

高等学校电子类系列教材

EDA技术 简明教程

赵 刚 等编著

02

四川大学出版社



EDA 技术简明教程

赵 刚 张志亮
张 菁 植 涌 编著

四川大学出版社

责任编辑:周树琴

责任校对:李文庆

封面设计:罗 光

责任印制:李 平

图书在版编目(CIP)数据

EDA 技术简明教程 / 赵刚等编著. —成都: 四川大学出版社, 2004.6

ISBN 7-5614-2843-X

I. E... II. 赵... III. 电子电路—电路设计: 计算机辅助设计—高等学校—教材 IV. TN702

中国版本图书馆 CIP 数据核字(2004)第 059340 号

内容简介

本书旨在使读者掌握利用现代计算机技术来高效设计并实现一块芯片、一个电路,甚至整个系统的基本思想和现代方法,力求简明、通俗、实用。

全书分为上、中、下三篇,共 10 章。上篇,芯片级 EDA 技术,包括:可编程逻辑器件, VHDL, MAX+plus II 开发软件, Quartus II 开发软件, Synplify Pro 综合器;中篇,电路级 EDA 技术,包括:EWB 电路仿真软件, Prote199 印刷电路板软件;下篇,系统级 EDA 技术,包括: SystemView 系统仿真软件, Simulink 动态仿真软件, VHDL 代码自动生成。

本书可作为通信工程、电子信息科学与技术、计算机科学与技术、微电子、仪器仪表、工业自动化等专业的本科生教材,以及电路与系统、信号处理、通信工程专业的硕士生的教材或参考书,亦可作为从事电子信息产业的产品研发人员、工程技术人员的参考书。

书名 EDA 技术简明教程

编 著 赵刚 等

出 版 四川大学出版社

地 址 成都市一环路南一段 24 号 (610065)

印 刷 成都宏明印刷厂

发 行 四川大学出版社

开 本 787mm×1 092mm 1/16

印 张 19.25

字 数 460 千字

版 次 2004 年 7 月第 1 版

印 次 2004 年 7 月第 1 次印刷

印 数 0001~3 000 册

定 价 31.00 元

◆读者邮购本书,请与本社发行科
联系。电 话:85408408/85401670/
85408023 邮政编码:610065

◆本社图书如有印装质量问题,请
寄回出版社调换。

◆网址:www.scupress.com.cn

版权所有◆侵权必究

前 言

EDA (电子设计自动化) 技术是国家教育部规定的综合性大学电子信息类专业的专业主干课, 是通信工程、电子信息科学与技术、计算机科学与技术、微电子、仪器仪表、工业自动化等专业必修的专业主干课。本书是作者在四川大学电子信息学院讲授“EDA 技术”课程的基础上, 总结多年的教学实践经验, 以原有的《数字系统的 VHDL 设计及 ASIC 实现》一书为底本, 并结合近年来国内外 EDA 领域所取得的最新发展成果, 反复精炼而成。

书中第 1 章讲述了常用可编程逻辑器件的内部结构及工作原理。第 2 章讲述了 VHDL 硬件描述语言, 包括程序的基本结构、数据类型、描述方式及常用数字电路的设计方法。第 3 章至第 5 章分别讲述了目前常用的 PLD 开发系统软件的使用方法。第 6 章讲述了数字/模拟电路设计仿真软件的使用方法。第 7 章讲述了电路原理图和印刷电路板图设计软件的使用方法。第 8 章和第 9 章讲述了电子系统设计仿真软件的使用方法。第 10 章介绍了目前最新的自顶向下的 PLD 开发实现流程。

本书有四个主要特点:

1. 覆盖面广。现代 EDA 技术覆盖了从系统级、电路级、芯片级到版图级四个设计层次, 书中对前三个层次内容进行了详尽介绍, 读者通过本课程的学习, 能较全面地掌握 EDA 技术。

