

国外计算机科学经典教材

Digital Signal Processing with
Field Programmable Gate Arrays

数字信号处理的 FPGA 实现

Uwe Meyer-Baese

刘凌 胡永生

著
译

72



清华大学出版社

<http://www.tup.tsinghua.edu.cn>



数字信号处理的 FPGA 实现

Uwe Meyer-Baese 著

刘凌 胡永生 译

清华大学出版社

北京市版权局著作权合同登记号：图字：01-2002-4078

内 容 简 介

本书是一本有关最新数字信号处理(DSP)的专著。书中通过大量的程序实例,全面、精辟地介绍了利用现场可编程门阵列(FPGA)实现数字信号处理的方方面面。本书首先介绍了当前的FPGA技术、元器件和用于设计最新DSP系统工具的概况;接着主要介绍了计算机算法的概念、理论、FIR和IIR滤波器的实现、多级信号处理和傅立叶变换等内容;最后讲解了一些专用算法,如数论变换和密码术算法等。

本书内容详实、讲解深入浅出、实用性极强,可作为高等院校电子、电气以及相关专业的课程教材,也可供从事数字信号处理的专业人员参考。

Uwe Meyer-Baese: Digital Signal Processing with Field Programmable Gate Arrays

EISBN: 3-540-41341-3

Copyright© 2002 by Springer-Verlag Berlin Heidelberg New York.

Authorized translation from the English language edition published by Springer.

All rights reserved. For sale in the People's Republic of China only.

Chinese simplified language edition published by Tsinghua University Press.

本书中文简体字版由德国施普林格公司授权清华大学出版社出版。未经出版者书面许可,不得以任何方式复制或抄袭本书内容。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

数字信号处理的FPGA实现/(美)贝斯著;刘凌,胡永生译.—北京:清华大学出版社,2002

书名原文:Digital Signal Processing with Field Programmable Gate Arrays

ISBN 7-302-06035-5

I.数... II.①贝...②刘...③胡... III.数字信号—信号处理 IV.TN911.72

中国版本图书馆CIP数据核字(2002)第084551号

出 版 者:清华大学出版社(北京清华大学学研大厦,邮编 100084)

<http://www.tup.tsinghua.edu.cn>

责任编辑:李阳

封面设计:康博

版式设计:康博

印 刷 者:国防工业出版社印刷厂

发 行 者:新华书店总店北京发行所

开 本:787×1092 1/16 **印张:**23.25 **字数:**595千字

版 次:2003年1月第1版 2003年1月第1次印刷

书 号:ISBN 7-302-06035-5/TP·3600

印 数:0001~4000

定 价:48.00元

译 者 序

众所周知，信号与信号处理是信息科学中近十几年来发展最为迅速的学科之一。而 FPGA(Field Programmable Gate Array)正处于革命性数字信号处理的前沿。全新的 FPGA 系列正在越来越多地替代 ASIC 和 PDSP 用作前端数字信号处理的运算。FPGA 具有许多与 ASIC 相同的特点，例如：在规模、重量和功耗等方面都有所降低。而且吞吐量更高、能够更好地防止未经授权复制、元器件和开发成本进一步降低，开发时间也大大缩短。还具有在线路中可重复编程的特性。从而可以产生更为经济的设计。正如我们现在已经看到的，随着 FPGA 在数字信号处理中的大规模应用，正在日渐深入地影响我们的生产和生活，也必将在这一领域引起深刻的变革。本书的主旨就是讲述如何用 FPGA 实现数字信号处理。

简洁明了是作者 Uwe Meyer-baese 先生的风格，也是本书的风格。与大多数强调信号处理理论的著作相比，本书摒弃了传统的说教内容，更多地从应用的角度出发，注重如何用 FPGA 实现这些理论，并且致力于找到最适合解决问题的途径。在本书的开头，简要地介绍了当前 FPGA 技术的发展和用于设计的元器件以及设计 DSP 系统的工具的技术发展水平，之后就给出了频率合成器的设计研究示例。以示例教学的方式讲述了设计的编译步骤、仿真、性能评估、功耗估算和平面布置图。在以后的章节中系统地介绍了数制、算法和 FIR、IIR 滤波器的实现以及多级信号处理和各种傅立叶变换形式及其实现。最后一章简要介绍了当前的前沿课题，对其中一些方面感兴趣的读者可以在其指导下，作进一步的研究。

在本书中，共给出 33 个程序示例(正文中是 VHDL 代码，附录中有相应的 Verilog 代码)，还有 213 张插图和 57 份表格，均充分体现了 Uwe Meyer-baese 先生这部著作的特点。译者认为这也是除了学习本书中的知识以外的更大收获。

鉴于 FPGA 是一项全新的技术，特别是对于国内而言，相应的中文资料少之又少，又加之本人水平有限，本书的翻译不能够说是填补空白，只是想起到抛砖引玉的效果，只要能够对读者学习、了解和掌握用 FPGA 实现数字信号处理起到一定的作用也就心满意足了。在翻译中定有不妥之处，恳请读者批评指正。

