

高等学校电子信息工程专业教材

CPLD

应用技术与数字系统设计

陈云洽 保延翔 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

高等学校电子信息工程专业教材

CPLD 应用技术与数字系统设计

陈云洽 保延翔 编著

电子工业出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书从数字系统设计的角度出发, 简明而系统地介绍了可编程逻辑器件及其开发与应用技术。内容包括: 在系统可编程逻辑器件的一般结构、原理, 作为设计工具的硬件描述语言 ABEL-HDL、VHDL, 相关的 EDA 软件以及基于可编程逻辑器件的数字系统设计方法。在取材和编排上, 力求理论联系实际、由浅入深、循序渐进。书中结合实际应用, 对用 ABEL-HDL 和 VHDL 设计数字系统的方法做了详细讨论, 并给出许多有代表性的实例。相关器件和 EDA 开发平台选取了国内广泛使用的 Lattice 公司的 ispLSI 器件及其开发工具 ispDesignEXPERT 设计系统。

本书可作为高等院校电子、通信、自控、计算机类各专业的教材, 也可作为上述学科及相关学科工程技术人员参考书。

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

CPLD 应用技术与数字系统设计/ 陈云洽, 保延翔编著. —北京: 电子工业出版社, 2003.5
高等学校电子信息工程专业教材
ISBN 7-5053-8677-8

I. C… II. ①陈… ②保… III. ①可编程序逻辑器件—系统设计—高等学校—教材 ②数字系统—系统设计—高等学校—教材 IV. ①TP211 ②TP271

中国版本图书馆 CIP 数据核字 (2003) 第 031028 号

责任编辑: 朱怀永 特约编辑: 明足群

印 刷: 北京牛山世兴印刷厂

出版发行: 电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×980 1/16 印张: 19.5 字数: 405 千字

版 次: 2003 年 5 月第 1 版 2003 年 5 月第 1 次印刷

印 数: 5 000 册 定价: 26.00 元

凡购买电子工业出版社的图书, 如有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系。联系电话: (010) 68279077

前 言

可编程逻辑器件 (PLD) 和现场可编程门阵列 (FPGA) 是新一代的数字逻辑器件,也是近年来集成电路中发展最快的品种之一。这种器件具有速度快、集成度高、可靠性强、用户可重复编程或动态重构其逻辑功能等特点。在 EDA (电子设计自动化) 设计工具的支持下,它使数字系统的设计变得非常灵活,且大大缩短系统研制的周期,减小系统的体积和所用芯片的品种,目前在计算机、通信、图像处理等多个领域已得到广泛应用。

大规模可编程逻辑器件的结构原理和设计方法有其独特之处,且内容丰富,它已发展为数字逻辑电路的一个独立分支。而且,随着这类器件和相应的 EDA 设计工具的发展和普及,也使 EDA 进入普通实验室成为现实。它所引发的基于芯片的设计方法,自顶向下的设计方法冲击着固定功能集成块加连线的传统设计方法,正逐步成为电子系统设计方法的主流。在大学电子、通信、自控、计算机类各专业的教学中引入大规模可编程逻辑器件和电子设计自动化的内容,以适应现代电子技术的发展,很有必要。

新型的可编程逻辑器件层出不穷,器件的内部结构也在不断地发展变化。在教学内容的选取上,应重在跟上新技术的进展。原理方面要注重器件的结构特点,使用方面要注重相应的设计方法,掌握相关的 EDA 工具。本书是作者结合多年教学和科研的经验编写而成的,在介绍可编程逻辑器件的一般结构、原理、设计方法的同时,介绍了两种硬件描述语言——ABEL-HDL 和 VHDL,并以 Lattice 公司的 ispLSI 器件为对象,以相应的 EDA 软件——ispDesignEXPERT 软件为工具,讨论 ispLSI 器件和相关数字逻辑系统的设计方法。力图做到与具体器件相关,但不拘泥于具体器件,从而更好地掌握 EDA 技术和高密度可编程逻辑器件的使用。

本书由陈云治主编,第 2 章由保延翔编写,其余各章由陈云治编写。李卓君参与了本书 8.3 节实例的设计。

在本书的编写过程中,参阅了 Lattice, Xilinx, Altera 等公司公开的技术资料、数据手册等,上海 Lattice 公司的陈恒先生也提供了有关资料,在此谨致衷心感谢。编写中,参考了许多相关的专著和教材,在此谨向相关专著和教材的作者表示诚挚谢意。

