速,造成芯片设计技术一直落后并不断追赶前者,另一方面也使得芯片的价格下降,应用越来越广。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,因此备受欢迎。

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 ASIC(application specific integrated circuit),归功于系统的单芯片集成技术——SOC(system-on-a-chip)。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%。今天,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 SOC 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。

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,已得到广泛的普及,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而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——EDA(electronic design automation),它是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(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,电子设计工程师可以从概念、算法、协议等开始设计电子系统,大量工作可以通过计算机完成,并可以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。本章先对基本的数字系统的设计方法作初步的介绍。

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,即电路的输入/输出关系,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,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。对行为、结构、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、门级、寄存器级、硬件模块级和处理机级。在实际应用时,可根据系统的复杂程度适当舍取。此时人们常采用电子设计Y图来表示相互之间的关系,如图1-1所示。

各域之间,人们又分别通过综合(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。

本章先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,设计基础都是CMOS反相器,图1-2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掩膜图。这三个图实际上就是从结构域、行为域和物理域来描述反相器的。

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

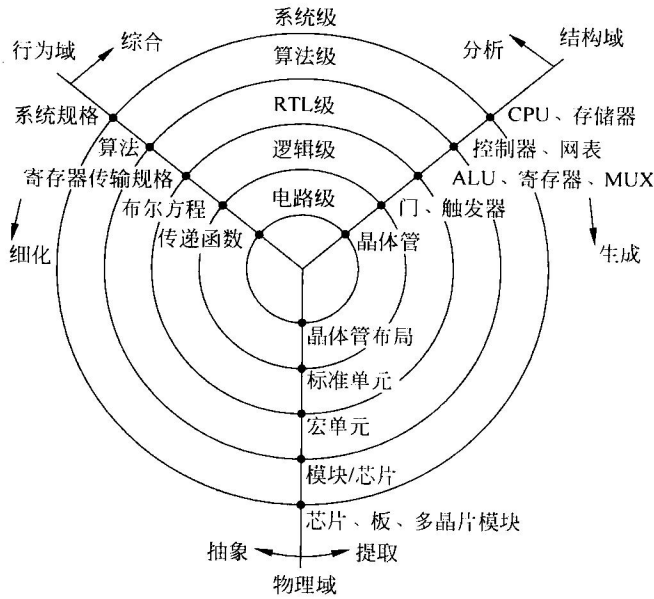


图 1-1 电子设计 Y 图

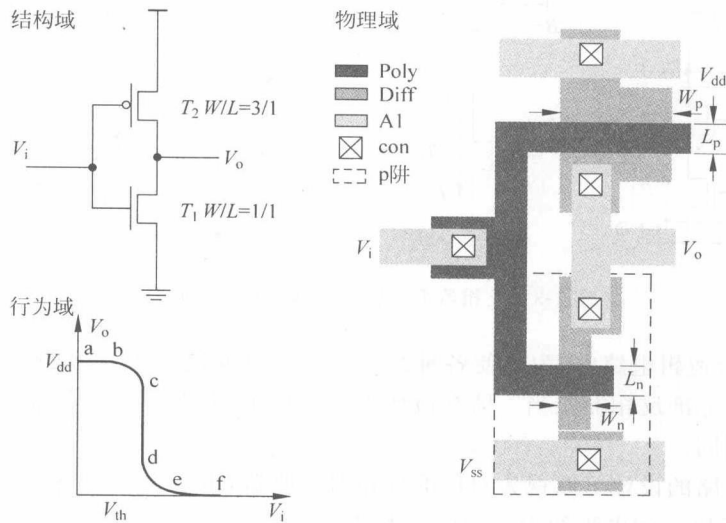


图 1-2 CMOS 反相器在各个域中的表示

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的两个极端状态,相当于开关的通断,所以又称开关闭。

由 CMOS 反相器的输入/输出特性可以得到行为域中反压器的工作特性,CMOS 反压器有以下优点:

- (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 $V_{ol} = 0$ 。

(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、断来获取输出高、低电压的, 而不像单管那样为保证 V_{oi} 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} , 为了有良好的噪声容限, 应要求 $V_{th} = V_{dd}/2$, 如果假设: $\beta_n = \beta_p$, $V_{th} = |V_{tp}|$, 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 管栅极宽度是 n 管栅极宽度的 μ_p/μ_n 倍。

上面提到的上拉管和下拉管的宽度和长度是指在物理域实现反相器时, 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, 决定反向器直流特性和交流特性^①。

1.2.2 逻辑级的门电路

为什么说 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来完成的, 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 β_{eff} 。