2. 内容新颖。随着计算机技术的飞速发展, EDA 技术中的理论方法和技术手段日新月异。为了让读者掌握最新的 EDA 技术, 并考虑到国内高校在实验条件方面的实际情况, 书中所介绍的各种 EDA 工具均采用目前在实际产品开发中被广泛使用、具有代表性的开发系统软件和流程。

3. 要点突出。在本科教学实践中, 丰富的 EDA 内容与有限的教学学时产生了严重的矛盾。本书在对 EDA 工具讲述时, 采用“功能简介—开发流程—库资源—单元设计实例—综合设计实例”为主线的引导性教学方式, 通过突出教学

要点, 达到有效控制学时的目的。

4. 注重实效。EDA 技术课程侧重于电子系统的设计及优化, 是一门实践性很强的技术课程。为了培养读者的工程实践能力, 书中提供了大量典型的设计范例, 以帮助初学者尽快入门。

本书若作为本科生“EDA 技术”课程的教材, 可安排 72 学时 (含实验课), 从第 1 章到第 10 章, 学时分配依次为: 8, 8, 8, 8, 4, 8, 8, 8, 8, 4。如作为专科生“EDA 技术”课教材, 主要讲授第 1、第 2、第 3、第 6、第 7 及第 8 章。作为电路与系统、信号处理、通信工程专业的硕士生和工程硕士的参考教材时, 应增大第 4、第 8、第 9 及第 10 章的学时数, 以便安排更多的实验上机时间。

本书亦可作为从事电子信息产业的产品研发人员、工程技术人员的参考书。

本书是由参加编写的教师集体讨论、分工编写、交叉修改后完成的。参加编写的主要人员有赵刚、张志亮、张菁、植涌。本书由赵刚担任主编, 并负责大纲拟定、组织编写与统稿工作。

本书在编写过程中, 得到了四川大学教务处、四川大学电子信息学院、美国 Altera 公司中国 (西南地区) PLD 技术培训中心的大力支持、帮助和鼓励。借本书出版之际, 向他们表示衷心的感谢!

本书在出版过程中, 得到了四川大学出版社领导及周树琴老师的诸多帮助, 在此表示诚挚的谢意!

限于编者水平, 书中难免有不妥和错误之处, 恳请读者批评指正 (Email: dftczhao@mail.sc.cninfo.net)。

编著者

2004 年 6 月于四川大学

目 录

上篇 芯片级 EDA 技术

第 1 章 可编程逻辑器件	(3)
1.1 基于“乘积项”(Product - Term)的可编程逻辑器件	(3)
1.1.1 乘积项和与或阵列	(3)
1.1.2 PROM	(4)
1.1.3 PLA	(7)
1.1.4 PAL	(9)
1.1.5 GAL	(11)
1.1.6 CPLD	(13)
1.1.7 非易失性可编程元件的编程原理	(16)
1.2 基于“查找表”(Look - Up Table)的可编程逻辑器件	(19)
1.2.1 SRAM	(19)
1.2.2 FPGA	(20)
1.2.3 FPGA 与 CPLD 使用上的比较	(24)
习 题	(25)
第 2 章 VHDL 硬件描述语言	(26)
2.1 概述	(26)
2.2 VHDL 程序基本结构	(27)
2.2.1 实体(Entity)说明	(28)
2.2.2 构造体(Architecture)	(29)
2.2.3 库(Library)	(30)
2.3 VHDL 语言的基本数据类型和操作符	(31)
2.3.1 数据对象及其分类	(31)
2.3.2 数据类型	(32)
2.3.3 运算操作符	(34)
2.4 VHDL 构造体的描述方式	(38)
2.4.1 顺序描述语句	(38)
2.4.2 并行描述语句	(49)