由于水平所限,书中错漏和不足之处,在所难免,恳请读者指正。

作 者

2003 年 2 月于中山大学

目 录

第 1 章 可编程逻辑器件简介	(1)
1.1 可编程逻辑器件的发展	(1)
1.2 可编程逻辑器件分类	(3)
1.2.1 可编程逻辑器件按集成度的分类	(3)
1.2.2 可编程逻辑器件按结构的分类	(4)
1.2.3 可编程逻辑器件按编程工艺的分类	(7)
1.3 PLD 的基本结构	(9)
1.3.1 与或阵列	(10)
1.3.2 宏单元	(11)
1.4 FPGA 的基本结构	(12)
1.4.1 查找表型 FPGA 的结构	(13)
1.4.2 多路开关型 FPGA 的结构	(13)
1.4.3 多级与非门型 FPGA 的结构	(14)
1.5 先进的编程和测试技术	(14)
1.5.1 在系统可编程技术	(14)
1.5.2 边界扫描测试技术	(15)
思考与练习	(16)
第 2 章 ispLSI 器件的结构与原理	(17)
2.1 ispLSI 器件概述	(17)
2.1.1 ispLSI 器件简介	(17)
2.1.2 ispLSI 器件的主要技术特性	(19)
2.2 ispLSI 器件的结构与原理	(20)
2.2.1 万能逻辑块 GLB (Generic Logic Block)	(21)
2.2.2 集总布线区 GRP (Global Routing Pool)	(25)
2.2.3 输入/输出单元 IOC (Input/Output Cell)	(25)
2.2.4 输出布线区 ORP (Output Routing Pool)	(26)
2.2.5 时钟分配网络 CDN (Clock Distribution Network)	(29)
2.2.6 宏模块结构 (Megablock)	(29)
2.3 ispLSI 1016 的主要性能指标和封装	(31)

2.3.1	ispLSI 1016 的主要性能指标	(31)
2.3.2	ispLSI/pLSI 1016 的封装和引脚定义	(34)
	思考与练习	(35)
第 3 章	ispLSI 器件的编程	(37)
3.1	在系统编程技术原理	(37)
3.1.1	ispLSI 器件的编程结构	(37)
3.1.2	ISP 状态机	(39)
3.1.3	ISP 编程的定时关系	(41)
3.2	ISP 器件的编程方式	(43)
3.2.1	通过 PC 的 I/O 口编程	(44)
3.2.2	利用用户目标系统或线路板上的单片机或微处理器编程	(44)
3.2.3	多个 ISP 器件的编程	(47)
3.3	互连的在系统编程	(49)
3.3.1	ispGDS 的结构与原理	(49)
3.3.2	ispGDS 器件的编程	(51)
	思考与练习	(53)
第 4 章	ABEL-HDL	(54)
4.1	ABEL-HDL 的基本元素与语法	(54)
4.1.1	字符集	(54)
4.1.2	标识符	(54)
4.1.3	字符串	(55)
4.1.4	注释	(56)
4.1.5	操作数	(56)
4.1.6	运算符、表达式和方程	(57)
4.1.7	集合	(60)
4.1.8	特殊常量值	(61)
4.1.9	块	(62)
4.1.10	变量及变量代换	(62)
4.2	ABEL-HDL 的语言结构	(63)
4.2.1	基本结构	(63)
4.2.2	文件头部	(64)
4.2.3	定义段	(65)
4.2.4	逻辑描述段	(69)
4.2.5	测试向量段	(75)

4.2.6 结束段	(76)
4.3 指示字	(76)
思考与练习	(81)
第5章 VHDL 简介	(83)
5.1 概述	(83)
5.2 VHDL 程序结构	(84)
5.2.1 VHDL 程序的基本结构	(84)
5.2.2 实体说明	(86)
5.2.3 结构体	(87)
5.2.4 配置	(91)
5.2.5 程序包和库	(94)
5.3 VHDL 的基本元素	(96)
5.3.1 标识符	(96)
5.3.2 数据对象	(97)
5.3.3 数据类型	(98)
5.3.4 属性	(102)
5.3.5 VHDL 的表达式与运算符	(103)
5.4 VHDL 的基本描述语句	(104)
5.4.1 顺序语句	(104)
5.4.2 并行语句	(109)
5.4.3 子程序	(119)
思考与练习	(123)
第6章 ispDesignEXPERT 及其应用	(126)
6.1 可编程逻辑器件设计的一般方法	(126)
6.1.1 开发工具	(126)
6.1.2 器件设计的一般方法	(126)
6.2 ispDesignEXPERT 设计软件	(129)
6.2.1 ispDesignEXPERT 系统的设计环境	(130)
6.2.2 ispDesignEXPERT 软件的基本命令	(133)
6.3 ispDesignEXPERT 设计软件的应用	(141)
6.3.1 创建新项目	(141)
6.3.2 电路原理图的输入	(142)
6.3.3 ABEL-HDL 文件的输入	(145)
6.3.4 建立顶层设计文件	(146)

