面向 21 世纪高等院校规划教材

数字系统电子自动化设计教程

——CPLD 原理与应用

梁　淼　刘会军　编著

北京理工大学出版社
BEIJING INSTITUTE OF TECHNOLOGY PRESS

需要全本请访问 www.ertongbook.com

版权专有　侵权必究

图书在版编目（CIP）数据

数字系统电子自动化设计教程：CPLD 原理与应用/梁淼，刘会军编著.
—北京：北京理工大学出版社，2008.8
ISBN 978-7-5640-1713-2

Ⅰ.数…　Ⅱ.①梁…②刘…　Ⅲ.①数字系统-系统设计-教材②可编程序逻辑器件-教材　Ⅳ.TP271　TP332.1

中国版本图书馆 CIP 数据核字（2008）第 117932 号

出版发行／北京理工大学出版社
社　　址／北京市海淀区中关村南大街 5 号
邮　　编／100081
电　　话／(010)68914775(办公室)　68944990(批销中心)　68911084(读者服务部)
网　　址／http：// www.bitpress.com.cn
经　　销／全国各地新华书店
印　　刷／保定市中画美凯印刷有限公司
开　　本／787 毫米×1092 毫米　1/16
印　　张／11.25
字　　数／269 千字
版　　次／2008 年 8 月第 1 版　2008 年 8 月第 1 次印刷
印　　数／1～4000 册　　责任校对／陈玉梅
定　　价／23.00 元　　责任印制／母长新

图书出现印装质量问题，本社负责调换

前　言

在信息技术高速发展的今天，随着可编程逻辑器件的飞速发展，以 FPGA、CPLD 为核心的片上系统（System On Chip，简称 SOC）和可编程片上系统（System On a Programmable Chip，简称 SOPC）已推广使用，数字系统自动化设计和应用已经进入一个全新的阶段。

在数字系统电子自动化设计领域中，由于数字系统的复杂程度和 EDA 工具的水平都在快速提高，传统的由定制集成电路块搭建数字系统的时代正悄然退出历史舞台。现代数字系统的设计，已经逐步由具备不同专业知识的工程技术人员使用相应的 EDA 工具对数字系统分阶段进行设计的时期，向使用基于 EDA 技术的 PLD 来设计数字系统的时期转变。采用硬件描述语言，从数字系统整体行为级别的层次开始，对电路做功能描述，自顶向下地跨越各个层次，从而完成整个数字系统的设计方法已成为现代数字系统设计的主流方法。

基于大规模、高密度的 PLD 芯片设计的技术给数字系统电子自动化设计带来了根本性的变化，可以认为 PLD 是当前数字系统设计领域中最具活力和发展前景的，其影响力与 20 世纪 70 年代单片机的发明齐名。当今的系统设计师，只需要一台计算机、一套与所使用的 PLD 相适应的开发系统和 PLD 芯片，就可以进行数字系统设计。此外，还可以边设计边修改，成为产品后仍能修改、升级，其方便性、灵活性可谓是前所未有。

为了适应电子系统设计技术的发展，培养应用型、创新型、综合型、可参与市场竞争的电子技术人才，提高学生综合应用数字系统理论、可编程逻辑器件和计算机等先进设备及技术的能力，编写了本教材。本教材既可以作为高年级本科生和研究生数字系统自动化设计方法等课程的教材，又可以作为电子工程技术人员的参考资料。

全书共有 6 章，其中第 1 章主要介绍了 PLD 的发展及现状；第 2 章展现了现代可编程逻辑器件的结构特点和应用优势；第 3 章讲述了数字系统设计的控制算法语言及在系统可编程技术；第 4 章讲解了在设计中需要使用的国际通用硬件描述语言（VHDL）基础；第 5 章给出了 Altera 公司的可编程逻辑器件开发软件 Quartus Ⅱ 的使用方法、使用技巧和相关事项；第 6 章为读者提供了许多应用 VHDL 语言的设计实例。

本书的第 1、第 2、第 3、第 5 章由梁淼编写，第 4、第 6 章由刘会军编写。梁淼担任主编，负责全书的组织、统稿和定稿工作。本书由北京理工大学信息科学技术学院赵宏图副教授主审，赵宏图副教授认真审阅了全部书稿，对全书各章、节中存在的问题提出了很好的修改意见和建议，编者谨此致以由衷的感谢。

限于编者水平及编写时间的仓促，书中的错误、疏漏和不妥之处在所难免，恳请读者批评指正。

编　者

目　　录

第 1 章　绪论 …… 1
§ 1.1　数字系统电子自动化设计和 PLD 的发展概况 …… 1
1.1.1　数字系统电子自动化设计的历程 …… 1
1.1.2　可编程逻辑器件的发展概述 …… 2
§ 1.2　可编程逻辑器件的基本结构 …… 3
1.2.1　简单可编程逻辑器件（SPLD）的基本结构 …… 4
1.2.2　PAL 的原理与使用 …… 5
1.2.3　GAL 的原理与特点 …… 9
§ 1.3　可编程 ASIC 及其特点 …… 14
1.3.1　可编程 ASIC 简介 …… 14
1.3.2　可编程 ASIC 的特点及发展 …… 15
§ 1.4　可编程逻辑器件的分类 …… 16
1.4.1　按互连特性分类 …… 16
1.4.2　按编程方法分类 …… 16
1.4.3　按器件结构的复杂程度分类 …… 17
1.4.4　PLD 的主流厂商及主要产品简介 …… 17

第 2 章　现代可编程逻辑器件 …… 20
§ 2.1　概述 …… 20
§ 2.2　Altera 的典型可编程逻辑器件 …… 20
2.2.1　FLEX10K 系列 …… 21
2.2.2　MAX7000 系列 …… 27
§ 2.3　Altera 器件的边界扫描测试 …… 31
2.3.1　引言 …… 31
2.3.2　IEEE 1149.1—1990 边界扫描测试的结构 …… 31
2.3.3　边界扫描寄存器 …… 33
2.3.4　JTAG BST 操作模式控制 …… 35
2.3.5　JTAG BST 操作的使能及原则 …… 42
2.3.6　边界扫描描述语言（BSDL） …… 43
§ 2.4　可编程逻辑器件设计中的基本问题及使用该器件的工作条件 …… 43
2.4.1　引言 …… 43
2.4.2　可编程逻辑器件设计中的若干基本问题 …… 43
2.4.3　可编程逻辑器件的工作要求 …… 52