另外，建议读者在阅读本书之前最好认真复习一下高等代数中多项式、行列式、矩阵、线性空间、线性变换和群、环、域等方面的概念和知识，能够熟练地完成各种矩阵的运算。并且最好是预先学习过信号与系统、数字信号处理和通信系统等相关课程，这样对您理解和掌握本书的知识要点会有极大的帮助，否则您会感到非常吃力，毕竟这并不是一本简单的科普教材。

译 者

2002 年 6 月于北京

前 言

正如可编程数字信号处理器(programmable digital signal processor, PDSP)在近 20 年前出现时的情形一样,如今,现场可编程门阵列(field-programmable gate array, FPGA)正处于革命性的数字信号处理技术的前沿。过去,前端的可编程数字信号处理(digital signal processing, DSP)算法,例如 FFT、FIR 和 IIR 滤波器,都是利用 ASIC 或者 PDSP 构建的,但现在大多为 FPGA 所代替。现代的 FPGA 系列都提供了支持以低系统开销、低成本实现高速乘-累加(Multiply-accumulate, MAC)超前进位链(Xilinx XC4000, Altera FLEX)的 DSP 算法^[1]。以前的 FPGA 系列大多面向 TTL “胶合逻辑”,没有 DSP 函数需要的大量的门数量。这些前端算法的有效实现就是本书要讲解的主要内容。

在 21 世纪初,我们就看到,两个可编程逻辑器件(programmable logic device, PLD)市场的领导者(Altera 和 Xilinx)都宣称获得了超过 10 亿美元的收入。在过去 10 年中, FPGA 一直保持以 20% 以上的速度稳步增长,超过 ASIC 和 PDSP 10% 以上。这源于 FPGA 具有许多与 ASIC 相同的特点,比如:在规模、重量和功耗等方面都降低了,同时还具有更高的吞吐量、更好的防止非授权复制的安全性、降低了元器件和开发的成本,并且还降低了电路板测试成本。此外,还声称具有优于 ASIC 的优势,例如:开发时间的缩短(快速的原型设计)、在线路中可重复编程的性质、更低的 NRE 成本,对于需求少于 1 000 个单元的解决方案而言,还可以产生更为经济的设计。与 PDSP 相比,典型的 FPGA 设计采用的都是并行操作,例如:实现多重乘-累加调用效率、消除零乘积项以及流水线操作,也就是每个 LE 都有一个寄存器,这样流水线操作就不再需要额外的资源了。

在 DSP 硬件设计领域中的另一个趋势就是从图形化设计入口转向硬件描述语言(hardware description language, HDL)。尽管很多 DSP 算法可以用“信号流程图”来描述,但是现在已经发现采用基于 HDL 的设计入口,其“代码复用”大大高于图形化设计入口的“代码复用”。这就对 HDL 设计工程师提出了更高的要求,我们已经在本科生的课堂上开设了采用 HDL 进行逻辑设计的课程^[2]。遗憾的是现在有两种流行的 HDL 语言。美国西海岸和亚洲倾向于采用 Verilog,而美国东海岸和欧洲则常使用 VHDL。对于用 FPGA 实现 DSP 而言,两种语言似乎都非常适用,尽管一些 VHDL 示例更容易阅读一些,这主要是因为 IEEE VHDL 1076-1987 和 1076-1993 标准中支持有符号算法和乘/除运算。这一差距有望在新的 Verilog IEEE 1364-1999 标准获得批准之后消失,这一标准也包括有符号算法。其他的约束条件可能包括个人的偏爱、EDA 库和工具包的可用性、数据类型、可读性、性能和采用 PLI 进行语言扩展,以及商业、企业和市场因素等^[3]。工具的提供商目前都支持这两种设计语言,而且这两种设计语言都适用于本书所采用的示例。

我们现在还是比较幸运的,因为不同来源的“基准”HDL 编译器基本上对于教学应用来说都是免费的。在本书中,我们就享受了这样的优惠。本书中所有给出的例子都是用 VHDL 和

HP 60515

Verilog 语言编写的, 可以很容易移植到其他适当的设计入口系统中。Xilinx 的“基础系列”、ModelTech 的 ModelSim 编译器和 Synopsys FC2 或者 FPGA 编译器都可以不需要在 VHDL 或者 Verilog 代码中做任何改动就能够运行。

本书结构是这样安排的。第 1 章首先简要介绍了当前的 FPGA 技术, 用于设计的元器件和工具的 DSP 系统的技术发展水平。还给出一个有关频率合成器的详细的案例研究, 包括编译步骤、仿真、性能评估、功耗估算和平面布置图。这一案例研究是后续章节中 30 多个设计示例的基础。第 2 章着眼于计算机算法方面, 包括可行的 DSP FPGA 算法的数制表达方式, 以及诸如加法器、乘法器或乘积和计算等的基本构造模块的实现。在这一章的结尾, 还讨论了对 FPGA 非常有用的两个计算机算法概念: 分布式算法(distributed arithmetic, DA)和 CORDIC 算法。第 3 章和第 4 章将研究 FIR 和 IIR 滤波器的理论和实现。我们将回顾如何确定滤波器的系数, 并讨论针对规模或者速度的可能最佳实现。第 5 章涵盖了许多应用于多级数字信号处理系统的概念, 例如: 抽取、插值和滤波器组, 在第 5 章结尾还讨论了采用双信道滤波器组实现小波处理器的多种可能性。第 6 章讨论了最重要的 DFT 和 FFT 算法的实现, 主要包括 Rader、chirp-z 和 Goertzel DFT 算法, 以及 Cooley-Tuckey、Good-Thomas 和 Winograd FFT 算法。在第 7 章介绍了更为专用的算法。与 PDSP 相比, 这有可能对改进 FPGA 实现具有更大的潜力。这些算法包括数论变换、密码术算法和错误校正, 以及通信系统的实现。附录包括 VHDL 和 Verilog 语言概述, 以及 Verilog HDL 描述的示例。