6.3.5	层次化操作	(149)
6.3.6	编译和设计的实现	(149)
6.3.7	逻辑功能仿真(逻辑模拟)	(150)
6.3.8	时序仿真	(154)
6.3.9	仿真调试	(154)
6.3.10	引脚锁定	(156)
6.3.11	ISP 器件的编程	(158)
6.4	VHDL 输入设计方式	(160)
6.4.1	VHDL 文件的输入	(160)
6.4.2	VHDL 源程序的综合	(161)
6.4.3	仿真测试	(162)
6.4.4	引脚锁定和器件的编程	(163)
6.5	ispDesignEXPERT 的文件后缀及含义	(163)
	思考与练习	(165)
第 7 章	采用 ISP 器件的数字系统设计	(167)
7.1	采用 ISP 器件的数字系统设计方法	(167)
7.1.1	数字系统的设计过程	(167)
7.1.2	数字系统设计的基本方法	(168)
7.2	组合逻辑电路的设计	(169)
7.2.1	运用 ABEL-HDL 设计组合逻辑电路	(169)
7.2.2	运用 VHDL 设计组合逻辑电路	(176)
7.3	时序逻辑电路的设计	(180)
7.3.1	运用 ABEL-HDL 设计时序逻辑电路	(180)
7.3.2	运用 VHDL 设计时序逻辑电路	(187)
7.4	测试向量序列的编写	(196)
7.4.1	编写测试向量序列的基本方法	(197)
7.4.2	编写测试向量的技巧	(198)
7.5	数字系统设计	(200)
7.5.1	系统设计	(200)
7.5.2	采用电路原理图/ABEL-HDL 描述系统功能	(203)
7.5.3	编译、仿真测试与适配	(210)
7.5.4	采用 ABEL-HDL 描述系统功能	(212)
7.5.5	采用电路原理图/VHDL 描述系统功能	(215)
7.5.6	采用 VHDL 描述系统功能(一)	(223)

7.5.7 采用 VHDL 描述系统功能(二)	(231)
思考与练习	(237)
第 8 章 数字系统设计实例	(238)
8.1 智力竞赛抢答器的设计	(238)
8.1.1 抢答器的功能描述	(238)
8.1.2 抢答器的设计	(238)
8.1.3 采用电路原理图/ABEL-HDL 描述的抢答器的设计	(240)
8.1.4 仿真与测试	(243)
8.1.5 采用 ABEL-HDL 描述的抢答器的设计	(245)
8.1.6 采用 VHDL 描述的抢答器的设计	(247)
8.2 交通信号灯控制器的设计	(250)
8.2.1 交通信号灯控制器的功能描述	(250)
8.2.2 交通信号灯控制器的设计	(251)
8.2.3 采用电路原理图/ABEL-HDL 描述的交通信号灯控制器的设计	(252)
8.2.4 仿真与测试	(257)
8.2.5 采用 VHDL 描述的交通信号灯控制器的设计	(259)
8.3 简易电子乐器的设计	(267)
8.3.1 乐曲演奏电路的基本原理	(267)
8.3.2 简易电子乐器的功能描述	(267)
8.3.3 电子萨克斯管的设计	(269)
8.3.4 采用电路原理图/ABEL-HDL 描述的电子萨克斯管的设计	(270)
8.3.5 采用 VHDL 描述的电子萨克斯管的设计	(274)
8.4 数字频率计的设计	(278)
8.4.1 数字频率计的功能描述	(278)
8.4.2 简易数字频率计的设计	(279)
8.4.3 采用电路原理图描述的简易数字频率计的设计	(279)
8.4.4 采用 VHDL 描述的简易数字频率计的设计	(282)
8.5 FIR 数字滤波器的设计	(288)
8.5.1 FIR 数字滤波器结构简介	(288)
8.5.2 FIR 滤波器的设计方案	(289)
8.5.3 采用 VHDL 描述的 FIR 滤波器的设计	(290)
思考与练习	(297)
参考文献	(299)

第 1 章 可编程逻辑器件简介

大规模可编程逻辑器件是现代数字系统设计的重要基础，本章概要介绍几类有代表性的可编程逻辑器件（Programmable Logic Device, PLD）的结构与原理，以及相关的工艺、编程和测试技术。可编程逻辑器件的技术发展很快，在学习时，尽管要结合某些器件进行讨论，但不应拘泥于具体的器件，重要的是掌握基本的概念、结构、原理和分析方法，掌握器件的使用，掌握先进的设计方法和工具，跟上新技术的发展。

1.1 可编程逻辑器件的发展

早期的可编程逻辑器件出现于 20 世纪 70 年代末和 80 年代初，其中有作为只读存储器的 PROM, EPROM, E²PROM；有由可编程“与阵列”和可编程“或阵列”构成，便于实现有较多公共与项的多输出函数的 PLA；有在 PROM 和 PLA 基础上发展起来，由可编程“与阵列”和固定的“或阵列”构成的 PAL（它是 PLD 中应用最广、结构最多的一类，缺点是一次性可编程、输出结构不可灵活组态，可测试性差，不可加密）；而 GAL 是在总结生产 PAL 器件经验的基础上，1983 年开始推出的性能优异的 PLD（GAL 具有输出逻辑宏单元（OLMC），使用更加灵活方便）。

20 世纪 80 年代中期以来，PLD 的发展异常迅速，已从简单的 PAL, GAL 发展到 CPLD, EPLD, MPLD, FPGA 等系列，单一芯片的门数也从几百门增加到数百万门。