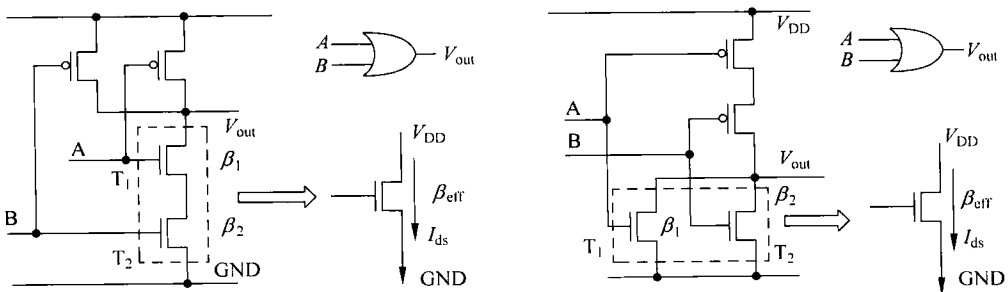


图 1-3 反相器下拉管串/并联构成与/或门

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。即在最坏工作条件下, 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。

CMOS 组合逻辑(如图 1-4 所示)形成规则为:

① CMOS 反相器直流特性包括:

nMOS 管的 $(W/L)_n$ 和 pMOS 管的 $(W/L)_p$

功耗及直流输入与输出特性

CMOS 反相器交流特性包括:

输出电容 $C_{out} = C_{GDn} + C_{GDp} + C_{DBn} + C_{DBp} + C_{line} + C_{in}$

开关时间: $\tau_n = C_{out}/\beta_n(V_{DD} - V_{Tn})$

$\tau_p = C_{out}/\beta_p(V_{DD} - |V_{Tp}|)$

规则 1: nMOS 晶体管串联, 实现 AND 运算。

规则 2: nMOS 晶体管并联, 实现 OR 运算。

规则 3: 并联 nMOS 的分支是 OR 各个分支的函数。

规则 4: 逻辑函数串联, 是函数 AND 串在一起。

规则 5: 输出是 nMOS 逻辑的补。

规则 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, 复杂逻辑门电路设计程序如下:

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

② 构造 nMOS 逻辑块, 考虑最大可能的串联晶体管数 m , 选择每个器件具有相同的 $(W/L)_n = m(W/L)_{n,inv}$

③ 构造 pMOS 逻辑块, 考虑最大可能的串联晶体管数 k , 选择每个器件具有相同的 $(W/L)_p = k(W/L)_{p,inv}$

一个数字系统不论它有多么复杂, 对其结构中不可能不包括的组合逻辑, 已有成熟的理论和方法有效地表达、变换和简化, 并可以方便地用积之和的形式来描述, 因此组合逻辑可以转换成两级与—或逻辑来实现。

所以, 组合逻辑也可以由一个可编程逻辑阵列 (PLA) 来实现, 它是将不规则的组合逻辑映射到规整的结构中, 如第 2 章的图 2-6 所示。

PLA 的规整结构中, 与阵列和或阵列二者都是可编程的, 特别适合半导体工艺中采用, 达到缩小面积的目的。实际上实现组合逻辑常常采用可编程阵列逻辑, 见第 2 章。

采用可编程逻辑器件 (PLD) 和可编程门阵列 (FPGA) 实现数字系统, 给系统的设计带来很多方便, 省去了掩膜图设计和投片等工作, 既节省费用, 又能使产品尽快面市, 尤其是选用 FPGA 进行数字系统设计已成为越来越明显的趋势。

1.2.3 寄存器传输级的有限状态机

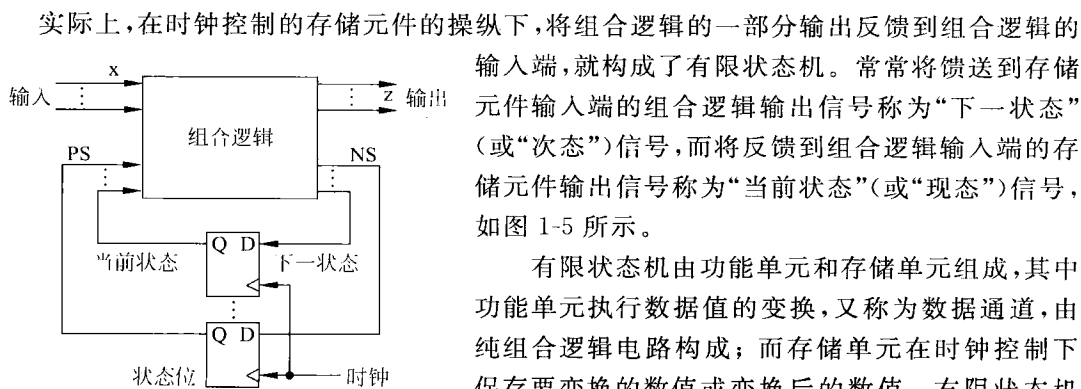


图 1-5 有限状态机构成

实际上, 在时钟控制的存储元件的操纵下, 将组合逻辑的一部分输出反馈到组合逻辑的输入端, 就构成了有限状态机。常常将馈送到存储元件输入端的组合逻辑输出信号称为“下一状态”(或“次态”)信号, 而将反馈到组合逻辑输入端的存储元件输出信号称为“当前状态”(或“现态”)信号, 如图 1-5 所示。