致谢

本书是在我在达姆施塔特(译者注: 德国西南部的一个城市, 位于法兰克福东南)技术大学讲授了 4 年的 FPGA 通信系统设计课程、我以前的(德文)书^[4, 5], 以及过去 10 年中我在达姆施塔特技术大学和位于盖恩斯维尔(译者注: 美国佛罗里达州中北部的城市)的佛罗里达大学指导的 60 多篇硕士研究生论文设计的基础上编写的。在此, 我要真挚地感谢所有在实验室和讨论会议上帮助我分析讨论的同仁们。这里, 我还要特别感谢: M. Acheroy、D. Achilles、F. Bock、C. Burrus、D. Chester、D. Childers、J. Conway、R. Crochiere、K. Damm、B. Delgutte、A. Dempster、C. Dick、P. Duhamel、A. Drolshagen、W. Endres、H. Eveking、S. Foo、R. Games、A. Garcia、O. Ghitza、B. Harvey、W. Hieberg、W. Jenkins、A. Laine、J. Mangen、J. Massey、J. McClellan、F. Ohl、S. Orr、R. Perrt、J. Ramirez、H. Scheich、P. Vaidyanathan、M. Vetterli、H. Walter 和 J. Wietzke。

我还要感谢我的学生们, 是他们花费无数小时来实现我的 FPGA 设计构想。特别是感谢: D. Abdolrahimi、E. Allmann、B. Annamaier、R. Bach、C. Brandt、M. Brauner、R. Bug、J. Burros、M. Burschel、H. Diehl、V. Dierkes、A. Dietrich、S. Dworak、W. Fieber、J. Guyot、T. Hattermann、T. Häuser、H. Hausmann、D. Herold、T. Heute、J. Hill、A. Hundt、R. Huthmann、T. Irmeler、M. Katzenberger、S. Kenne、S. Kerkmann、V. Kleipa、M. Koch、T. Krüger、H. Leitel、J. Maier、A. Noll、T. Podzimek、W. Praefcke、R. Resch、M. Rösch、C. Scheerer、R. Schimpf、B. Schlanske、J. Schleichert、H. Schmitt、P. Schreiner、T. Schubert、D. Schulz、A. Schuppert、O. Six、O. Spiess、O. Tamm、W. Trautmann、S. Ullrich、R. Watzel、H. Wech、S. Wolf、T. Wolf 和 F. Zahn。

此外，还要特别感谢我的妻子 Anke Meyer-Baese 博士和佛罗里达大学(位于盖恩斯维尔)的 J. Harris 博士和 Fred. Taylor 博士以及 Springer 的 Paul. DeGroot。是他们帮助我完成了英文修订本。

还要感谢 DAAD、DFG、欧洲航天局和 Max Kade Foundation 大力提供了经济支持。

如果您在阅读本书的过程中发现任何错误或者是有任何改进本书的建议，请您通过 Uwe.Meyer-Baese@ieee.org 或者是通过出版商与我联系。

Uwe Meyer-baese

塔拉哈西市 2001 年 5 月

目 录

第 1 章 绪论	1
1.1 数字信号处理(DSP)概述	1
1.2 FPGA 技术	2
1.2.1 按颗粒度分类	3
1.2.2 按技术分类	6
1.2.3 FPL 的基准	7
1.3 DSP 的技术要求	8
1.4 设计实现	10
1.4.1 FPGA 的结构	13
1.4.2 Altera EPF10K20RC240-4	15
1.4.3 案例研究: 频率合成器	17
1.5 练习	22
第 2 章 计算机算法	24
2.1 概述	24
2.2 数字表示法	24
2.2.1 定点数	25
2.2.2 非传统定点数	27
2.2.3 浮点数	36
2.3 二进制加法器	37
2.3.1 流水线加法器	39
2.3.2 模加法器	43
2.4 二进制乘法器	44
2.5 乘-累加器(Multiply-Accumulator, MAC)与乘积之和 (Sum of Product, SOP)	49
2.5.1 分布式算法基础	50
2.5.2 有符号的 DA 数制	52
2.5.3 改进的 DA 解决方案	54
2.6 利用 CORDIC 计算特殊函数	55
2.7 练习	63
第 3 章 有限脉冲响应(FIR)数字滤波器	66
3.1 数字滤波器	66
3.2 FIR 理论	66
3.2.1 具有转置结构的 FIR 滤波器	67