§ 2.5 Altera 器件的配置与下载 …… 53
2.5.1 引言 …… 53
2.5.2 ByteBlaster 并口下载电缆及使用 …… 54
2.5.3 BitBlaster 串行下载电缆及使用 …… 59

第 3 章 现代数字系统设计 …… 61
§ 3.1 概述 …… 61
§ 3.2 现代数字系统的设计方法 …… 61
3.2.1 ASM 图的图形符号 …… 62
3.2.2 ASM 图的硬件实现 …… 65
3.2.3 用 ASM 图设计小型数字系统 …… 69
§ 3.3 用寄存器传输语言（RTL）实现数字系统简述 …… 74
§ 3.4 数字系统的自动化设计流程 …… 78
3.4.1 自顶向下的设计方法 …… 78
3.4.2 数字系统自动化设计的流程 …… 79
3.4.3 在系统编程技术（ISP） …… 80

第 4 章 VHDL 语言基础 …… 82
§ 4.1 硬件描述语言概述 …… 82
§ 4.2 VHDL 语言数据类型及运算操作符 …… 83
4.2.1 VHDL 语言的标识符 …… 83
4.2.2 VHDL 语言中的对象 …… 85
4.2.3 VHDL 语言的数据类型 …… 87
4.2.4 VHDL 语言的运算操作符 …… 91
§ 4.3 VHDL 语言程序结构 …… 93
4.3.1 实体 …… 93
4.3.2 构造体 …… 94
4.3.3 库（Library） …… 95
4.3.4 包集合（Package） …… 96
4.3.5 配置（Configuration） …… 96
§ 4.4 VHDL 语言的主要描述语句 …… 97
4.4.1 顺序（Sequential）描述语句 …… 98
4.4.2 并发（Concurrent）描述语句 …… 106

第 5 章 可编程逻辑器件的开发系统 …… 117
§ 5.1 概述 …… 117
§ 5.2 Quartus II 开发系统 …… 117
5.2.1 Quartus II 6.0 的安装及设计流程 …… 118
5.2.2 Quartus II 软件的快捷键 …… 120

需要全本请访问 www.ertongbook.com

5.2.3 VHDL 文本设计输入的流程 …… 121
5.2.4 原理图设计输入的流程 …… 134
§5.3 Quartus II 开发系统使用进阶 …… 140
5.3.1 编辑用户库 …… 140
5.3.2 Altera 公司的 IP Core …… 141
5.3.3 在编译平面图中查看适配结果 …… 142
5.3.4 由 RTL Viewer 观看电路结构 …… 142
5.3.5 由 Technology Map Viewer 观看综合结果 …… 144

第6章 逻辑电路设计实例 …… 146
§6.1 组合逻辑电路设计 …… 146
6.1.1 简单门电路设计 …… 146
6.1.2 编码器、译码器电路设计 …… 148
6.1.3 运算器电路设计 …… 150
6.1.4 缓冲器电路设计 …… 153
§6.2 时序逻辑电路设计 …… 156
6.2.1 触发器电路设计 …… 157
6.2.2 分频器电路设计 …… 159
6.2.3 计数器电路设计 …… 160
6.2.4 移位寄存器电路设计 …… 162
§6.3 逻辑电路应用设计 …… 164
6.3.1 应用设计实例一：数字密码锁设计 …… 164
6.3.2 应用设计实例二：提升机松绳故障自动保护器设计 …… 166

参考文献 …… 172

第1章　绪　论

§1.1　数字系统电子自动化设计和PLD的发展概况

1.1.1　数字系统电子自动化设计的历程

电子线路是由模拟电路和数字电路构成的，这两种电路在发展历史上相辅相成。但是随着信息时代的到来，单纯用模拟电路实现的电子设备已渐渐淡出，从计算机到移动电话、从各种家用电器到军用雷达、从医用CT仪器等设备到航空、航天通讯设备，仅仅在微弱信号放大、高速数据采集和大功率输出等局部电路采用模拟电路，其余部分几乎都采用数字电路。因此，现代的电子设备主体部分是数字系统。

早期，通常将基本逻辑门、触发器称为逻辑器件，将单一功能的中、小规模集成电路，如编码器、加法器、计数器等称为逻辑功能部件，将由若干逻辑部件组成的，能执行较复杂逻辑功能的电路称为数字系统。近年来，一般将数字系统分为三部分：控制器、处理器和存储器。随着半导体技术、集成电路技术和计算机辅助设计技术的发展，数字系统的实现方法，经历了由电子管、晶体管、小规模集成电路（SSI）、中规模集成电路（MSI）到大规模集成电路（LSI）、超大规模集成电路（VLSI）的过程。超大规模集成电路的发展速度惊人，目前，一个微处理器乃至一个数字计算机系统完全可以集成在一片或数片VLSI之中。

在小规模集成电路时代，数字电路多采用“试凑法”，用固定逻辑功能的标准“积木块”搭成，即凭借其设计者的经验人工设计。所设计的电路，费时费力、器件多体积大、可靠性差及不能实时修改。到了20世纪60年代中期，人们相继开发出各种计算机辅助设计工具，主要有布线PCB（Printed Circuit Board）设计、电路模拟、逻辑模拟以及版图的绘制等，通过应用计算机，有效地将设计人员从繁琐重复的计算和绘图工作中解脱出来。这个时期的辅助电路设计工具一般被称为CAD（Computer Aided Design）设备。20世纪80年代出现了第二代CAD系统，之所以称为CAD系统，是源于该时期大量软件公司进入市场，提供了带原理图输入、模拟验证、逻辑综合、芯片布图及印刷电路板布图等功能的电子系统设计软件。这些软件按照设计、分析和测试等的不同阶段而分别使用，每个软件仅完成其中的一项任务，通过分别顺序地使用这些软件，即可完成整体设计。随着微电子技术与计算机图形学、拓扑学、逻辑学、计算数学、计算机辅助测试（CAM）、计算机辅助工程（CAE）等技术的发展融合，逐渐形成了电子系统设计自动化的概念——简称EDA（Electronic Design Automation）。