有限状态机由功能单元和存储单元组成, 其中功能单元执行数据值的变换, 又称为数据通道, 由纯组合逻辑电路构成; 而存储单元在时钟控制下保存要变换的数值或变换后的数值。有限状态机常由如图 1-6 所示的数据通道+控制器的结构来

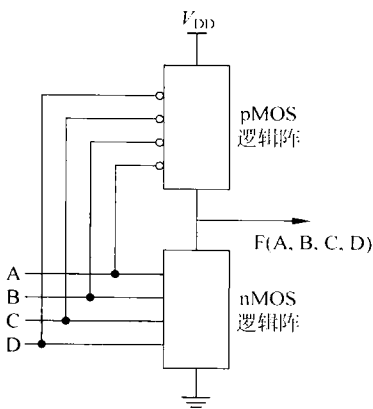


图 1-4 CMOS 组合逻辑

表示,称为带数据通道的有限状态机(FSMD)。将有限状态机划分成数据通道和控制器可以使系统结构清晰,系统设计简化,数据通道可以看作一个执行部件,用数据流图方便地描述,而控制单元可以看作一个有限状态机,用算法的状态转移图直观地描述。

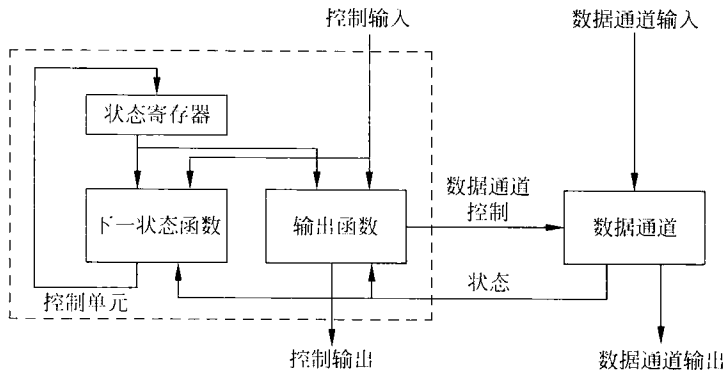


图 1-6 由数据通道+控制器构成的有限状态机

在 FSMD 中,实际上在数据通道引入一组在寄存器、寄存器堆和存储器中存储的整数或浮点变量,每个变量代替了上千的状态,即可以不用考虑数据位变化所决定的状态,从而免除了数据通道可观的状态数目,使得 FSMD 适用于复杂的数字系统设计,其中控制器的作用是确定输出函数和下一状态函数,数据通道的作用是进行数值的运算和变换。

大多数数据通道包括算术单元,如算术累加单元 ALU、加法器、乘法器、移位器等,把这些执行算法的部件集中起来可以构成处理器单元,数据通道要顺序完成的操作和执行的指令直接影响状态控制器的构成。存储器的加入和状态机的出现,使得数字系统发生了质的变化,在此基础上,才有可能发明以处理器为核心的现代数字计算机。

1.2.4 数字系统的系统级构成

在数字系统的设计中,有限状态机是一个重要的部件,位于层次结构中的多个有限状态机,每个有限状态机的“下一状态”信号和“当前状态”信号可以通过相应的通信方式相互作用,构成由嵌套的有限状态机组成的复杂数字系统,如图 1-7 所示。所以数字系统可以归结为一个分层次嵌套的有限状态机。

实际上,每个系统用一组过程、程序和函数来描述,这样一个系统描述可以分解为:由一组元件互连组成,这些元件包括处理器、存储器控制器、总线判决器、DMA 控制器和接口逻辑等,每个元件可以用一个或多个通信的 FSMD 实现,信号数目和信号之间在通信期间的相互关系称为通信协议,最常用的协议是请求和应答的握手协议。

在数字系统中利用总线的方式来分配数据信号和控制信号是比较方便的,对数据通道的综合是将数据流图的功能单元与数据通道的资源结合起来,并对其使用的资源进行调度。常常不会将数据流图的功能单元分割成一个接一个顺序执行的功能块,而是采用如图 1-8 所示的总线方式或寄存器方式,使功能单元按调度的要求执行多次操作来节省资源。采用插入寄存器的流水线结构来缩短关键路径,或者增加功能单元处理的并行度,二者都能提高系统的性能,但以增加硬件资源为代价。