3.2.2	FIR 滤波器的对称性	70
3.2.3	线性相位 FIR 滤波器	71
3.3	设计 FIR 滤波器	72
3.3.1	直接窗函数设计方法	73
3.3.2	等同纹波设计方法	75
3.4	常系数 FIR 设计	76
3.4.1	直接 FIR 设计	77
3.4.2	具有转置结构的 FIR 滤波器	80
3.4.3	采用分布式算法的 FIR 滤波器	82
3.5	练习	97
第 4 章	无限脉冲响应(IIR)数字滤波器	99
4.1	IIR 理论	101
4.2	IIR 系数的计算	103
4.3	IIR 滤波器的实现	106
4.3.1	有限字长效应	109
4.3.2	滤波器增益系数的最优化	110
4.4	快速 IIR 滤波器	111
4.4.1	时域交叉	111
4.4.2	群集和分散预先考虑的流水线技术	114
4.4.3	IIR 抽取设计	115
4.4.4	并行处理	116
4.4.5	采用 RNS 的 IIR 设计	119
4.5	练习	119
第 5 章	多级信号处理	121
5.1	抽取和插值	121
5.1.1	Noble 恒等式	123
5.1.2	用有理数因子进行采样速率转换	124
5.2	多相分解	124
5.2.1	递归 IIR 抽取器	128
5.2.2	快行 FIR 滤波器	129
5.3	Hogenuer CIC 滤波器	131
5.3.1	单级 CIC 案例研究	132
5.3.2	多级 CIC 滤波器理论	134
5.3.3	幅值与混叠畸变	139
5.3.4	Hogenuer “剪除”理论	140
5.3.5	CIC RNS 设计	145
5.4	多级抽取器	147

5.5	作为通频带抽取器的频率采样滤波器	149
5.6	滤波器组	152
5.6.1	均匀 DFT 滤波器组	153
5.6.2	双信道滤波器组	156
5.7	小波分析	169
5.8	练习	175
第 6 章	傅立叶变换	178
6.1	离散傅立叶变换算法	179
6.1.1	用 DFT 近似傅立叶变换	179
6.1.2	DFT 的属性	180
6.1.3	Goertzel 算法	183
6.1.4	Bluestein Chirp-z 变换	183
6.1.5	Rader 算法	186
6.1.6	Winograd DFT 算法	191
6.2	快速傅立叶变换(Fast Fourier Transform, FFT)算法	193
6.2.1	Cooley-Tukey FFT 算法	194
6.2.2	Good-Thomas FFT 算法	205
6.2.3	Winograd FFT 算法	207
6.2.4	DFT 和 FFT 算法的比较	210
6.3	傅立叶相关的变换	212
6.3.1	利用 DFT 计算 DCT	213
6.3.2	快速直接 DCT 实现	214
6.4	练习	215
第 7 章	前沿课题	220
7.1	矩形变换和数论变换	220
7.1.1	算术模 $2^{b \pm 1}$	222
7.1.2	采用 NTT 的高效卷积	223
7.1.3	采用 NTT 的快速卷积	223
7.1.4	NTT 的多维索引映射和 Agarwal-Burrus NTT	227
7.1.5	用 NTT 计算 DFT 矩阵	229
7.1.6	NTT 的索引映射	230
7.1.7	用矩形变换计算 DFT	232
7.2	差错控制和加密技术	233
7.2.1	源自编码理论的基本概念	234
7.2.2	分组码	238
7.2.3	卷积码	242
7.2.4	FPGA 的加密技术算法	249



7.3 调制和解调	263
7.3.1 基本的调制概念	263
7.3.2 非相干解调	267
7.3.3 相干解调	272
7.4 练习	279
附录 A Verilog 源代码	283
附录 B VHDL 和 Verilog 编码	327
B.1 示例列表	329
B.2 参数化的模块库(LPM)	330
B.2.1 参数化的触发器兆函数(lpm_ff)	331
B.2.2 参数化的加法器/减法器兆函数(lpm_add_sub)	333
B.2.3 参数化的乘法器兆函数(lpm_mult)	337
B.2.4 参数化的 ROM 兆函数(lpm_rom)	340
附录 C 术语汇编	343
参考文献	349

第1章 绪 论

本章概述将要在本书中研究的算法和技术。首先简要介绍一下数字信号处理技术，然后重点讨论 FPGA 技术。最后研究 Altera EPF10K20 芯片和一个包括芯片合成、时序分析、平面布置图和功耗分析的较大规模设计示例。

1.1 数字信号处理(DSP)概述

长期以来，信号处理技术一直用于转换或产生模拟或数字信号。其中最为频繁应用的领域就是信号的滤波，我们将在第 3 章和第 4 章讨论这一问题。此外，从数字通信、语音、音频和生物医学信号处理到检测仪器仪表和机器人技术等许多领域中，都广泛地应用了数字信号处理技术。表 1-1 给出了 DSP 技术的一些应用概况^[6]。