电子系统设计自动化就是应用计算机来完成电子系统的设计。应用EDA技术，设计者可以从高层次开始，使用标准化的硬件描述语言来描述被设计电路系统的行为特性，可以自顶向下将电路系统分割成各个子模块，分时分人设计。然后通过逻辑综合，即对不同层次和

不同形式的设计描述进行转换，可将电子系统的高层行为描述转换到低层硬件描述和确定的物理实现。电路系统的仿真、测试等均由计算机完成。EDA 工具都提供单元库、宏功能库甚至嵌入式模块，既能保证设计质量，又能提高设计效率。除此之外，现代的 EDA 工具还特别强调设计的可交互性和可再利用性。EDA 工具还在高速发展，未来的设计者仅需重视概念设计，而绝大多数设计中的技术问题都可以依赖 EDA 工具得以解决。

1.1.2 可编程逻辑器件的发展概述

可编程逻辑器件（Programmable Logic Device——简称 PLD），是可以由用户进行编程而实现所需逻辑功能的数字集成电路。PLD 的诞生，使人们告别了在大面积的印刷电路板上，用各种分离器件拼凑大规模逻辑电路的时代。集成电路制造厂商向市场提供已封装完毕的芯片，其逻辑功能却可以由用户自己使用 EDA 工具写入。从生产厂家来看，PLD 是通用集成电路，可以批量生产以降低成本。从用户角度来看，自己可以将设计好的电路“写入”芯片，使之成为专用集成电路（Application Specific Integrated Circuit——简称 ASIC）。由于这种方式给厂家和用户都带来了好处，因而发展特别迅速，目前已成为实现 ASIC 的一种重要手段。

随着亚微米 CMOS 集成电路制作技术的成熟和发展，可编程逻辑器件迅速地更新换代，特别是近几年发展起来的 CPLD 和 FPGA 格外引人注目。可编程逻辑器件以其现场设计、现场修改、现场实现的可达数万门级的数字系统单片化的应用优势，逐渐受到世界各地电子系统应用领域的设计工程师广泛的关注和欢迎。时至今日，电子系统设计自动化技术不再是 ASIC 技术领域的一个点缀和补充，而跃之成为电子应用（包括通讯技术、计算机应用、自动控制、仪器仪表）等领域广泛受欢迎的实用化技术，成为数字系统的科研实验、样机试制、小批量产品的即时实现的最佳途径。

现代可编程逻辑器件 CPLD 和 FPGA 的出现，使得电子设计工程师或科研人员有条件在实验室内快速方便地开发 ASIC，这些专用集成电路往往就是一个数字系统。而在用可编程逻辑器件设计数字系统时，传统的自底向上的设计方法常常难以高效率、高质量地完成，因此本书介绍一种自顶向下的模块化设计方法。这种改革不单纯是知识的更新，而是为培养高水平的国际化科技人才打下坚实的基础。

目前，可编程逻辑器件不仅具有可编程和可再编程能力，而且只要把器件插在系统内或者线路板上，就能对其进行编程或者再编程，这种技术称为在系统编程技术（In System Programmable），简称 ISP。ISP 技术打破了使用可编程逻辑器件必须先编程后装配的惯例，而可以先装配后编程，成为产品后还可反复编程。ISP 技术使得系统内硬件的功能像软件一样被编程配置，从而可以实时地进行灵活和方便的更改和开发，真正做到了硬件的“软件化”自动设计。

随着集成制造技术的不断发展，芯片的外尺寸越来越小，而集成度却越来越高，用常规的测试方法对印刷电路板及器件测试的难度也越来越大，而且也越来越重要。1985 年欧洲的联合测试行动组（Joint European Test Action Group——简称 JETAG），提出了边界扫描测试标准。后来由 IEEE 于 1990 年采用，定义为 IEEE 1149.1—1990 边界扫描测试技术规范，即 JTAG（Joint Test Action Group）。JTAG 标准提供了板极和芯片级的测试，通过定义输入、输出引脚，逻辑控制函数和指令，所有 JTAG 的测试功能都仅需一个四线或五线

的接口及相应的软件即能完成。

可编程逻辑器件的边界扫描测试结构可以在不使用物理测试探针的情况下测试引脚的连接，而且可以在器件正常工作时，捕获功能数据。器件的边界扫描单元可以从引脚或者从器件核心逻辑信号中捕获数据，并且在测试过程中所加入的测试数据将串行移入边界扫描单元，捕获的数据串行移出并可以在器件外部同预期的结果进行比较。

电子工业和信息产业的飞速发展，推动着电子产品朝着高速度、高集成度方向迅速发展，而且电子产品更新换代的周期极大地缩短，这样对电子设计自动化的要求也就越来越高。为了快速准确地设计复杂的电子系统，必须采用自顶向下的综合技术。自顶向下的综合技术可使得集成电路的设计者直接进行数字系统从行为描述到芯片描述的设计工作，行为描述一般不需要说明一个设计具体采用何种方式实现，行为描述通常要比低层次的结构描述简洁，而且易于编写与理解。综合工具一般包括从原理图输入和高层描述工具，逻辑仿真器及低层综合工具的一系列软件包。低层的综合工具对设计进行逻辑描述，并执行逻辑优化，器件映射，布局布线的网表优化，从而产生最终的设计结果。自顶向下的综合技术可接受数字系统硬件描述语言，并且能将综合结果实现设计过程的自动文档化。目前，IEEE 公布的硬件描述语言的工业标准是 VHDL（Very high speed IC Hardware Description Language）语言，设计者可采用 VHDL 语言描述作为设计的规范说明与需要说明，直接用 VHDL 语言描述作为设计文档。

