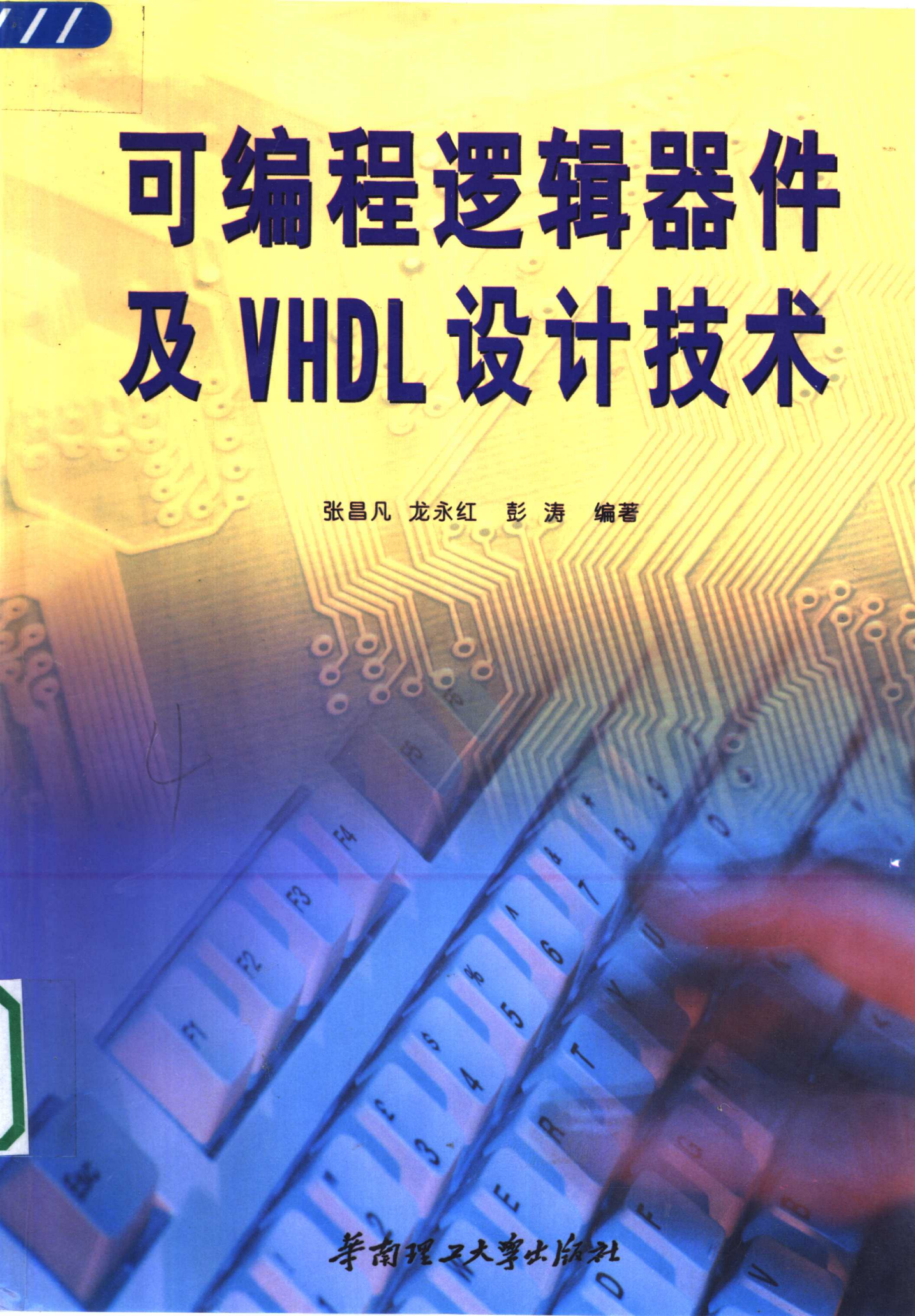




可编程逻辑器件 及 VHDL 设计技术

张昌凡 龙永红 彭涛 编著

华南理工大学出版社

可编程逻辑器件及 VHDL 设计技术

张昌凡 龙永红 彭涛 编著

华南理工大学出版社

·广州·

图书在版编目 (CIP) 数据

可编程逻辑器件及 VHDL 设计技术/张昌凡, 龙永红, 彭涛编著. —广州: 华南理工大学出版社, 2001.7

ISBN 7-5623-1711-9

I. 可… II. ①张…②龙…③彭… III. 可编程序控制器-设计-应用软件, VHDL
IV. TP332.3

中国版本图书馆 CIP 数据核字 (2001) 第 038447 号

总发行: 华南理工大学出版社 (广州五山华南理工大学 17 号楼, 邮编 510640)