2.5 基本逻辑电路设计	(57)
2.5.1 组合逻辑电路设计	(57)
2.5.2 时序逻辑电路设计	(60)
习 题	(68)
 第 3 章 MAX + plus II 开发软件及其使用	(70)
3.1 MAX + plus II 开发软件简介	(70)
3.2 MAX + plus II 开发流程	(70)
3.3 MAX + plus II 的原理图编辑器与库资源	(72)
3.3.1 1 位半加器的设计	(72)
3.3.2 1 位全加器的设计	(86)
3.4 使用 VHDL 进行设计	(88)
3.5 MAX + plus II 上的设计实例:简易电子琴设计	(90)
3.5.1 简易电子琴概述	(90)
3.5.2 源文件的编辑	(91)
3.5.3 项目的编译	(94)
3.5.4 系统仿真	(95)
3.5.5 目标芯片的编程实现	(97)
习 题	(97)
 第 4 章 Quartus II 开发软件及其使用	(99)
4.1 Quartus II 开发软件简介	(99)
4.2 Quartus II 开发流程	(100)
4.3 Quartus II 的库资源	(100)
4.4 Quartus II 上的设计实例	(101)
4.4.1 利用向导建立新工程	(101)
4.4.2 源文件的编辑	(103)
4.4.3 设置编译选项	(105)
4.4.4 启动编译过程	(106)
4.4.5 设计仿真	(106)
4.4.6 目标芯片的编程实现	(110)
4.5 使用框图进行设计	(113)
习 题	(119)
 第 5 章 Synplify Pro 综合器	(120)
5.1 Synplify Pro 综合器简介	(120)
5.2 Synplify Pro 的基本使用	(121)
5.3 有限状态机编译器的使用	(129)

5.4 利用 MAX + plus II 进行适配	(133)
5.5 利用 Quartus II 进行适配	(135)
5.6 设计优化技术	(138)
5.6.1 资源共享	(138)
5.6.2 流水线设计	(141)
5.6.3 寄存器配平	(145)
习 题	(147)

中篇 电路级 EDA 技术

第 6 章 EWB 数模混合电路仿真软件	(153)
6.1 EWB 简介	(153)
6.2 EWB 的运行环境介绍	(154)
6.2.1 主菜单与工具栏	(154)
6.2.2 元件库	(157)
6.2.3 虚拟仪器库	(163)
6.2.4 仿真控制	(167)
6.3 模拟电路的仿真分析	(168)
6.3.1 电路原理图的输入	(168)
6.3.2 电路仿真	(169)
6.3.3 电路分析	(171)
6.4 数字电路的仿真	(187)
6.5 子电路的生成与使用	(187)
习 题	(189)
第 7 章 Protel99 SE 印刷电路板设计软件	(192)
7.1 Protel99 SE 简介	(192)
7.2 印刷电路板设计流程	(192)
7.2.1 设计准备	(192)
7.2.2 参数设置	(192)
7.2.3 加载元件封装库与网络表	(192)
7.2.4 布局和布线	(193)
7.2.5 存盘和输出	(193)
7.3 电路原理图设计	(193)
7.3.1 进入 Protel99 SE 集成环境	(193)
7.3.2 创建新的设计数据库	(193)
7.3.3 在设计数据库中创建新的文档	(195)
7.3.4 原理图编辑	(197)

7.4 生成网络表文件	(202)
7.5 电路板规划	(204)
7.6 加载封装信息库与载入网络表	(209)
7.6.1 加载封装信息库	(209)
7.6.2 载入网络表	(210)
7.7 元件布局	(212)
7.7.1 自动布局	(212)
7.7.2 手工布局	(214)
7.8 布线	(215)
7.8.1 自动布线	(215)
7.8.2 手工布线	(218)
7.9 电路板 3D 预览	(218)
习 题	(219)

下篇 系统级 EDA 技术

第 8 章 SystemView 系统仿真软件	(223)
8.1 SystemView 简介	(223)
8.1.1 用户界面灵活友好	(223)
8.1.2 库资源丰富	(223)
8.1.3 仿真分析与数据处理功能强大	(223)
8.1.4 硬件设计接口灵活	(223)
8.1.5 自我诊断功能完善	(224)
8.1.6 方便的 APG 功能	(224)
8.2 SystemView 的运行环境	(224)
8.2.1 系统窗口	(224)
8.2.2 分析窗口	(236)
8.3 SystemView 的设计仿真示例	(241)
8.3.1 正弦波的平方	(241)
8.3.2 AM 超外差接收机	(246)
习 题	(250)
第 9 章 Simulink 动态仿真系统	(252)
9.1 Simulink 简介	(252)
9.2 Simulink 基本使用方法	(252)
9.2.1 创建模型	(253)
9.2.2 在模型中添加模块	(253)
9.2.3 修改各模块的参数属性	(254)