VHDL 语言是一种多层次的硬件描述语言，并且设计的顶层描述可以非常简练，经过层层细化求精，最终成为可直接付诸生产的电路级或版图参数描述。VHDL 有良好的可读性，用 VHDL 书写的源文件，既是程序又是文档，既可作为技术人员之间交换信息的文件，又可作为合同签约者之间的文件。VHDL 语言的硬件描述与工艺无关，不会因工艺变化而使描述过时，与工艺技术有关的参数可以通过 VHDL 提供的属性加以描述，工艺改变时，只需修改相应程序中的属性参数即可。VHDL 还支持大规模设计的分解和已有设计的再利用，因此 VHDL 现已成为普遍采用的一种通用、标准、生命力强的硬件设计者的交流语言和设计工具。

目前的电子设计自动化软件一般都包含了符合 IEEE—1076 标准的 VHDL 高层综合工具，因此为可编程 ASIC 的设计带来了极大的方便。但是人们并没有满足支持 VHDL 行为语言描述阶段，能够支持框图概念级或系统功能流程图或者设计输入方式共存的自动处理、综合、优化的 EDA 工具已出现，必将进一步推动电子系统通用化设计技术的发展。

1997 年英国 Pilkington Micro—Electronic Limited 公司又推出了具有数字、模拟混合功能的现场可编程阵列 FPMA，采用一定的 EDA 软件，在该芯片上可以实现数字滤波器、锁相环及 D/A 和 A/D 转换等多种功能。同时还生产出了模拟型现场可编程阵列 FPAA，这种器件一出现，立即引起全球应用领域的广泛关注和兴趣。随着新器件、新技术、新工具的出现，电子设计的传统方式将被彻底打破。

§1.2 可编程逻辑器件的基本结构

在 20 世纪 70 年代中期，相继出现了可编程只读存储器，即 PROM（Programmable Read Only Memory）、可编程逻辑阵列，即 PLA（Programmable Logic Array）、可编程阵

列逻辑，即 PAL（Programmable Array Logic）、通用阵列逻辑，即 GAL（Generic Array Logic）等早期可编程逻辑器件。现在将这些早期可编程逻辑器件统称为简单可编程逻辑器件——SPLD（Simple Programmable Logic Device）。

SPLD 的结构简单、设计灵活、对其开发软件的要求也较低，但 SPLD 规模小，其寄存器、I/O 引脚、时钟资源的数目有限，并且缺少内部互联。现代可编程逻辑器件是在 SPLD 基本结构的基础上发展起来的，了解 SPLD 基本结构仍然具有十分重要的现实意义。

1.2.1 简单可编程逻辑器件（SPLD）的基本结构

典型的可编程逻辑器件（PLD）一般都是二级结构，如图 1－1 所示，第一级通常是由逻辑门构成的“与阵列”，第二级为由逻辑门构成的“或阵列”，“与阵列”和“或阵列”是电路的主体。电路由“与阵列”输入，进行“与”逻辑组合，形成乘积项，再由这些不同的乘积项通过“或阵列”构成所要求的逻辑函数输出。为了适应各种输入情况，“与阵列”的每个输入端，包括内部反馈信号的输入端均设有缓冲电路，目的是使输入信号具有足够的驱动能力，并产生原变量（如 A）和反变量（如 $\overline{A}$）两个互补信号。输入信号是在逻辑阵列中的布线通道上运行的。PLD 器件的“与、或阵列”结构，可以十分有效地利用硅片面积，而且对于逻辑电路的设计也非常方便。因为，任何组合逻辑函数均可化为标准“与—或”表达式，从而可用“与门—或门”二级电路实现。而任何时序电路又都是由组合电路加上触发器构成的，所以，PLD 器件的结构正是直接实现了“与或式”，即积之和。因此，PLD 器件的结构对实现数字电路具有普遍的意义。

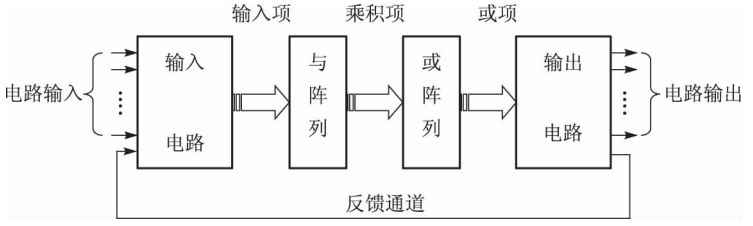


图 1－1 SPLD 的基本结构框图

PLD 器件根据结构的不同，其输入电路还包含锁存器及可以组态的宏单元，用于对输入信号进行预处理。PLD 的输出方式有多种，可以由“或阵列”直接输出，也可以通过寄存器输出，输出可以是低电平有效，也可以是高电平有效。无论采用何种输出方式，PLD 的输出电路端口上一般含有三态电路，并且有内部反馈通道可使输出信号反馈到“与阵列”的输入端。

简单 PLD 器件中，PROM 的结构是“与阵列”固定，“或阵列”可编程，其“与阵列”是一个向“或阵列”提供地址的全译码电路，由于阵列面积大、信号开关延迟时间长、工作速度低等原因，主要用作存储器。PLA 器件其结构为“与阵列”、“或阵列”皆可编程，但由于缺少高质量的支撑软件、价格昂贵等因素而未得到广泛应用。PAL 器件的特点是，“与阵列”可编程，而“或阵列”固定，其输出既有极性转换和 I/O 方式，还可用触发器输出，并且输出端对“与阵列”有内部反馈通道，加之触发器具有可上电清零及加密等功能，设计

者借助编程器即可自行编程。因而，可以说真正有效地利用 PLD 制作 ASIC 是从 PAL 开始的。

1.2.2 PAL 的原理与使用

1. PAL 器件的基本结构与命名

PAL16L8 器件的内部结构逻辑图如图 1－2 所示，在图 1－2 中可编程的“与阵列”是以阵列形式出现的，而固定的“或阵列”则是沿用“或”门的画法。“与阵列”中每条纵线均代表一个输入信号，每条横线对应一个“与”门，即代表一个乘积项。PAL16L8 器件的“与阵列”总共有 8 个输入端（见图 1－2 的 2～9 引脚），每个输入变量均由输入缓冲器产生原、反两个互补变量，对应于 16 个输入变量，所以有 32 条纵线，形成 64 个乘积项，并且这 64 个乘积项被分为 8 组，每组通过一个“或”门构成一个输出函数，因而 PAL16L8 器件共有 8 个输出函数，如图 1－2 的 12～19 引脚。所有的输出函数都由三态反相器引至输入端，且为低电平有效。PAL 器件的型号命名恰好描述了这几个参数，见图 1－3。