发行电话: 020-87113487 87111048 (传真)

E-mail: scut202@scut.edu.cn <http://www2.scut.edu.cn/press>

责任编辑: 乔丽

印刷者: 华南理工大学印刷厂印装

开本: 787×1092 1/16 印张: 10.625 字数: 265 千

版次: 2001 年 7 月第 1 版第 1 次印刷

印数: 1~2000 册

定价: 17.50 元

版权所有 盗版必究

目 录

第一章 绪 论	(1)
第一节 引言	(1)
第二节 可编程逻辑器件及 EDA 技术的发展史	(2)
第二章 可编程逻辑器件基础	(7)
第一节 可编程逻辑器件的分类	(7)
一、可编程逻辑器件的集成度分类	(7)
二、可编程逻辑器件的其他分类方法	(10)
第二节 PLD 类器件的基本结构	(10)
一、与或阵列	(11)
二、宏单元	(12)
第三节 FPGA 的基本结构	(14)
一、查找表型 FPGA 的结构	(14)
二、多路开关型 FPGA 的结构	(15)
三、多级与非门型 FPGA 的结构	(16)
第四节 可编程元件	(17)
一、熔丝开关和反熔丝开关	(17)
二、浮栅编程技术元件	(18)
三、SRAM 配置存储器	(19)
第五节 先进的可编程逻辑器件的编程和测试技术	(19)
一、在系统可编程技术	(19)
二、边界扫描测试技术	(21)
第三章 Xilinx 公司的 FPGA	(22)
第一节 可编程逻辑器件	(22)
一、Xilinx 公司简介	(22)
二、正确选择器件	(23)
三、FPGA 简介	(26)
四、EPLD 简介	(31)
五、硬连线门阵列	(33)
六、军品器件	(33)
七、Xilinx 器件说明	(34)
八、后缀 A 的含义	(34)
第二节 XC2000 逻辑单元阵列	(35)
一、概述	(35)
二、XC2000 逻辑单元阵列系列	(36)

第三节 XC3000 逻辑单元阵列	(46)
一、概述	(46)
二、XC3000, XC3000A, XC3000L, XC3100 和 XC3100A 逻辑单元阵列系列	(47)
第四节 XC4000 逻辑单元阵列	(56)
一、概述	(56)
二、XC4000 逻辑单元阵列系列	(57)
第四章 数字硬件系统设计的描述方法	(73)
第一节 传统的硬件系统设计方法	(73)
第二节 VHDL 设计方法学	(74)
第三节 采用 VHDL 设计硬件系统的优点	(75)
第五章 VHDL 设计描述的基本结构	(77)
第一节 VHDL 的术语	(77)
第二节 VHDL 的基本结构	(79)
一、实体 (ENTITY)	(79)
二、结构体 (ARCHITECTURE)	(80)
三、程序包 (PACKAGE)	(81)
四、子程序 (Subprogram)	(81)
五、库 (LIBRARY)	(84)
第三节 VHDL 设计描述的语法结构	(86)
第六章 MAX + PLUS II VHDL 的数据类型和操作运算	(92)
第一节 对象声明 (Object declaration)	(92)
第二节 数据类型 (Data Types)	(93)
第三节 操作运算符	(96)
一、操作运算符的分类	(96)
二、运算符的优先级	(98)
第四节 类型转换	(98)
第七章 MAX + PLUS II VHDL 机构体的主要描述语句	(100)
第一节 并发描述语句	(100)
一、进程 (PROCESS) 语句	(100)
二、简单信号赋值 (Concurrent Signal Assignment) 语句	(101)
三、条件信号赋值 (Conditional Signal Assignment) 语句	(101)
四、选择信号赋值 (Selected Signal Assignment) 语句	(102)
五、元件例化 (Component Instantiation) 语句	(102)
六、生成 (GENERATE) 语句	(104)
七、块 (BLOCK) 语句	(105)
第二节 顺序描述语句	(107)
一、简单信号赋值语句	(108)