1984 年，Altera 公司率先采用 CMOS 和 UVEPROM 工艺研制出一种新型的可擦写 PLD，它将 EPROM 直接合成于 PLD 芯片中，简称 EPLD。其典型产品如 Classic, MAX 5000, MAX 7000 系列。VANTIS 公司（AMD 公司属下的可编程逻辑器件子公司）的 MACH 1, MACH 2, MACH 3, MACH 4, MACH 5 等系列也是 EPLD 中很有特色的器件。不同型号的 EPLD 大都由不同个数的宏单元组合而成，这使 EPLD 应用设计的编程灵活性较之传统的 PAL, GAL 器件有很大的改善。由于其保留了逻辑块级的编程，内部连线相对固定，即使对大容量的器件，内部延时仍较小，故有利于器件在较高频率下工作。

作为可编程器件另一分支的现场可编程门阵列（Field Programmable Gate Array, FPGA），是一种可由用户根据设计的数字系统要求，在现场由用户配置、定义的高密度专用数字集成电路。它将 VLSI 逻辑集成的优点和可编程器件设计灵活、制作及上市快速的长处相结合，使设计者在 FPGA 开发系统软件的支持下，可在现场直接根据系统要求定义和修改其逻辑功能，使一个包含数千个、数万个逻辑门的数字系统可在几天之内完成

设计并且实现, 将以前由许多 TTL, PLD, EPLD 执行的逻辑功能集成到单一芯片的 FPGA 上。由于芯片数目的减少, 印制板面积缩小, 可使系统规模全面缩减, 从而降低成本, 还可大大提高系统的可靠性。Xilinx 公司于 1985 年推出了世界首块现场可编程门阵列, 随后开发了多个系列的 FPGA, 如 XC 2000, XC 3000, XC 4000, XC 5000 等。针对逻辑系统往往需要大量内部存储器的情况, 1995 年 Altera 公司推出 FLEX 10K 系列嵌入式 RAM 的 FPGA, 这是 10 万门级的器件, 它在芯片内集成了很大容量的内部 SRAM, 可用于存储器密集型的需要。随之, Xilinx 公司也推出与其类似的 XC 6200 系列的 FPGA, 它除内部有大容量存储器外, 还具有可以按照数据总线的速度传输配置数据的特点, 可在 200 μ s 内把全部配置文件传输完毕, 适用于对条件变化需要迅速做出反应的系统。1996 年 Lattice 公司则向市场推出称为“子块化”(Cell-based)的可编程器件 ispLSI 6192, 在单一芯片中集成了逻辑功能子块、存储器子块和计数器/寄存器子块。

“在系统编程”(ISP)是 Lattice 公司发明的一种新技术, Lattice 公司于 1992 年研制出的大规模可编程逻辑器件 ispLSI, 不仅可以反复编程, 而且首次使 PLD 彻底摆脱编程器。用户通过下载电缆对系统中安装在印刷电路板上的 ispLSI 器件进行编程和再编程。随之, Xilinx 公司、Altera 公司也相继推出自己的 ISP 器件, 如 XC 9520, MAX 9000 系列等。ISP 技术使系统中曾经是一成不变的硬件, 变成动态的和可以改变的。人们不仅可以改变软件, 还可改变定型的硬件。如安装到现场的通信设备, 随着用户的变化、容量需求的变化、信令的变化, 需要升级或更新, 这将归结于硬件的变化; 而对采用 ISP 器件的设备来说, 硬件的变化将变得轻而易举。

近年来, 可编程逻辑器件在高密度、高速度、低功耗等方面发展很快, Xilinx, Altera, Lattice 等著名半导体器件公司, 研发出若干新型的 CPLD, FPGA 器件。在 CPLD 方面, Xilinx 公司推出了高性能、低成本的 XC 9500[tm]系列以及在单个器件内结合了极低功耗和高速度、高密度和多 I/O 引脚特点的 CoolRunner[tm]系列。XC 9500 系列的密度从 36 至 288 宏单元, 并支持 2.5V、3.3V 和 5V 供电电压, 且支持在系统编程, 适用于需要快速的设计开发、更长的系统寿命和强大的现场升级能力的系统。CoolRunner 具有快速零功耗的特点, 器件在闲置模式几乎不消耗任何电流, 对快速增长的由电池供电的便携式电子设备(如笔记本电脑、电话手机等)非常理想; 这类 CPLD 在实际运行时的动态功耗也比传统 CPLD 小得多, 对高性能、散热敏感的设备(如电信交换机、视频会议系统、高端的测试设备等)来说, 也是非常理想的。Lattice 公司于 1999 年收购了 Vantis 公司, 提高了在 PLD 市场上的占有率, 加速了新产品的开发; 随之推出了低功耗, 高速度, 支持 2.5V 和 1.8V 供电电压, 增强 I/O 性能的新一代可编程逻辑器件 ispMACH 4000 以及 ispMACH 5000VG 系列。在 FPGA 方面, Xilinx 公司于 1999 年推出了超过 300 万门的 VirtexE 系列, 2000 年推出的 VirtexEM 系列是率先采用先进的金属铜工艺的 FPGA, 还提供了网络交换应用的额外片上存储器。Xilinx 公司的 Spartan[tm] FPGA 对低成本、大批量应用的系统是