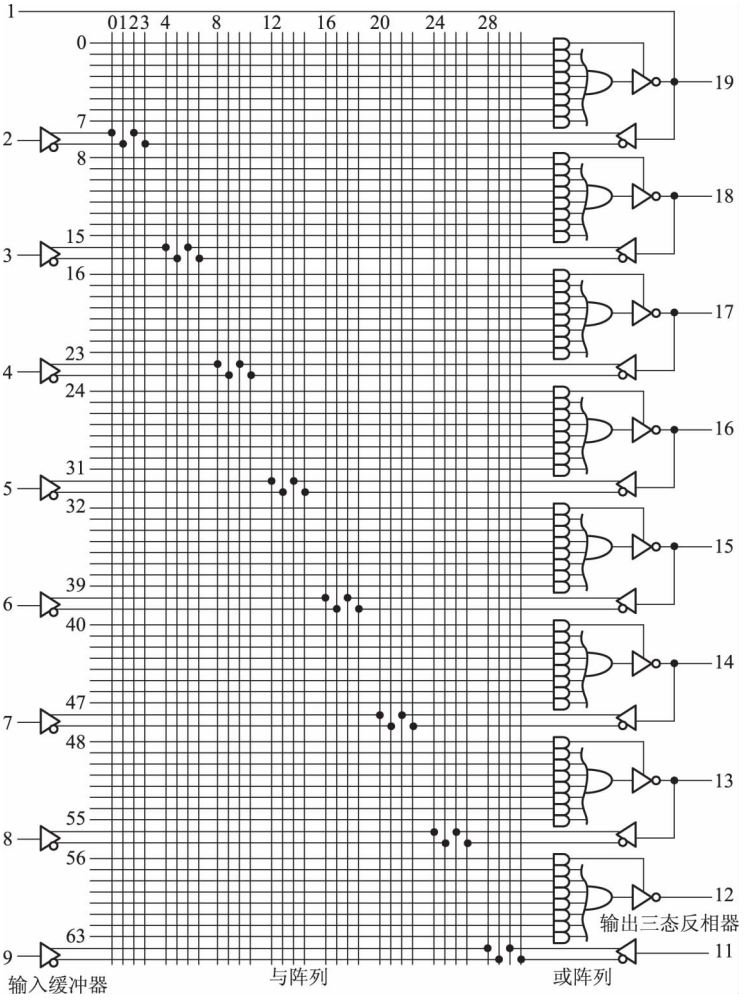


图 1－2 PAL16L8 的逻辑图

需要全本请访问 www.ertongbook.com

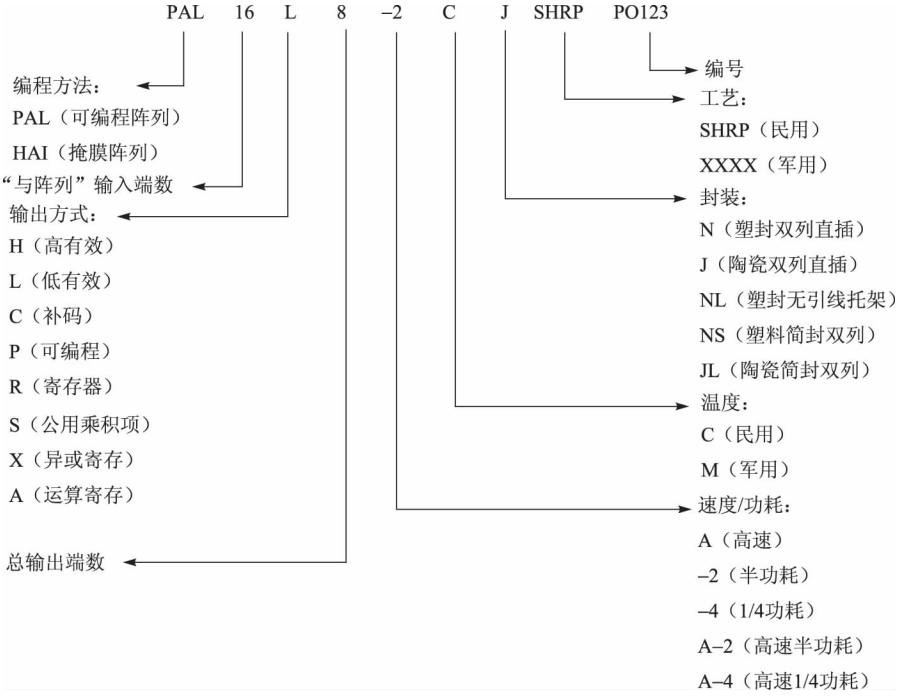


图 1－3　PAL 器件型号命名图

PAL 器件的品种很多，但主要分 20 引脚和 24 引脚两大类，此外还有 40 和 80 引脚的宏 PAL 器件。除了按电路的功能分类外，每一系列又分为标准型、半功耗和四分之一功耗型。按运行速度又可分为标准速、高速和超高速。按工艺可靠性又可分为军用和民用等。

PAL 器件的种类很多，但其基本门阵列结构是类似的，仅门阵列的规模有所差异。不同类型的 PAL 器件主要区别在于其输出和反馈结构。根据 PAL 器件的输出和反馈结构的不同，可以把 PAL 分为以下几种基本类型。

（1）基本阵列结构

这种类型的 PAL 器件内部只有“与阵列”和“或阵列”，其输出有三种配置形式：一是输出“高”电平有效；二是输出“低”电平有效；三为“互补”输出。基本阵列结构的 PAL 器件输出结构图如图 1－4 所示，基本阵列结构的 PAL 器件适合于构成组合逻辑电路。