表 1-1 数字信号处理的应用

应用领域	DSP 算法
通用领域	滤波和卷积、自适应滤波、检测和校准、谱估计和傅立叶变换
语音处理	编码和解码、加密和解密、语音识别和合成、扬声器识别、回波消除、人造耳蜗植入的信号处理
音频处理	hi-fi 编码和解码、噪声消除、音频平衡、环境声学仿真、混频和编辑、声音合成
图像处理	压缩和解压缩、旋转、图像传输与分解、图像识别、图像增强、人造视网膜植入的信号处理
信息系统	语音信箱、传真、调制解调器、蜂窝移动电话、调制/解调、线路均衡器、数据加密和解密、数字通信和局域网、延拓频谱技术、无线局域网、广播和电视、生物医学信号处理
控制	伺服控制、磁盘控制、打印机控制、发动机控制、定向和导航、振动控制、动力系统监控器、自动化仪器仪表
仪表设备	波束成型、波形发生器、瞬态分析、稳态分析、科学仪器设备、雷达和声纳

数字信号处理(digital signal processing, DSP)已经发展成为一项成熟的技术，并且在许多应用领域逐步代替了传统的模拟信号处理系统。DSP 系统具有几项优势，例如：元器件对温度变化、老化以及对容许偏差的不敏感性。在过去，模拟芯片设计可以生产出越来越小的小片尺寸，可是发展到今天，随着现代亚微米设计所带来的噪声，使得数字设计在集成度方面可以比模拟

设计做得更好。这些产品就是紧凑的、低功耗并且是低成本的数字设计。

有两个事件加速了 DSP 技术的发展。其一是 Cooley 和 Tuckey(1965 年)对一种计算离散傅立叶变换(Discrete Fourier Transform, DFT)的有效算法的解密。我们将在第 6 章详细讨论这类算法。另一个里程碑就是可编程数字信号处理器(programmable digital signal processor, PDSP)在 20 世纪 70 年代后期的引入。这种 PDSP 能够在仅仅一个时钟周期内完成(定点数)“乘-累加”的计算,与同一时代“冯·诺伊曼(Von Neuman)”式微处理器为基础的系统相比较而言,有着本质上的改进。现代的 PDSP 可以包含更为复杂的功能,例如:浮点数乘法器、筒状移位器、存储体以及零架空的 A/D 和 D/A 转换器接口。EDN 每年都要出版一份有关可用的 PDSP 的详细综述^[7]。图 1-1 给出了一个依靠 PDSP 来实现模拟系统的典型应用。在研究了 FPGA 的体系结构之后,我们将在 1.2.1 节和第 2 章中返回头来研究 PDSP。

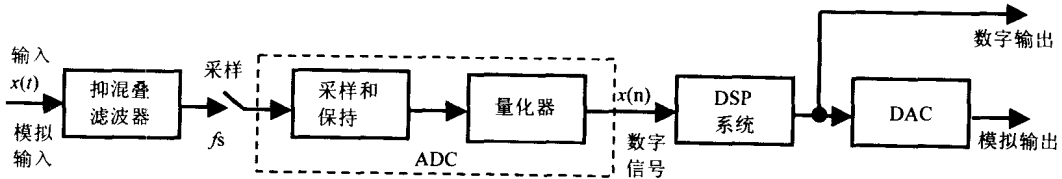
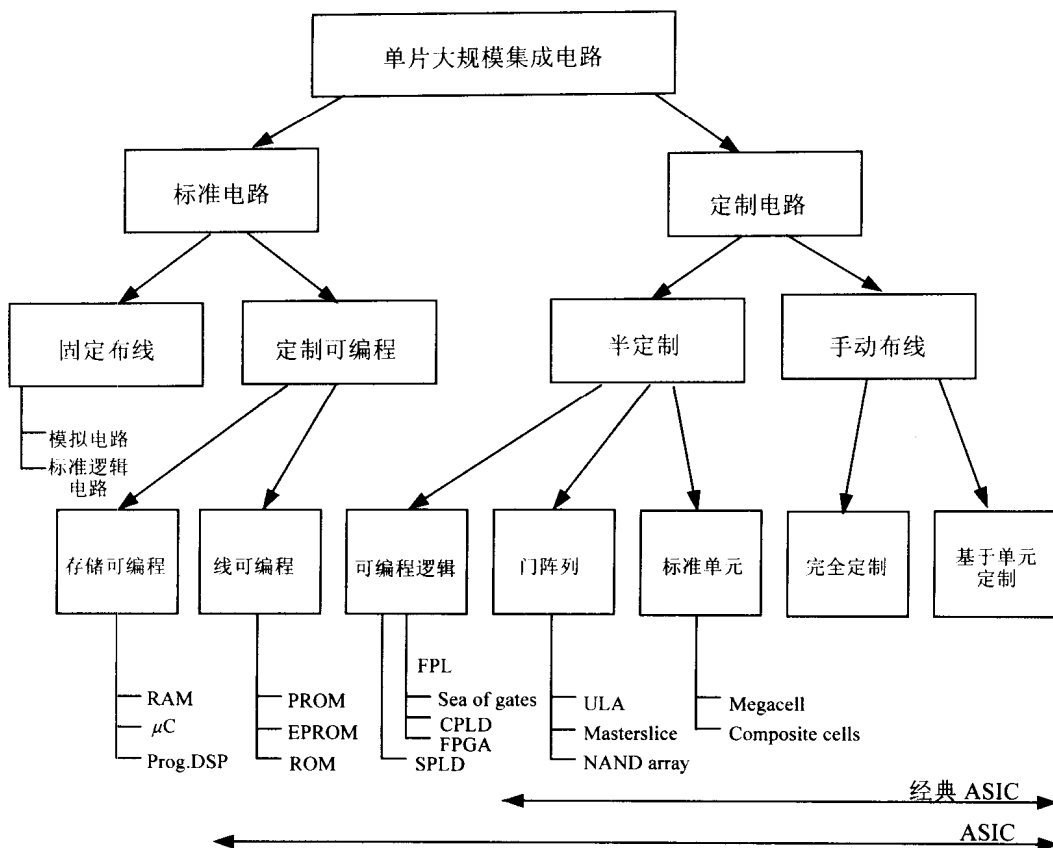