二、变量赋值语句	(109)
三、WAIT 语句	(110)
四、IF 语句	(111)
五、CASE 语句	(113)
六、LOOP 语句	(114)
七、NEXT 语句	(115)
八、RETURN 语句	(116)
九、NULL 语句	(116)
十、断言 (ASSERTION) 语句	(116)
十一、过程调用 (Procedure Call) 语句	(117)
第八章 组合逻辑的 MAX+PLUS II VHDL 描述	(118)
第一节 并发的信号赋值语句 (Concurrent Signal Assignment Statement)	(118)
第二节 进程语句	(119)
第九章 时序逻辑的 MAX+PLUS II VHDL 描述	(121)
第一节 寄存器的 VHDL 实现	(121)
第二节 计数器的 VHDL 实现	(124)
第三节 状态机的 VHDL 实现	(132)
第十章 MAX+PLUS II VHDL 的设计实例	(137)
第一节 微波炉定时器芯片的设计	(137)
一、微波炉定时器芯片的功能要求	(137)
二、微波炉定时器芯片的模块划分	(138)
三、微波炉定时器芯片的顶级实体实现	(139)
四、微波炉定时器芯片各功能模块的实现	(141)
第二节 电机转速控制器的设计	(150)
一、电路系统工作原理	(150)
二、VHDL 描述方法及有关语法分析	(151)
三、测试台	(154)
第三节 神经元计算机的设计	(155)
一、神经元计算机简介	(155)
二、NPS 的行为级描述	(156)
三、NPS 的测试	(158)
第十一章 MAX+PLUS II VHDL 工具的使用	(159)
参考文献	(163)

第一章 绪 论

第一节 引 言

人类已进入数字化的时代。由于数字技术在处理与传输信息方面具有许多优点，因此其使用已渗透到人类生活的各个领域。

数字系统像其他电子系统一样，往往采用积木式的方法进行设计。由器件搭成电路板，由电路板搭成电子系统。数字系统最初的“积木块”是具有固定功能的标准集成电路，如 74/54 系列（TTL）、4000 和 4500 系列（CMOS）芯片以及一些具有固定功能的大规模集成电路。用户只能根据需求选择合适的器件，并按照器件推荐的电路搭成系统。在设计时，设计者没有灵活性可言，搭成的系统需要的芯片种类多且数目大。

随着半导体技术和 EDA（Electronic Design Automation）技术的发展，传统的系统设计方法有了飞跃性的进步。在半导体技术方面，可编程技术被广泛应用于器件设计，给数字系统的设计带来很大的灵活性。现在构成数字系统仅需要三类“积木块”，即微处理器、存储器和可编程逻辑器件。微处理器和存储器从问世以来就是一种可编程的器件。在电子系统中，以前尽管采用了微处理器和存储器，但是仍然需要大量的中、小规模数字集成电路来进行各种逻辑控制，直至最近 10 年，随着可编程逻辑器件的出现，才给逻辑控制提供了可编程的现实性和灵活性。由于器件可以进行编程，因此设计硬件便可以像设计软件那样方便。现在，已有越来越多的设计者采用可编程逻辑器件实现逻辑控制，即用所谓“CPU+RAM+CPLD（FPGA）”的模式进行设计。

图 1-1 所示为传统的电子系统设计方法和基于芯片的设计方法。

可编程逻辑器件的出现，改变了传统的数字系统设计方法。传统的数字系统设计方法只能对电路板进行设计，通过设计电路板来实现系统功能。采用可编程器件则可以对器件进行设计，通过设计芯片来实现系统功能的方法称为基于芯片的设计方法。新的设计方法能够由设计者定义器件的内部逻辑和管脚，将原来由电路板设计完成的大部分工作放在芯片设计中进