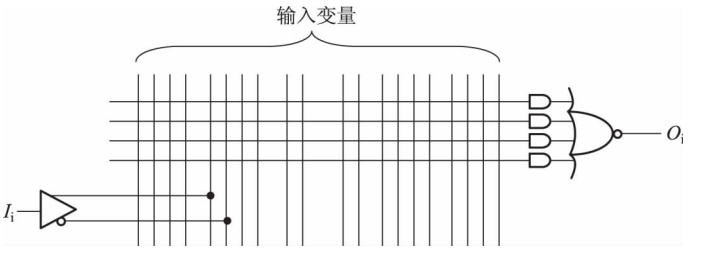


图 1－4　基本阵列输出结构

（2）可编程的输入、输出结构

PAL16L8 即属于这种输出结构，其输出结构示意图如图 1－5 所示。在图 1－5 中，

每个输出函数最多可包含七个乘积项，最上端的“与”门是用来控制三态反相器的，当该“与”门输出为“1”时，相对应的输出函数才能通过三态反相器输出，这时一则相应“或”门的输出函数可送到对应的引脚，二来该输出还可通过右边的缓冲电路反馈到“与阵列”。当“与”门输出为“0”时，三态反相器处于高阻状态，“或”门与对应引脚间的联系被隔断，但却可以从该引脚经由缓冲器向“与阵列”输入外信号。因此，该引脚是一个 I/O 端口。另一方面，由于输出是受控制的，因而整个阵列的 8 个逻辑函数的输出时间便可以不一致，此性能被称为“异步”。所以，图 1－5 的输出结构又被称为异步 I/O 输出结构。

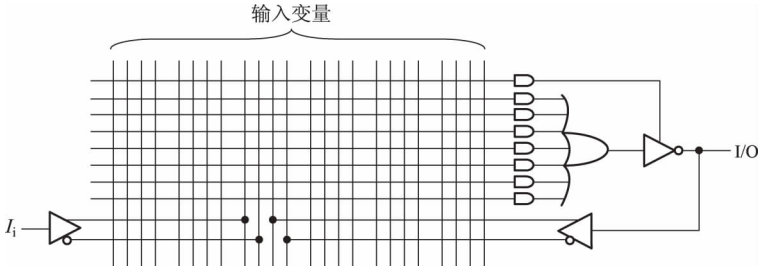


图 1－5　可编程输入、输出结构

（3）寄存器输出结构

该结构在“或”门后面是一个上升沿触发的 D 触发器，如图 1－6 所示，触发器的 $\overline{Q}$ 端通过缓冲电路反馈回“与阵列”，该反馈功能使这类 PAL 器件能记忆先前的状态。从图 1－6 可知，“或”门的输出端没有向“与阵列”的反馈，三态门反相器是由外输入信号控制的。这种结构将八个乘积项的“和”输出，D 触发器由系统时钟控制，因而这类 PAL 器件适用于实现时序逻辑功能，如计数、移位和分支等功能。PAL16R8 就是这种输出结构。

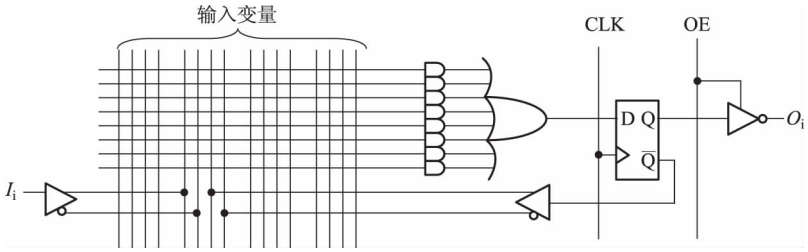


图 1－6　寄存器输出结构

（4）“异或”输出结构

这种结构将阵列中的“与－或”项分为两部分，再对它们进行“异或”后输入到 D 触发器，当系统时钟的有效沿到达后，将信号存入 D 触发器内，如图 1－7 所示。该种结构对于实现诸如计数器和状态机的保持操作提供了简易的实现方法。

（5）异步寄存器输出结构

该种结构其输出极性可由“异或”门编程设置，并且可在每组输入变量中设置一个时钟乘积项（CLK），用来作为 D 触发器的时钟信号。经过编程使每个触发器获得单独的时钟信号，从而实现异步时序逻辑电路。另外，选通乘积项（OE）和选通端（OC）共同控制三态

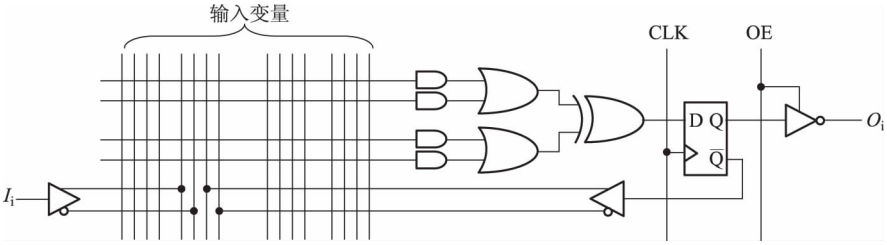


图 1-7 “异或”输出结构

输出缓冲器的使能端；寄存器的异步复位（R）和置位（S）端分别由复位（RP）和置位（ST）两个乘积项独立控制，见图 1-8 所示。

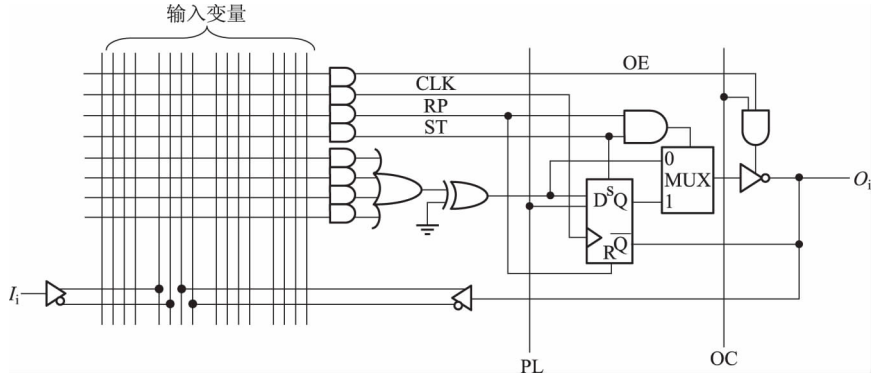


图 1-8 异步寄存器输出结构

该种结构的另一特点是，接在寄存器和输出缓冲器之间有一个多路选择器（MUX）。MUX 的选通信号来自乘积项（ST）和（RP），当 $ST \cdot RP = 0$ 时，MUX 对寄存器旁路，这时“异或”门的输出将直接送到三态输出缓冲器，以便实现组合逻辑电路；当 $ST \cdot RP = 1$ 时，MUX 选通 D 触发器，将 D 触发器的输出 Q 接至三态输出缓冲器，实现同步时序逻辑电路。