9.2.4 连接模块完成模型	(256)
9.2.5 系统仿真	(257)
9.3 Simulink 子系统	(258)
9.4 Simulink 基本模块简介	(260)
9.5 Simulink 综合实例:数字信号载波传输系统仿真设计	(269)
习 题	(274)
 第 10 章 VHDL 代码自动生成	(275)
10.1 概述	(275)
10.1.1 传统的 FPGA 开发流程	(275)
10.1.2 VHDL 代码自动生成流程	(276)
10.2 DSP Builder 自动生成代码工具	(278)
10.3 基于 DSP Builder 的设计流程	(280)
10.4 DSP Builder 的库资源	(289)
10.5 DSP Builder 应用实例	(295)
10.5.1 FIR 滤波器	(295)
10.5.2 FFT	(296)
10.6 小结	(296)
习 题	(297)
 参考文献	(298)

上篇 芯片级EDA技术

本篇主要讲述芯片级 EDA 技术。随着工艺技术的发展、集成度的提高和价格成本的下降,可编程逻辑器件 PLD (Programmable Logic Device) 得到了越来越多的应用,并使得 SPC (System on a Programmable Chip, 可编程片上系统) 成为现实,当今世界上最大的两家可编程器件生产商 Altera 和 Xilinx 均可提供该类的解决方案。使用单个的可编程逻辑器件,不仅可以取代原来众多的 74 系列、4000 系列中小规模数字集成电路器件,实现从简单到复杂的数字逻辑系统,还可以实现高速的语音处理、图像处理、通信等数字信号处理,甚至可以在其上实现多个用户定制的 CPU 并行运算与控制。可编程逻辑器件已经成为现今数字系统实现的一大有力工具。

本篇首先介绍可编程逻辑器件的发展历程以及 Altera 公司的典型器件结构与特性,然后介绍芯片设计和比较常用的硬件描述语言 VHDL;接着介绍 Altera 公司的两套集成开发工具 MAX+plus II 与 Quartus II;最后针对 MAX+plus II 综合器性能的不足,补充介绍了现今比较流行的一款逻辑综合优化工具 Synplify Pro。读者学习完本篇各章的内容后,可基本掌握可编程逻辑器件的设计流程与开发技术。

第1章 可编程逻辑器件

可编程逻辑器件 PLD (Programmable Logic Device) 是 20 世纪末出现的一种新型大规模电子集成电路。它的出现,使电子工程师能在实验室甚至家庭里制造出用户自行设计的专用集成电路 ASIC (Application Specific Integrated Circuits)。可编程逻辑器件经历了从 PROM、PLA、PAL、GAL、EPLD 到 CPLD 和 FPGA 的发展过程,在结构、工艺、集成度、功能、速度和灵活性方面逐步得到了改进和提高。

可编程逻辑器件的发展大致经历了如下过程:

① 20 世纪 70 年代中期,熔丝编程的 PROM 和 PLA 器件是最早的可编程逻辑器件。

② 20 世纪 70 年代末期,AMD 公司开始推出 PAL 器件。

③ 20 世纪 80 年代初期,Lattice 公司发明电擦写的,使用上比 PAL 更灵活的 GAL 器件。

④ 20 世纪 80 年代中期,Xilinx 公司提出现场可编程概念,同时生产出世界上第一片现场可编程门阵列 FPGA(Field Programmable Gate Array)器件。同一时期,Altera 公司推出可擦除的可编程逻辑器件 EPLD(Erasable PLD,也称为早期的 CPLD),较 GAL 器件有更高的集成度,可以用紫外线或电擦除。