行。这样不仅可通过芯片设计实现多种数字逻辑系统功能，而且由于管脚定义的灵活性，大大减轻了电路图设计和电路板设计的工作量和难度，从而有效地增强了设计的灵活性，

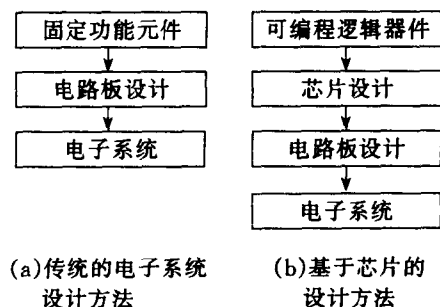


图 1-1 传统的电子系统设计方法和基于芯片的设计方法

提高了工作效率；同时，基于芯片的设计可以减少芯片数量，缩小系统体积，降低能源消耗，提高系统的性能指标和可靠性。

在现代电子系统设计领域,EDA 已经逐渐成为电子系统的重要设计手段。无论是设计逻辑芯片还是数字系统,其设计作业的复杂程度都在不断增加。如今,仅仅依靠手工进行数字系统的设计已经不能满足要求,所有的设计工作都需要在计算机上借助 EDA 工具进行。

EDA 即电子设计自动化,也就是利用计算机完成电子系统设计。EDA 技术以计算机为工具,代替人完成数字系统的逻辑综合、布局布线和设计仿真等工作。设计者只需要完成对系统功能的描述,就可以由计算机软件进行处理,得到设计结果。而修改设计如同修改软件一样方便。利用 EDA 工具进行设计,可以极大地提高设计的效率。

在 EDA 技术方法中,设计仿真是进行设计验证的主要手段。逻辑仿真工具能够验证设计的功能,时延仿真工具可以估计系统时延。高级的仿真工具还可以对系统的时延、驱动能力、电磁兼容特性、机械特性和热特性进行仿真。利用设计仿真工具,设计者可以预知设计结果,减少设计的盲目性。

可编程逻辑器件和 EDA 技术给今天的硬件系统设计者提供了强有力的工具。现在,只要拥有一台计算机、一套相应的 EDA 软件和空白的可编程逻辑器件,在实验室就可以完成数字系统的设计与生产。可以说,当今的数字系统设计离不开可编程逻辑器件和 EDA 设计工具。

另外,高速发展的 CPLD/FPGA 器件又为 EDA 技术的不断进步奠定了坚实的物质基础。CPLD/FPGA 器件更广泛的应用及厂商间的竞争,使得普通的技术人员获得廉价的器件和 EDA 软件成为可能。

现在的 EDA 工具软件已突破早期仅能进行 PCB 版图设计的局限,配备了系统设计自动化的全部工具(如各种常用的硬件描述语言平台 VHDL, Verilog HDL 和 ABEL-HDL 等)和多种逻辑描述输入工具(如硬件描述语言文本输入法以及原理图输入法、波形输入法等),同时还配备了高性能的逻辑综合、优化和仿真模拟工具。

所有的一切都为今天的电子设计工程技术人员提供了强有力的工具。在过去令人难以置信的事,今天已变得平常,只需拥有一台计算机、一套 EDA 软件和一片 CPLD 或 FPGA 芯片,您就能在家中完成大规模集成电路和数字系统的设计。

第二节 可编程逻辑器件及 EDA 技术的发展史

伴随着半导体技术的飞速发展,数字电子技术的应用经历了从分立元件、小规模集成电路(SSI),到中规模集成电路(MSI)、大规模集成电路(LSI)乃至超大规模集成电路(VLSI)的过程。数字系统应用的基本特征由中小规模集成度的通用集成电路,向用户定制的专用集成电路(ASIC)过渡。20 世纪 80 年代出现的可编程逻辑器件(PLD—Programmable Logic Devices),在一定程度上使数字系统设计工程师进行快捷、灵活的设计成为可能。PLD 器件的应用,使一系列功能强、速度高、灵活性大的积木式系统设计得以成功。但是,随着现代数字系统设计的发展,PLD 器件无论是在集成容量、功耗、速度,还是在逻辑设计的灵活性上,均不能满足要求。FPGA 器件的产生正是由此而来,它作为