异步寄存器输出结构是 PAL 器件中应用最为灵活的一种，既可实现异步时序逻辑，又可实现同步时序逻辑，还可实现组合逻辑。

可编程的“与阵列”使得 PAL 具有许多输入端，而固定的“或阵列”又使得 PAL 的体积小、速度快。

2. PAL 器件的使用

使用 PAL 器件的关键在于将逻辑电路的输出函数、触发器的激励输入都写成“与－或”表达式。这一点对于组合逻辑电路是不成问题的，因为任何一个组合逻辑函数都可以用逻辑代数或者卡诺图化简成若干个最小项的“和”，即标准“与－或”表达式，而且这种形式是唯一的。

对于时序逻辑电路，可根据其特征方程将触发器的激励输入写成标准“与－或”表达式，例如：带有置数（S）和清零（R）控制端的 JK 触发器，先写出其特征方程为：

$$Q^{n+1} = S + \overline{R} \cdot \overline{K} \cdot Q^n + \overline{R} \cdot J \cdot \overline{Q^n}$$

$$\overline{Q}^{n+1}=R+\overline{S}\cdot\overline{J}\cdot\overline{Q}^{n}+\overline{S}\cdot K\cdot Q^{n}$$

因而，可用两个 D 触发器分别输出，即令它们的激励信号为：

$$D_1=S+\overline{R}\cdot\overline{K}\cdot Q^{n}+\overline{R}\cdot J\cdot\overline{Q}^{n}$$

$$D_2=R+\overline{S}\cdot\overline{J}\cdot\overline{Q}^{n}+\overline{S}\cdot K\cdot Q^{n}$$

从而可以用时序型 PAL 器件实现。

总而言之，任何逻辑电路，只要其输入变量的个数、每个方程的乘积项数、总的“或门”数、总的寄存器数和输出端数不超过 PAL 器件所能提供的范围，基本都可以用 PAL 器件实现。

3. PAL 器件的特点

PAL 器件将可编程的灵活性和接口功能的可选择性结合在一起，因而在数字逻辑设计领域占有独特的地位。其主要特点有如下几个方面：

① 可以直接取代标准逻辑器件。通常一片 PAL 器件可代替 4～12 片 SSI 或 2～4 片 MSI。

② 设计灵活。改变了传统数字逻辑设计中，采用中小规模集成电路（SSI/MSI）和微处理器（CPU）来实现的弊端，即采用 SSI/MSI 时所需的逻辑繁杂，以致行不通；而用微处理器（CPU）芯片时，使用效率又不高。

③ 能满足高速度、高性能的要求。由于 PAL 器件的高速度，因而易于实现大规模逻辑运算和高速控制程序，还可用于微处理器中高速数据传送接口。

④ 有加密功能。PAL 器件可采用标准 PROM 编程器快速编程，PAL 中有可编程的加密位，一旦对其编程，PAL 仍能执行所设计的逻辑功能，读写仪器却无法读出器件内部的设计内容，即可防止非法复制。

虽然相对于标准 TTL 电路而言，PAL 器件有许多优点，但是仍有很大的局限性，主要缺点是其集成度低，并且由于 PAL 器件的输出结构是固定的，即不能重新组态，因此编程灵活性较差。另外，PAL 器件采用的是 PROM 工艺，只能一次性编程。一般来说，PAL 器件仅能用来实现组合逻辑电路、计数器和移位寄存器，而大多数时序机因功能复杂，用 PAL 器件往往是难以胜任的。

1.2.3　GAL 的原理与特点

GAL 器件是在 PAL 器件结构的基础上产生的新一代器件，它的结构直接继承了 PAL 器件的“与－或”结构，但它的输出结构增加了输出宏单元（Output Logic Macro Cell，简称 OLMC），这些输出宏单元可由设计者通过编程进行不同模式的组合，即可重组态功能，从而为设计者提供了高度的灵活性。

GAL 器件采用了 E^2PROM 浮栅技术和高性能的 E^2COMS 工艺，具有电可擦除、重复编程（重复编程次数约为 100 左右）、编程速度快（约为秒量级）、数据保持时间长（可达 20 年）、功耗低（约为 45 mA）、工作速度快（读、写速度约 12～25 ns）等优点。

GAL 器件型号的命名规则与 PAL 器件相同，如 GAL16V8 中的 16 表示“与阵列”中的输入变量数，8 表示输出端数，V 则是输出方式可以改变的意思。

1. GAL 器件的基本结构

图 1－9 是 GAL16V8 的内部逻辑电路图，从图 1－9 中可知，GAL 器件的“与阵列”和

PAL 器件是相似的，但“或阵列”以及输出寄存器均被输出宏单元（OLMC）所取代了。所有系列的 GAL 器件，其输出端均设有输出逻辑宏单元（OLMC），GAL 器件作为比 PAL 器件新一代的 PLD，其许多优点都源自于 OLMC。