⑤ 20 世纪 80 年代末,Lattice 公司又提出在系统可编程 ISP(In - System Programmability)技术,并且推出了一系列具有在系统可编程能力的复杂可编程逻辑器件 CPLD (Complex Programmable Logic Device)。

近年来,可编程逻辑器件(CPLD/FPGA)已在工业上得到了广泛应用,市场份额逐年提高。随着制造工艺的进步,可编程逻辑器件正向高密高速方向迅速发展,使其在一块芯片内实现一个完整的电子系统 SC(System on Chip,片上系统)成为可能。目前,各大集成电路厂商正在大量地研究模拟可编程器件,在不远的将来,数模混合可编程器件将使电子设计及实现技术发生质的飞跃。

1.1 基于“乘积项”(Product - Term)的可编程逻辑器件

1.1.1 乘积项和与或阵列

乘积项是布尔代数的一个术语,指与运算的输出项。

具体实现过程如下:输入信号首先通过一个与矩阵,产生一系列输入信号相与的组合,每组组合都称为一个乘积项,然后这些乘积项在或矩阵中相加,再经输出单元或宏单元输出。

基于乘积项的可编程逻辑器件包含两个基本部分:一个是逻辑阵列,另一个是输出单元或宏单元(Macrocell)。逻辑阵列是用户可编程的部分,它由与矩阵、或矩阵和反相器所组成。宏单元的作用是使设计者能改变 PLD 的输出结构。

与或这种结构可直接实现任何以积之和形式表达的逻辑,任何组合逻辑功能都可以通过采用卡诺图和摩根定理的方法得到积之和的逻辑方程。

与或阵列的结构可以通过改变与或阵列的连接实现不同的逻辑功能。无论改变与阵列还是或阵列的连接,都可以使所实现的逻辑函数发生变化。表 1.1 给出各种器件对应的与或阵列结构:

表 1.1 各种 PLD 器件对应的与或阵列结构表

器件类别	与或阵列结构
PROM	与阵列固定、或阵列可编程
PLA	与阵列、或阵列均可编程
PAL、GAL、EPLD、CPLD	与阵列可编程、或阵列固定

随着对 PLD 器件研究的深入,前两种形式的与或阵列结构暴露出一定的缺陷。第 1 种结构的器件在输入数目增加时,与阵列的输出信号线数目以 2 的级数增加;第 2 种结构的器件制造工艺复杂,器件工作速度慢。

基于前两类与或阵列结构的 PLD 器件处于被淘汰的边缘,相对地,第 3 种形式具备一定的技术优势,是 PLD 目前发展的主流。

【例 1.1】 试用与或阵列实现一位全加器。

解:一位全加器的输入变量为加数 A_n, B_n 和低位进位 C_n ,输出变量为和数 S_n 及进位数 C_{n+1} 。经过化简的最简与或表达式的逻辑方程如下:

$$S_n = \bar{A}_n \bar{B}_n C_n + \bar{A}_n B_n \bar{C}_n + A_n \bar{B}_n \bar{C}_n + A_n B_n C_n$$

$$C_{n+1} = A_n B_n + A_n C_n + B_n C_n$$

可用一个 3 输入/2 输出的与或阵列实现该逻辑函数,如图 1.1 所示。

与或阵列的与阵列输入线、乘积项、或门和输出线等在具体器件内都是有限的,被称为可编程逻辑器件的资源。上面这个例子中,使用的资源有 3 根与阵列输入线、7 个乘积项、1 个四输入或门和 1 个三输入或门。