图 1-1 一个典型的 DSP 应用示例

1.2 FPGA 技术

VLSI(Very Large Scale Integration, 超大规模集成)电路可以如图 1-2 所示进行分类。FPGA 是一类称为现场可编程逻辑(field-programmable logic, FPL)器件中的一员。FPL 被定义为含有现场可反复使用的小规模逻辑模块和单元的可编程器件¹。鉴于 FPGA 是特定用途的集成电路,所以 FPGA 被认为是一种专用集成电路(application specific integrated circuit, ASIC)技术。但是,通常设计 ASIC 类电路需要额外的半导体处理步骤,而 FPL 是不需要这些步骤的。这些额外步骤能够提供更高级别的、更高性能的 ASIC,但同时也增加了不可重复的工程成本(non-reoccurring engineering, NRE)。另一方面,门阵列通常是由“与非门之海”构成,后者的功能是在“网络表”中提供的。在整个制造过程中都要使用这一网络表,以便获得最终金属层明显的清晰度。但是,可编程门阵列解决方案的设计者可以完全控制设计的实现过程,而不需要任何实际的集成电路制造设备或者因为后者而延缓设计进度。

注 1: Xilinx 称之为可配置的逻辑模块(configurable logic block, CLB), 而 Altera 称之为逻辑单元(logic cell, LC)或者逻辑元件(logic element, LE)。

图 1-2 VLSI 电路的分类(©1995 年, VDI 出版社^[41])

1.2.1 按颗粒度分类

逻辑模块规模与元器件的颗粒度相关,而元器件的颗粒度又与模块之间需要完成的布线(路由通道)工作量相关。3 种常见的不同颗粒度分类如下:

- 小颗粒度(Pilkington 或者“门海(sea of gates)”结构)
- 中等颗粒度(FPGA)
- 大颗粒度(CPLD)

1. 小颗粒度元器件

由 Pilkington 半导体公司供应的小颗粒度元器件最初得到 Plessey 公司的认可,然后是 Motorola 公司的认可。基本逻辑单元包括一个单一与非门和一个锁存器(请参阅图 1-3)。由于采用与非门可以实现任何二进制逻辑函数(请参阅练习 1.1),所以与非门被称为通用函数。这一技术连同已经被认可的逻辑合成工具(例如 ESPRESSO)一起,还应用于门阵列的设计之中。在门阵列的与非门之间布线是采用额外的金属层来实现的。但对于可编程的结构来讲,这就成了一个瓶颈,因为与已经实现的逻辑函数相比,它对布线资源的利用率非常高。此外,构建一个简单的 DSP 对象就需要大量的与非门。例如:一个高速 4 位加法器就要用掉大约 130 个与非门。这使得小颗粒度技术在实现大多数 DSP 算法时并没有什么吸引力。

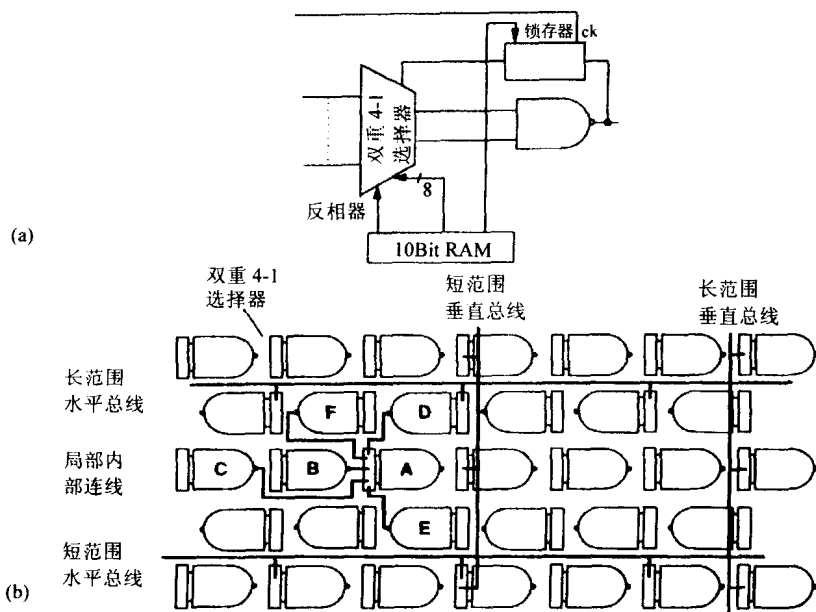


图 1-3 具有 10K 个与非逻辑模块的 Plessey ERA60100 结构^[8]