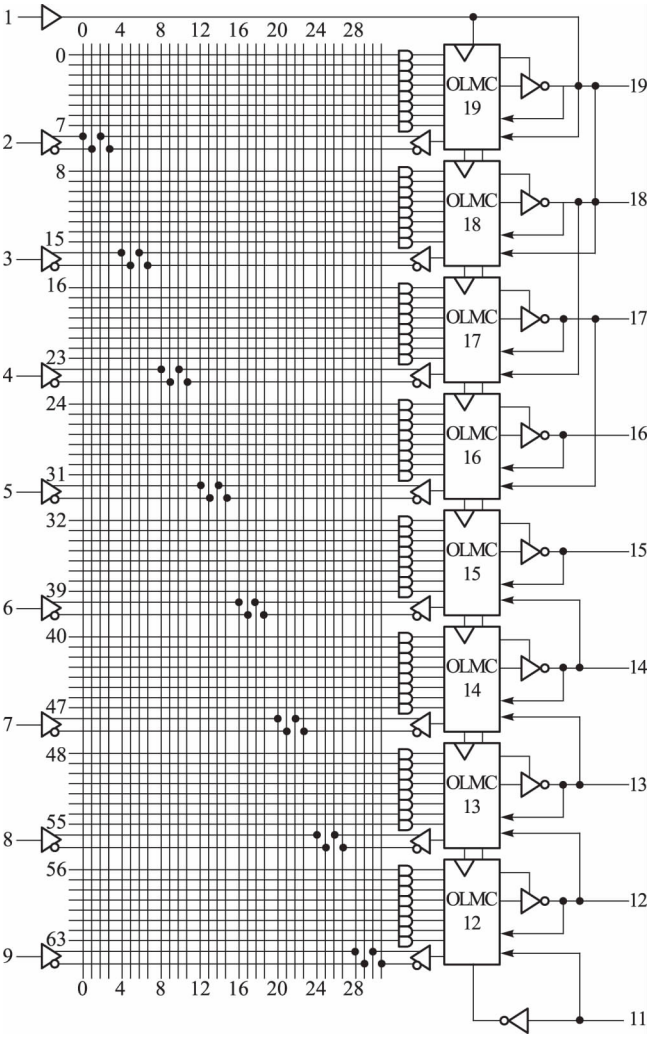


图 1－9　GAL16V8 的逻辑图

图 1－10 是输出逻辑宏单元的内部逻辑电路图，由图 1－10 可以看出，被输出逻辑宏单元所替代的“或”门和 D 触发器仍然是 OLMC 的核心，但多出了四个多路开关，即 MUX。它们分别是：乘积项多路选择器（PTMUX）、输出多路选择器（OMUX）、输出允许控制多路选择器（TSMUX）和反馈源多路选择器（FMUX）。

乘积项多路选择器（PTMUX）是二选一多路选择器，PTMUX 的两个输入端，一个是“地”信号；一个是来自“与”阵列中的第一个乘积项。由结构控制字来决定是将这个乘积项用作“或”门的输入信号，还是将“地”信号用作“或”门的输入信号。

输出多路选择器（OMUX）也是二选一多路选择器，它的两个输入端，一个是“异或”门的输出；一个是 D 触发器的输出。用 OMUX 来选择是寄存器输出，还是组合电路输出。

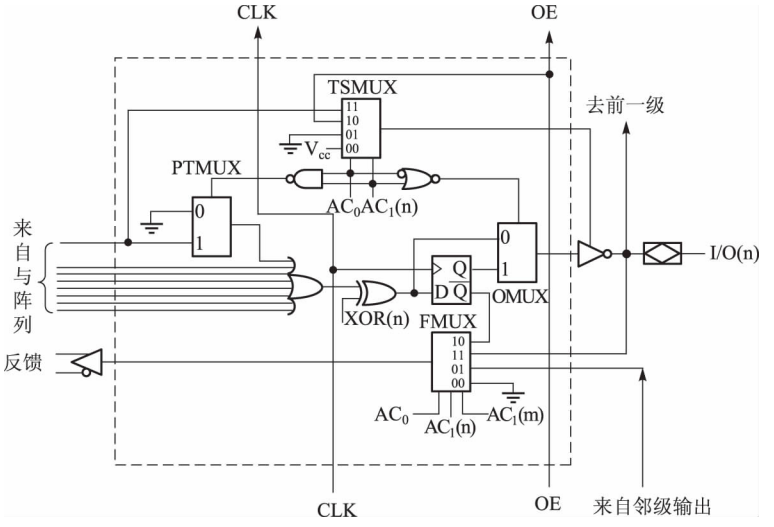


图 1－10　OLMC 的结构图

输出允许控制多路选择器（TSMUX）**为四选一多路选择器，其作用是控制输出三态缓冲器的使能端，根据结构控制字可提供四种使能控制：① 由 V_{CC} 控制使能端；② 由“地”电平控制使能端；③ 由 OE 控制使能端；④ 由来自“与”阵列的第一个乘积项控制使能端。**

反馈源多路选择器（FMUX）**仍为四选一多路选择器，其作用是确定反馈信号的来源。可有四种反馈信号供选择：① 来自 I/O 引脚；② 来自 D 触发器的 $\overline{Q}$ 端；③ 来自相邻级的输出；④ 接“地”信号。**

2. GAL **器件的工作原理**

GAL **器件的各种组态配置是由图 1－11 所示的结构字来控制的：**

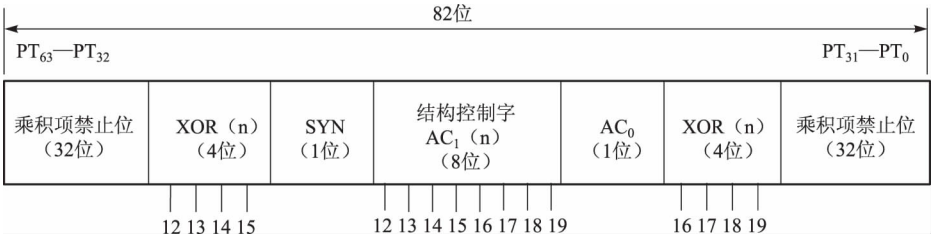


图 1－11　GAL16V8 的结构控制字

① SYN——**同步位，该字为全局位，即该位置为所有宏单元所共有。当 $SYN=1$ 时，器件是组合工作方式；当 $SYN=0$ 时，器件是时序工作方式。**

② AC_0——**结构控制位**（1 **位**），**全局位。该字与各个** OLMC（n）**各自的** AC_1（n）**配合，以控制各个多路选择器。**

③ AC_1——**结构控制字**（8 **位**），**局部位。每个** OLMC（n）**有各自的** AC_1（n），**这里**（n）**对应** OLMC **的输出端编号**（**对于** GAL16V8，n＝12～19；**对于** GAL20V8，n＝15～22）。

④ XOR（n）——**极性控制位**（8 **位**），**局部位。每个** OLMC（n）**有各自的** XOR（n），

需要全本请访问 www.ertongbook.com