与或阵列在 PLD 器件中只能实现组合电路功能,时序电路的功能需由包含触发器或寄存器的宏单元来实现。

1.1.2 PROM

只读存储器 ROM(Read - Only Memory)里的数据只能读出,不能写入。工作时,在地址码

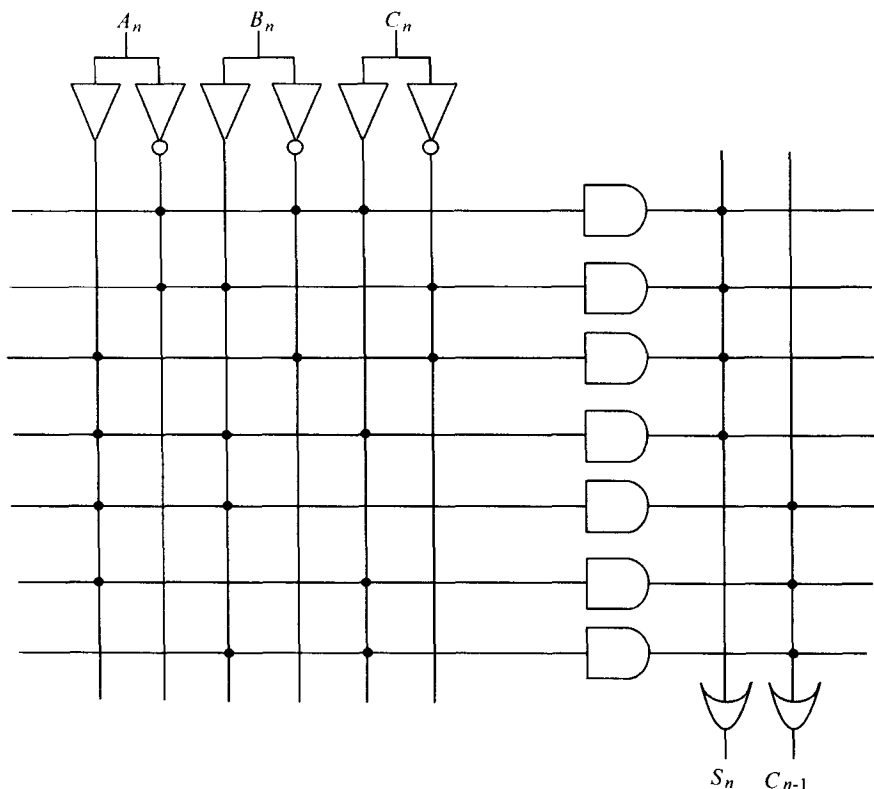


图 1.1 与或阵列实现的全加器

输入端输入一个确定的地址信号,便可以在输出端得到一个事先存入的确定数据。它的最大优点是具有非易失性,即使停止供电,ROM中存储的信息也不会丢失,因此可应用于一些需要永久保存数据的场合。

ROM 包含一个不可编程的与阵列和一个可编程的或阵列,分别用于地址译码和用作存储阵列。图 1.2 是 ROM 的基本结构框图。图中地址信号 $A_{n-1} \sim A_0$ 是与阵列的 n 个输入变量,经不可编程的与阵列全译码后输出 2^n 条字线 $W_0 \sim W_{2^n-1}$,每条字线对应一个最小项(乘积项)。当地址译码器选中某一字后,可编程的或阵列按编程结果将该字的所有位同时读出。

用户对可编程只读存储器 PROM(Programmable ROM)中的或阵列可以编程,并仅可以编程一次(OTP:One - Time Programmable)。若我们改变某一位线与译码所得字线的连接情况,便可以改变这一位上的输出。从存储器角度看,我们可以说是改变了这一位所存的数据。从组合逻辑电路来看,我们可以说是改变了这一位的逻辑函数。只要对或阵列进行适当的编程,即一次性向 PROM 中写入适当的数据,我们就可在任一输出位上实现输入的任何组合逻辑函数。正是基于这一点,PROM 可作为 PLD 器件使用。PROM 的容量用字数与位个数的乘积来表示。