(a) 基本逻辑模块 (b) 布线结构(© 1990 Plessey)

2. 中等颗粒度元器件

最为常见的 FPGA 结构如图 1-4(a)所示。图 1-5 给出了一个当前中等颗粒度 FPGA 元器件的具体示例。具有代表性的基本逻辑模块是小规模的表(例如: Xilinx 的 XC2k-4k, 具有 4 位到 5 位的输入表, 1 位或者 2 位的输出)或者由专用的多路复用器(multiplexer, MPX)逻辑来实现, 例如: 在 Actel 的 ACT-2 元器件中的所使用的 MPX^[9]。布线通道的选择范围是从短到长。带有触发器的可编程 I/O 模块就附在元器件的物理边缘。

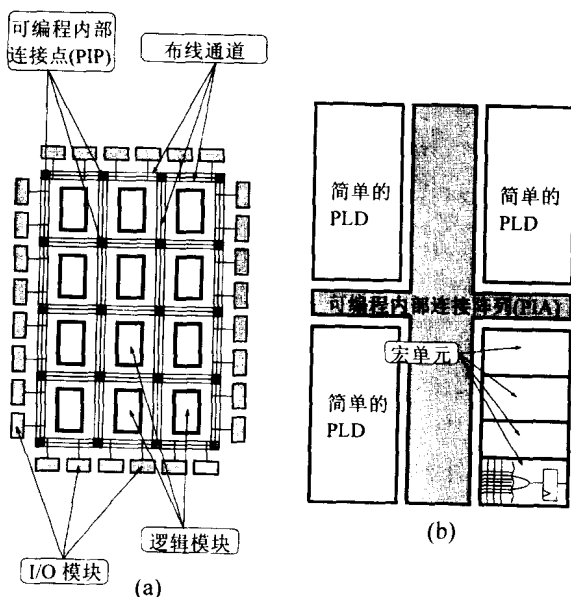


图 1-4 (a) FPGA 和(b) CPLD 的结构(©1995 年, VDI 出版社^[4])

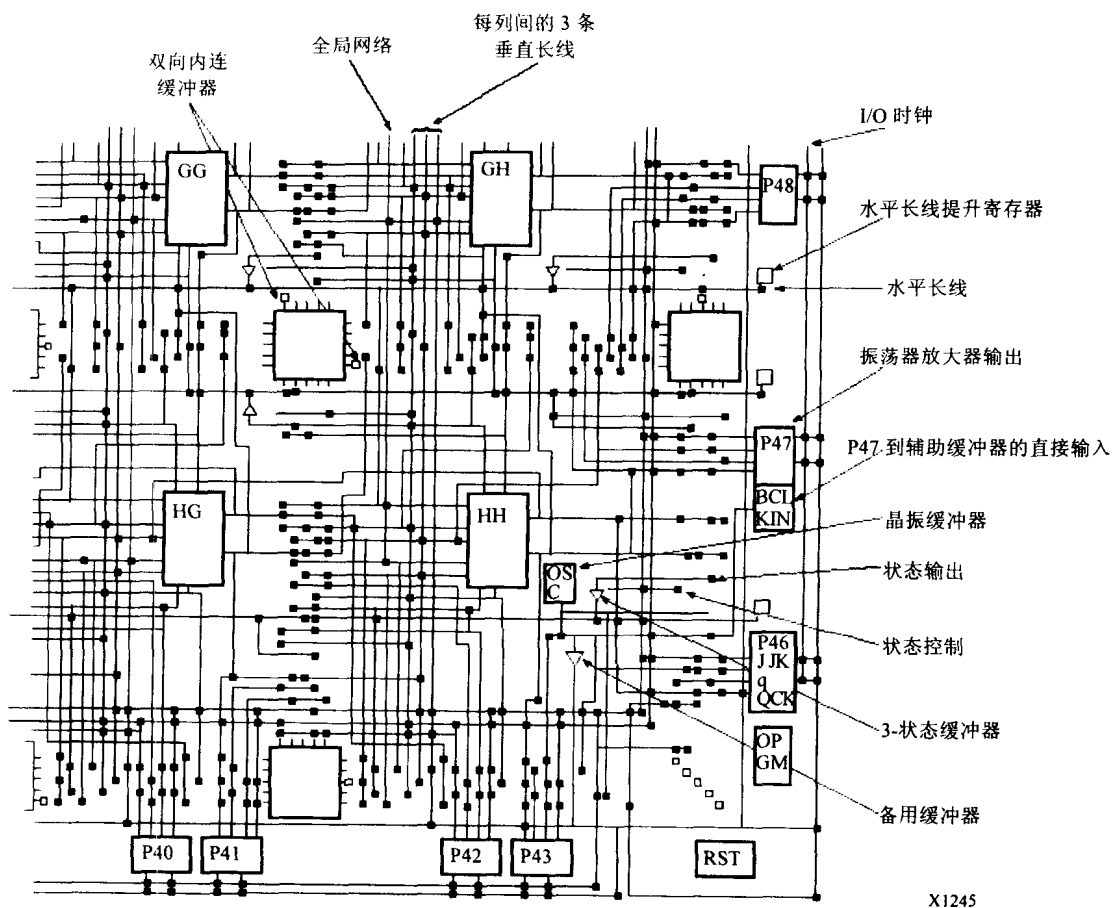


图 1-5 中等颗粒度元器件的示例(©1993 年, Xilinx)