一种新型的现场可编程的专用集成块，显示出诱人的应用前景。

历史上，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、速度和灵活性方面都有很大的改进和提高。

1. 可编程逻辑器的演变过程

可编程逻辑器件大致的演变过程如下：

- (1) 20 世纪 70 年代，熔丝编程的 PROM 和 PLA 器件成为最早的可编程逻辑器件；
- (2) 20 世纪 70 年代末，AMD 公司开始推出 PAL 器件；
- (3) 20 世纪 80 年代初，Lattice 公司发明可电擦写的、比 PAL 使用灵活性更高的 GAL 器件；
- (4) 20 世纪 80 年代中期，Xilinx 公司提出现场可编程概念，同时生产了世界上第一片 FPGA 器件。同一时期，Altera 公司推出 EPLD 器件，该器件较 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 公司提出在系统可编程技术，并推出一系列具备在系统可编程能力的 CPLD 器件。

进入 20 世纪 90 年代后，可编程逻辑集成电路技术进入飞速发展时期。在生产工艺方面，可编程逻辑集成电路的线宽从 1990 年的 $1\ \mu\text{m}$ 发展到 1997 年的 $0.35\ \mu\text{m}$ ，甚至已经达到深亚微米级，在一块硅片上最多可以集成 20 万个逻辑门。与此同时，器件速度指标也在飞速提高，到 1996 年为止，FPGA 的门延时已小于 $3\ \text{ns}$ ，CPLD 器件的系统工作速度已达 $180\ \text{MHz}$ （Lattice 的 ispLSI2032 芯片），如图 1-2 所示。CMOS 工艺在速度上超过双极型工艺（Bipolar），成为可编程逻辑集成的主要工艺手段。在便携式计算机和移动电话等产品上使用的（低电压） $3.3\ \text{V}$ 器件的市场份额越来越大。在器件的编程和测试技术方面，除传统的熔丝（Fuse）和紫外线工艺（UVEPROM）外，出现了性能更佳的反熔丝（Antifuse）和电擦写（EEPROM）工艺，EEPROM 有完全取代 UVEPROM 的趋势。FPGA 器件的现场可编程技术和 CPLD 器件的在系统可编程技术使得编程器件在使用上更为方便，而边界扫描测试技术又大大加强了芯片、电路板和系统的可测试性。

2. 可编程逻辑器件的发展

可编程逻辑器件技术未来发展的动力仍来自于实际应用的要求和芯片制造商之间的竞争。可以断定，可编程逻辑器件在结构、密度、功能、速度和灵活性方面将得到进一步发展，各制造商受商业利益驱使会继续推出一些新的技术来提高器件的性能，以增强竞争力。在最近几年内，可编程逻辑器件技术的发展热点可能体现在以下两个方面。

(1) 从互连延时入手解决系统速度

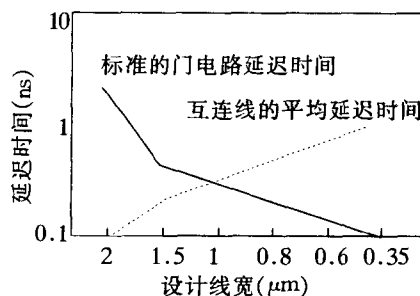


图 1-2 生产工艺、集成密度和器件门延时的关系

注：设计线宽为 2, 1.5, 1, 0.8, 0.6 和 $0.35\ \mu\text{m}$ 时，相应的门电路规模为 2, 5, 15, 25, 40 和 80 万门。

问题。过去,由于门延时较大,所以系统的速度主要取决于门延时。然而随着技术的发展,门延时已从过去的几百 ns 降低到不足 2ns。由于硅片几何尺寸的限制,继续减少门延时变得越来越困难。与此同时,和门延时相比,互连延时却越来越大,成了系统延迟时间的关键因素。

(2) 可再配置技术。在系统可编程技术和现场可编程技术都能够对器件功能进行重新配置,故可称之为初级的可再配置技术。随着技术的发展,将出现能够在极短时间内重新配置的器件。这将使得硬件的概念发生革命性改变,即系统硬件的功能不再固定,而是像软件一样可以灵活地加载配置。可再配置技术是继续提高计算机运算速度最有潜力的技术之一,对于复杂的数学运算,例如 DSP (数字信号处理) 等,硬件的处理速度大大高于软件,用可再配置的硬件代替软件能够显著地提高计算机的运算速度。