【例 1.2】 用 PROM 实现 4 位二进制码到格雷码的转换。

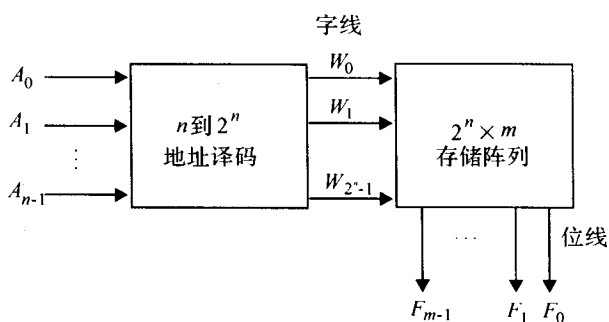


图 1.2 ROM 的一般结构框图

解:①列出 4 位二进制码转换为格雷码的真值表,见表 1.2。我们只需要把 B_3, B_2, B_1, B_0 作为地址信号,把 G_3, G_2, G_1, G_0 作为数据写入到一片容量为 $2^4 \times 4\text{bit}$ 的 PROM 中,即可实现代码的转换。使用时,将待转换的二进制码加在 PROM 地址输入端,在 PROM 的数据输出端就可得到转换后的格雷码。

表 1.2 四位二进制码转换为格雷码的真值表

二进制码				格雷码			
B_3	B_2	B_1	B_0	G_3	G_2	G_1	G_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

② 由真值表写出最小项表达式:

$$G_3 = \sum (8, 9, 10, 11, 12, 13, 14, 15)$$

$$G_2 = \sum (4, 5, 6, 7, 8, 9, 10, 11)$$

$$G_1 = \sum (2, 3, 4, 5, 10, 11, 12, 13)$$

$$G_0 = \sum (1, 2, 5, 6, 9, 10, 13, 14)$$

根据最小项表达式,可画出 PROM 中实现 4 位二进制码转换为格雷码的阵列转换的 PROM 阵列图,如图 1.3 所示。

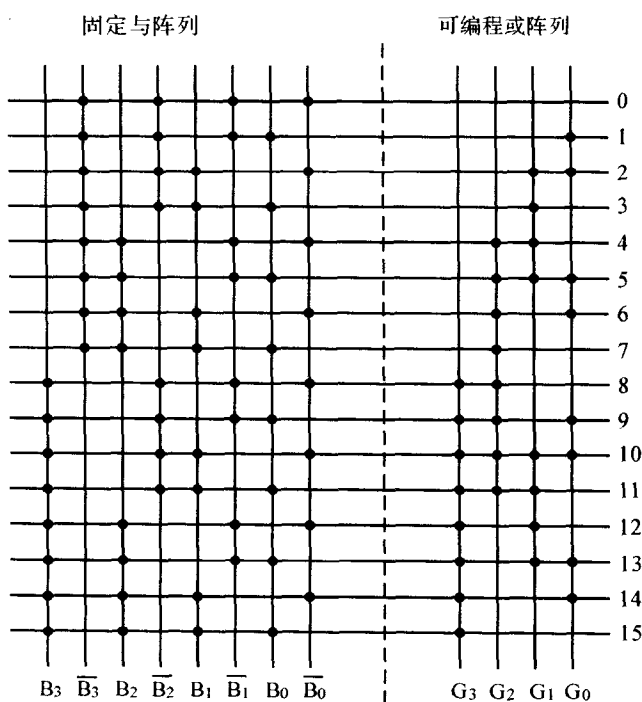


图 1.3 四位二进制码到格雷码

1.1.1.3 PLA

ROM 的输入地址和存储的信息之间有着一一对应的关系。给定一个地址,就能选中并读出一个字。为此,它的与阵列必须对输入地址信号进行全译码产生 2^n 个输出(字线)。从逻辑设计角度来看,不管实际的逻辑函数是否需要,与阵列都产生全部 2^n 个最小项。对于没有使用的最小项,我们可以在芯片中省去。为了提高芯片的利用率,希望与阵列不一定产生 2^n 个最小项,而是产生逻辑函数所需要的乘积项即可。

可编程逻辑阵列 PLA(Programmable Logic Array)器件,就是把 ROM 中的全译码器改为乘积项发生器的一种可编程逻辑器件。图 1.4 是 PLA 的阵列结构图。在 PLA 中,与阵列的内容不