非常理想的，主要目标是替代固定逻辑门阵列产品和专用标准产品（ASSP）。Spartan II E 系列器件采用目前一些最先进的 FPGA 技术，包括支持高达 19 种 I/O 标准（包括差分 I/O，如 LVDS）、范围广泛的 IP（包括先进的 DSP 和处理器核心）、片上 RAM 块，以及用于片级和板级时钟管理的数字锁相环等，这对快速完成系统设计很有价值。Lattice 公司于 2002 年收购了杰尔（Agere）公司的 FPGA 业务后，步入了 FPGA 器件的市场。目前除了提供 3 个 ORCA 系列的 FPGA 器件外，还提供一种突破性的 FPGA 产品——现场可编程系统芯片 FPSC。FPSC 在单个可编程芯片上综合了通用 FPGA 逻辑和内嵌的 IP（知识产权）核，使用户可以以灵活的方式快速实现复杂的高性能的系统级设计。2002 年 7 月又推出了第一个在系统可编程且动态可重构的、瞬时上电的 FPGA 产品 ispXPGA™ 系列以及可让用户有效地交替使用快速逻辑和块存储资源的新型的 PLD 器件 ispXPLD™ 系列。ALTERA 公司于 1999 年推出带有 PLL, CAM, EAB, LVDS, 2.5 V/1.8 V 百万门级的 APEX20K/20KE 系列，2002 年 2 月又推出新一代可编程逻辑器件 Stratix 系列。该系列采用 1.5 V 内核，0.13μm 的全铜工艺；内嵌三级存储单元以及乘加结构的 DSP 块（包括硬件乘法器/硬件累加器和流水线结构），适用于高速数字信号处理和各类算法的实现；采用全新的布线结构，在保证延时可预测的同时，提高了资源利用率和系统速度；增强时钟管理和锁相环能力，实现了 K*M/N 的任意倍频/分频，且参数可动态配置；增加片内终端匹配电阻，提高信号完整性，简化了 PCB 布线。近年来出现的这些新器件，正作为各 CPLD/FPGA 厂商的主流产品，有力地推动着电子设计领域技术的进步。

1.2 可编程逻辑器件分类

常用的可编程逻辑部件有 PROM, PLA, PAL, GAL, EPLD, CPLD, FPGA 等。在可编程器件的发展过程中，对器件的命名不很规范，通常以某种特征命名，然而一种器件往往具备几种特征，导致难于进行严格的分类，也就没有统一的分类标准。本节介绍几种比较流行的分类方法。

1.2.1 可编程逻辑器件按集成度的分类

可编程逻辑器件按集成度可分为低密度可编程逻辑器件（LDPLD）和高密度可编程逻辑器件（HDPLD）两类。单片密度在 1000 门以下的为低密度可编程逻辑器件，在 1000 门以上的为高密度可编程逻辑器件。这里所指的门是用来衡量 PLD 规模的 PLD 等效门，而不是指传统的与门或者或门。低密度可编程逻辑器件通常采用与或阵列、单一区块、8~16 个左右的输入/输出引脚、芯片封装为 20~28 个引脚；高密度可编程逻辑器件通常采用与或阵列/查表、门阵列、多区块、36 个以上的输入/输出引脚、芯片封装为 44 个引脚到数百个引脚。在可编程逻辑器件的发展史上，GAL22V10 可视为低密度可编程逻辑器

件与高密度可编程逻辑器件的分水岭，其集成度在 500~700 门之间（因制造商不同而有所不同）。以上述分类法为依据，则 PROM, PLA, PAL, GAL 等器件为低密度可编程逻辑器件，而 EPLD, CPLD, FPGA 等为高密度可编程逻辑器件。可编程逻辑器件按集成度的分类情况如图 1.1 所示。

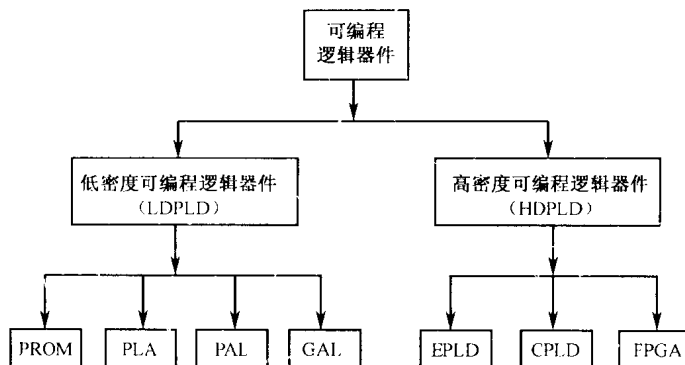


图 1.1 可编程逻辑器件按集成度的分类

低密度可编程逻辑器件易于编程，对开发软件的要求较低，一片 PAL 或 GAL 芯片通常可以替代若干片通用的 SSI, MSI 逻辑电路，在 20 世纪 80 年代得到了广泛的应用。随着技术的进步，低密度可编程逻辑器件也显露出它在集成度和性能方面的不足。低密度可编程逻辑器件的 I/O 引脚、时钟、寄存器等资源的数目有限，没有内部可编程互连结构，使设计的灵活性受到较大的限制。高密度可编程逻辑器件在提高集成度的同时，在结构、性能等方面进行了许多改进，20 世纪 90 年代以来，发展更为迅速，应用也更广泛，备受人们的关注。