3. 大颗粒度元器件

在图 1-4(b)中给出了大颗粒度元器件的特性, 诸如复杂的可编程逻辑元器件(complex programmable logic devices, CPLD)。这些复杂的可编程逻辑元器件(CPLD)可以定义成是由简单可编程逻辑元器件(simple programmable logic devices, SPLD)组合而成的, 例如: 如图 1-6 所示的传统 GAL16V8 芯片。这类 SPLD 芯片由一个充当与/非阵列的可编程逻辑阵列和一个通用 I/O 逻辑模块组成。通常, CPLD 中的 SPLD 具有 8 到 10 个输入端, 3 到 4 个输出端, 并且支持大约 20 个乘积项。在这些 SPLD 模块之间的宽带总线(Altera 称之为可编程内连阵列, (programmable interconnect arrays, PIAs))上有可能存在短暂的延迟。通过将总线与固定的 SPLD 时限结合起来就能够提供与 CPLD 之间可预先计算的短暂的管脚到管脚之间的延迟。

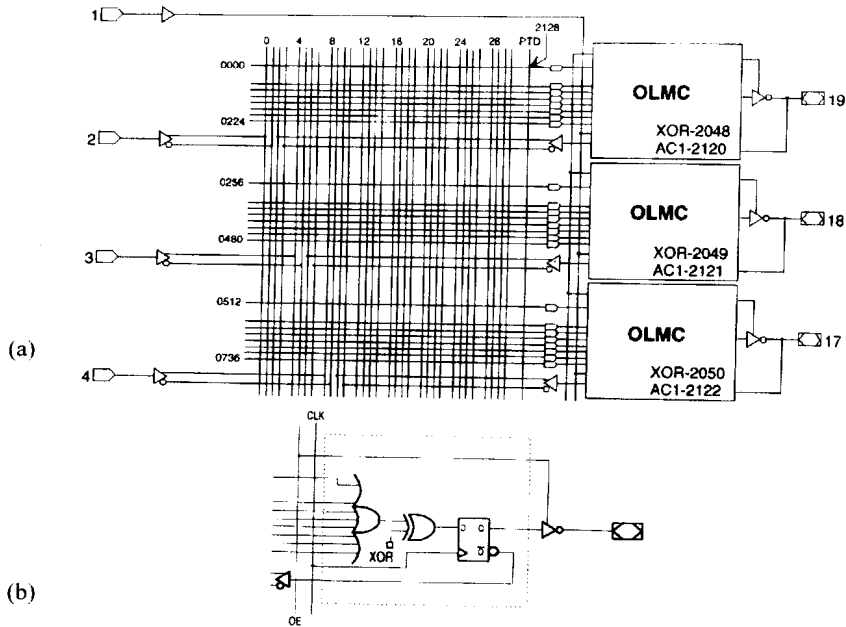


图 1-6 GAL16V8 (a) 8 个宏单元中的前 3 个 (b) 输出逻辑宏单元(OLMC)(©1997 年, Lattice)

1.2.2 按技术分类

实际上, FPL 在所有存储技术(SRAM、EPROM、E²PROM 和抑熔断技术^[10])中都是可行的。根据具体的技术可以将元器件定义为是可重复编程的还是一次性编程的。大多数 SRAM 元器件都可以通过一位流编程,从而降低了对布线的要求,但是也相应地增加了编程的时间(通常情况下,范围是毫秒(ms)级的)。对于 FPGA 来说,具有优势地位的 SRAM 元器件是基于静态 CMOS 存储技术的,并且是系统内编程和可重复编程的。然而它们还需要一个外部“引导”器件作为组态。由于电可编程只读存储器(electrically programmable read-only memory, EPROM)需要用紫外线照射来擦除,所以经常用在一次性 CMOS 可编程模式中。CMOS 电可擦可编程只读存储器(electrically erasable programmable read-only memory, E²PROM)可以用作可重复编程和系统内编程器件。EPROM 和 E²PROM 都具有设置时间短的优势。因为编程信息不是下载(downloaded)到元器件上的,所以能够得到更好的保护,禁止未经授权的使用。近来出现了一项基于 EPROM 的技术,称为闪速存储器(flash memory)的革新。这类元器件通常被视作是“页方式”的、具有更小的单元,系统内可重复的编程系统,等同于 E²PROM 元器件。最后,在表 1-2 中简要地给出了不同元器件技术的主要优点和缺点。

表 1-2 FPL 技术

技 术	SRAM	EPROM	E ² PROM	抑 熔 断	Flash
可重复编程	√	√	√	—	√
系统内可编程	√	—	√	—	√
易失性	√	—	—	—	—
复制保护	—	√	√	√	√