1.2.2 可编程逻辑器件按结构的分类

从结构上看，可编程逻辑器件大体是从与或阵列和门阵列两类基本结构上发展起来的，所以，依据结构的特征，把可编程逻辑器件分为两大类：PLD（Programmable Logic Device）和 FPGA（Field Programmable Gate Array）器件。基本结构为与或阵列结构的器件，称为可编程逻辑器件 PLD；基本结构为门阵列结构的器件，称为现场可编程门阵列 FPGA。

1. PLD（可编程逻辑器件）

PLD 的基本逻辑结构是与/或阵列，它能有效地实现与/或形式的逻辑函数。PLD 包括低密度的 PLD（PROM, PLA, PAL 和 GAL，它们也被称为简单的 PLD）以及 EPLD 和 CPLD。

PROM: 可编程只读存储器。其结构为与阵列固定、或阵列可编程，工艺上为熔丝型，只能写入一次，不能擦除或重写。随着技术的发展，又出现了 EPROM（紫外线擦除可编程

只读存储器)和 E^2 PROM (电可擦除可编程只读存储器)。**PROM** 的特点是价格低、易于编程,适合于存储函数和数据表格等,在微型计算机系统中它也用于存放程序和数据。

PLA (Programmable Logic Array): 可编程逻辑阵列。其与阵列和或阵列均可编程。**PLA** 曾被认为是很有发展前途的可编程逻辑器件,但由于器件资源的利用率低,现在较少使用。

PAL (Programmable Array Logic): 可编程阵列逻辑。20 世纪 70 年代后期 **PAL** 由 MMI 公司推出。其与阵列可编程而或阵列固定,输出则具有多种结构形式,使器件既有规则的阵列结构,又能实现灵活多变的逻辑功能。**PAL** 的型号随输出结构而异,因而其型号较多。

GAL (Generic Array Logic): 通用阵列逻辑。**GAL** 是 20 世纪 80 年代由 Lattice 公司推出的一种采用先进的 E^2 CMOS 工艺,可电擦除、可重复编程、可设置加密位的低密度 **PLD**。按 **GAL** 器件与或阵列的编程结构, **GAL** 器件分为两类,一类与 **PAL** 相似,与阵列可编程而或阵列固定,称为普通型的 **GAL** 器件,现有的大部分 **GAL** 器件,如 **GAL16V8**, **ispGAL16Z8**, **GAL20V8** 都属于这一类,图 1.2 所示为 **GAL20V8** 的结构框图(若删除该结构框图中虚线框部分,则为 **GAL16V8** 的结构框图);另一类器件的与阵列和或阵列都可编程,称为新一代

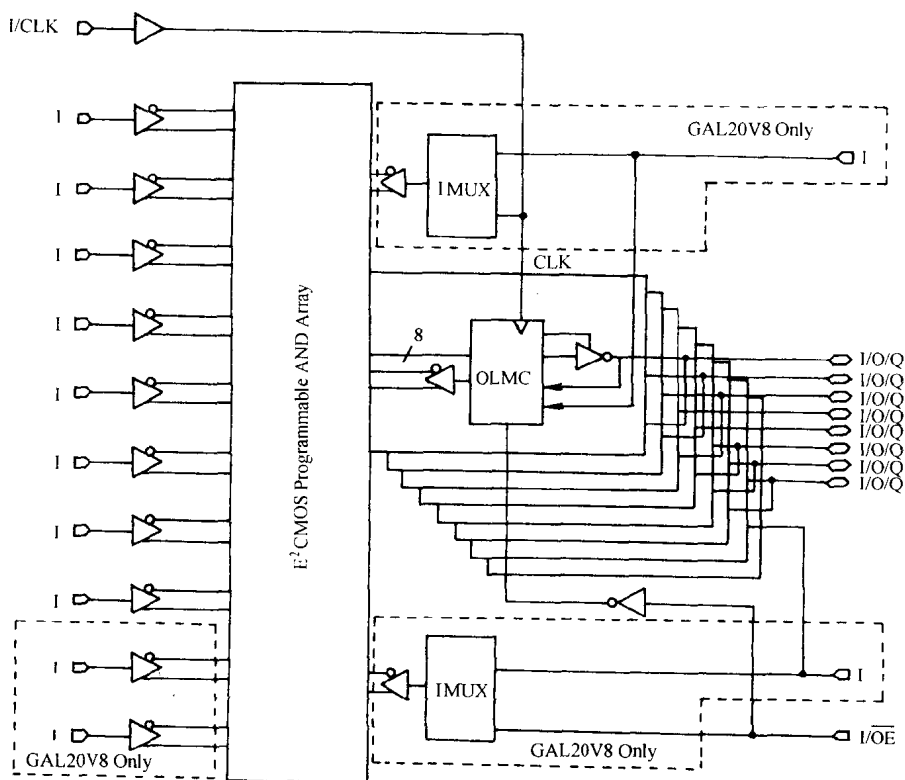


图 1.2 GAL20V8 的结构框图

GAL 器件,如 GAL39V18 等。GAL 的输出采用输出逻辑宏单元结构(OLMC),通过对 OLMC 的配置可以得到多种形式的输出和反馈,给设计带来很大灵活性。在实际应用中,由于 GAL 器件对 PAL 器件的仿真具有 100%的兼容性,故常用 GAL 器件代替 PAL 器件。

EPLD (Erasable Programmable Logic Device): 20 世纪 80 年代中期由 Altera 公司推出的一种基于 EPROM 和 CMOS 技术的新型可擦除可编程逻辑器件。EPLD 的基本逻辑单位是宏单元,宏单元由可编程的与或阵列、可编程寄存器和可编程 I/O 三部分组成。宏单元及整个器件的逻辑功能都由 EPROM 定义和规划。可以说 EPLD 是改进的 GAL,其特点是大量增加输出宏单元的数目,提供更大的与阵列。EPLD 特有的宏单元结构,使设计灵活性比 GAL 进一步提高;而高集成度,使单一芯片能够实现更强的逻辑功能;相对固定的内部连线,使内部延时很小,有利于器件在高频下工作。

CPLD (Complex PLD): 复杂可编程逻辑器件。与 EPLD 相比,CPLD 增加了内部连线,对逻辑宏单元和 I/O 单元也做了重大改进。通常,CPLD 中至少包含可编程逻辑宏单元、可编程 I/O 单元、可编程内部连线等部分。有些 CPLD 器件内部还集成了 RAM, FIFO 或双口 RAM 等存储器,适用于 DSP 的应用设计。典型的 CPLD 器件,如 Lattice 公司的 pLSI/ispLSI 系列、Xilinx 公司的 7000 和 9000 系列、Altera 公司的 MAX 9000 系列、VANTIS 公司的 MACH 系列等。图 1.3 所示为 Lattice 公司的 ispLSI 1016 芯片的结构框图。ispLSI 器件的重要特性在于具有在系统可编程能力(In System Programmable),对焊到线路板上的器件可进行编程和再配置。ISP 器件在使用上和 FPGA 器件一样方便。

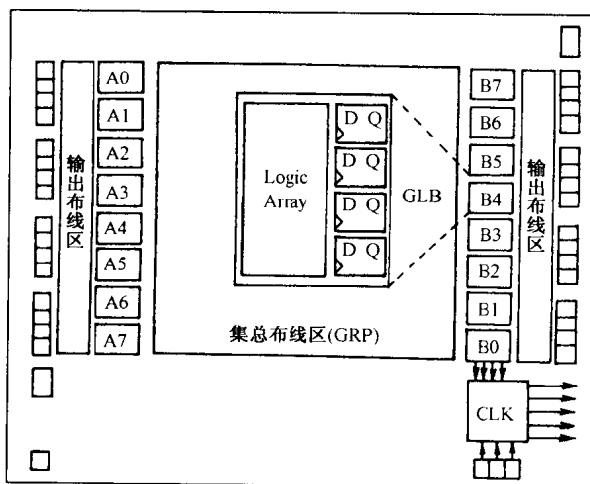


图 1.3 ispLSI 1016 的结构图

CPLD 由于具有与 FPGA 相似的集成度和易用性,在速度上还有一定优势,因而在可编程器件技术的竞争与发展中,CPLD 和 FPGA 成为引导可编程逻辑器件技术发展的两个主要方向。

2. FPGA（现场可编程门阵列）

FPGA 是最近 10 多年发展起来的新型可编程逻辑器件。1985 年 Xilinx 公司推出了世界上首片 FPGA 器件。不同厂家生产的 FPGA 器件，具体结构不尽相同，性能也各具特色。但它们的共同之处在于：广义上它们具有掩膜编程门阵列的通用结构，由逻辑功能块排列为阵列，并由可编程的互连资源连接这些逻辑功能块，以实现不同的逻辑设计。

典型的 FPGA 结构如图 1.4 所示，它通常包含三类可编程资源：可编程逻辑功能块 CLB（Configurable Logic Blocks），可编程 I/O 模块 IOB（Input/Output Block）和可编程内部互连 PI（Programmable Interconnect）。CLB 是实现逻辑设计的基本单元，它们排列为阵列，散布于整个芯片。CLB 的功能很强，不仅实现逻辑函数，还可配置为 RAM 等复杂形式。IOB 作为芯片上逻辑与外部封装引脚的接口，通常围绕着阵列排列于芯片的四周。PI 包括各种长度的连线和一些可编程连接开关，通过它们把各个 CLB、IOB 按设计要求连接起来，构成特定功能的电路。

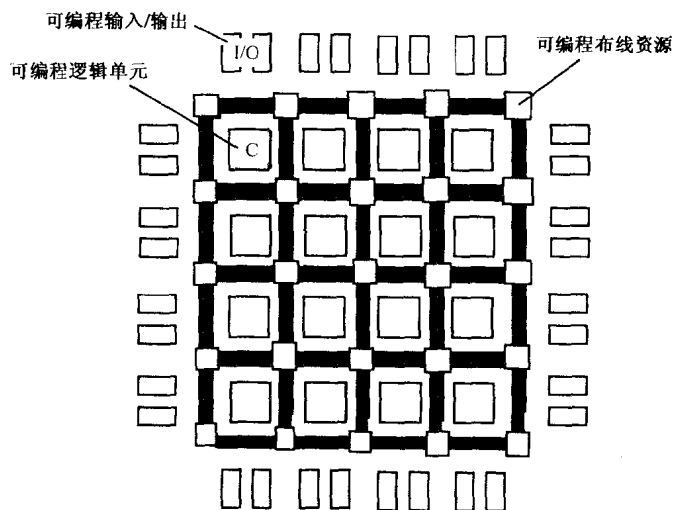


图 1.4 FPGA 器件的基本结构

FPGA 器件的功能由逻辑结构的配置数据决定，工作时，配置数据存放于片内的 SRAM 或熔丝图上。使用 SRAM 的 FPGA 器件，工作前需从芯片外部加载配置数据。配置数据可存储于片外的 EPROM 或其他存储体上。用户可控制加载过程，在现场修改器件逻辑功能，即现场可编程。

1.2.3 可编程逻辑器件按编程工艺的分类

按编程所采用的工艺，可编程逻辑器件大体上可分为 4 类。

1. 熔丝 (Fuse) 和反熔丝 (Anti-fuse) 编程器件

熔丝编程器件是用熔丝作为开关元件，这些开关元件在未编程时处于导通状态，编程时，在不需要连接处将熔丝熔断，保留在器件内的熔丝模式决定相应器件的逻辑功能。采用反熔丝 (Anti-fuse)，也称熔通技术编程的器件，是用逆熔丝作为开关元件。这些开关元件在未编程时处于开路状态，编程时，在需要连接处的逆熔丝开关元件两端加上编程电压，逆熔丝将由高阻抗变为低阻抗，实现两点间的连接，编程后器件内的反熔丝模式决定了相应器件的逻辑功能。熔丝和反熔丝编程器件为一次性可编程器件，比较适合定型产品和大批量应用，也常用于需要高性能及保密性要求高的场合。

2. 紫外线擦除的可编程器件

紫外线擦除的可编程器件采用浮栅编程技术，目前浮栅管主要采用雪崩注入 MOS 管 (FAMOS 管) 和叠栅注入 MOS 管 (SIMOS 管)。浮栅管相当于一个电子开关，如 N 沟浮栅管，当浮栅中没有注入电子时，浮栅管导通；当浮栅中注入电子后，浮栅管截止。浮栅管的浮栅在原始状态没有电子，如果把源极和衬底接地，且在源-漏极间加电压脉冲，产生足够强的电场，使电子加速跃入浮栅中，则使浮栅带上负电荷，电压脉冲消除后，浮栅上的电子可以长期保留；当浮栅管受到紫外光照射时，浮栅上的电子将被扫向衬底，擦除所记忆的信息，而为重新编程做好准备。EPROM 以及大多数的 EPGA 器件采用这种方式编程。

3. 电擦除的可编程器件

电擦除的可编程器件在采用浮栅编程技术的同时，采用了 E^2 CMOS 工艺。在 SIMOS 管的浮栅与漏极间有一薄氧化层区，其厚度为 $10\mu\text{m}\sim 15\mu\text{m}$ ，可产生“隧道效应”。编程时，漏极接地，控制栅加 20 V 的脉冲电压，衬底中的电子将通过隧道效应进入浮栅，脉冲消除后，浮栅上的电子可以长期保留，使该管正常工作时处于截止状态；若将其控制栅接地，漏极加 20 V 的脉冲电压，浮栅上的电子又将通过隧道效应返回衬底，则使该管正常工作时处于导通状态，达到对该管擦除的目的。编程和擦除都是通过向漏极和控制栅上加入一定幅度和极性的电脉冲来实现，可由用户在“现场”用编程器来完成。实际上，编程和擦除是同时进行的，每编程一次，就以新的信息代换原有的信息。GAL, ispLSI 属于这类器件。闪速存储器 (闪速 EPROM) 也是一种电擦除的浮栅编程器件，其特点是在若干 ms 内可擦除全部或一段存储器。部分 FPGA 器件就是使用闪速 EPROM 存储其构造代码，它使器件具有非易失性和可重编程的双重优点，但在编程灵活性上比 SRAM 型的 FPGA 稍差，不能实现动态重构。