3. EDA 技术的发展史

历史上,EDA 经历了几次大的进步,分述如下。

(1) 20 世纪 70 年代,EDA 工具的供应商只有几家,产品几乎全部面向 LSI 或 PCB 布线设计。例如,现在常用的 TANGO 软件就属于 PCB 布线软件。这个时期的 EDA 一般称为电路 CAD (计算机辅助设计)。

(2) 20 世纪 80 年代,情况发生了变化。当时 Mentor 公司、Daisy Systems 公司及 Logic System 公司进入市场,开始供应带电路图编辑工具和逻辑模拟工具的 EDA 软件。这个时期的 EDA 工具是以数字电路分析工具为代表,主要解决电路设计没有完成之前的功能检验问题。

(3) 20 世纪 90 年代起直至目前使用的 EDA 软件属第三代 EDA 软件。它以逻辑综合、硬件行为仿真、参数分析和测试为重点,设计工具门类齐全,能够提供系统设计所需的全部工具,如描述设计意图的设计输入工具、具有逻辑综合和设计优化能力的设计工具,以及验证设计和评估性能的仿真工具,能够在系统级、电路级和 RTL 及门级进行设计描述、综合和仿真。图 1-3 反映了 EDA 技术的发展趋势。

4. EDA 技术的发展方向

未来的 EDA 技术将向广度和深度两个方向发展。

(1) 在广度上,EDA 会日益普及,成为每一个电子工程师的主要设计工具。过去,由于 EDA 软件价格高昂,对硬件环境要求高,其运行平台是工作站和 UNIX 操作系统,只有少数实验室和学校有条件使用 EDA 工具。近几年来,EDA 软件的 PC 平台化进展迅速,普通的设计师也可以得到廉价的、在 Windows 或 Windows NT 上运行的 EDA 软件,这些 PC 平台上的 EDA 软件具有整套的逻辑设计、仿真和综合工具。随着 PC 机性能的提高,PC 平台上的软件功能将会更加完善。EDA 软件的 PC 平台化是 EDA 普及的前提,也是 EDA

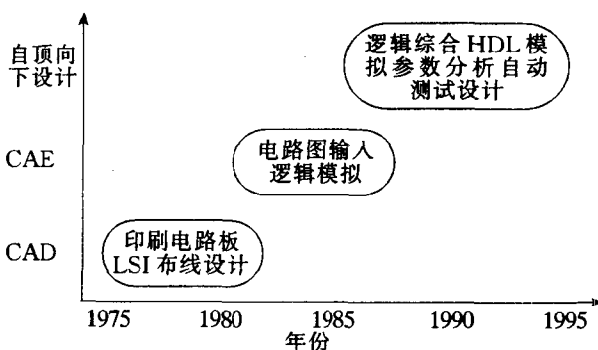


图 1-3 EDA 技术的发展趋势

技术发展的必然趋势。

(2) 在深度上, EDA 技术发展的下一步是 ESDA (Electronic System Design Automation, 电子系统设计自动化) 和 CE (Concurrent Engineering, 并行设计工程)。目前的各种 EDA 工具, 如系统仿真、PCB 布线、逻辑综合、DSP 设计工具等彼此是独立的, 技术的发展及各厂商在缩短电子系统设计周期方面的竞争, 要求所有的系统工具在统一的数据库及管理框架环境下工作, 由此提出了 ESDA 和 CE 的概念。

ESDA 和现在的 EDA 不同, 它强调建立从系统到电路的统一描述语言, 同时考虑仿真、综合与测试; 它将定时、驱动能力、电磁兼容性、机械和散热等约束条件都加到设计综合中, 统一进行设计描述和优化, 提高了设计的一次成功率。ESDA 要求系统设计者改变优先考虑具体的物理实现方式的传统思路, 用抽象的概念确定系统的设计目标, 将精力集中于系统的设计构思、综合比较方案和寻找优化的设计, 以便实现用户对复杂产品的功能和性能要求。简单地说, 就是设计者应优先考虑设计什么样的产品, 而不是一开始就想该怎么设计。

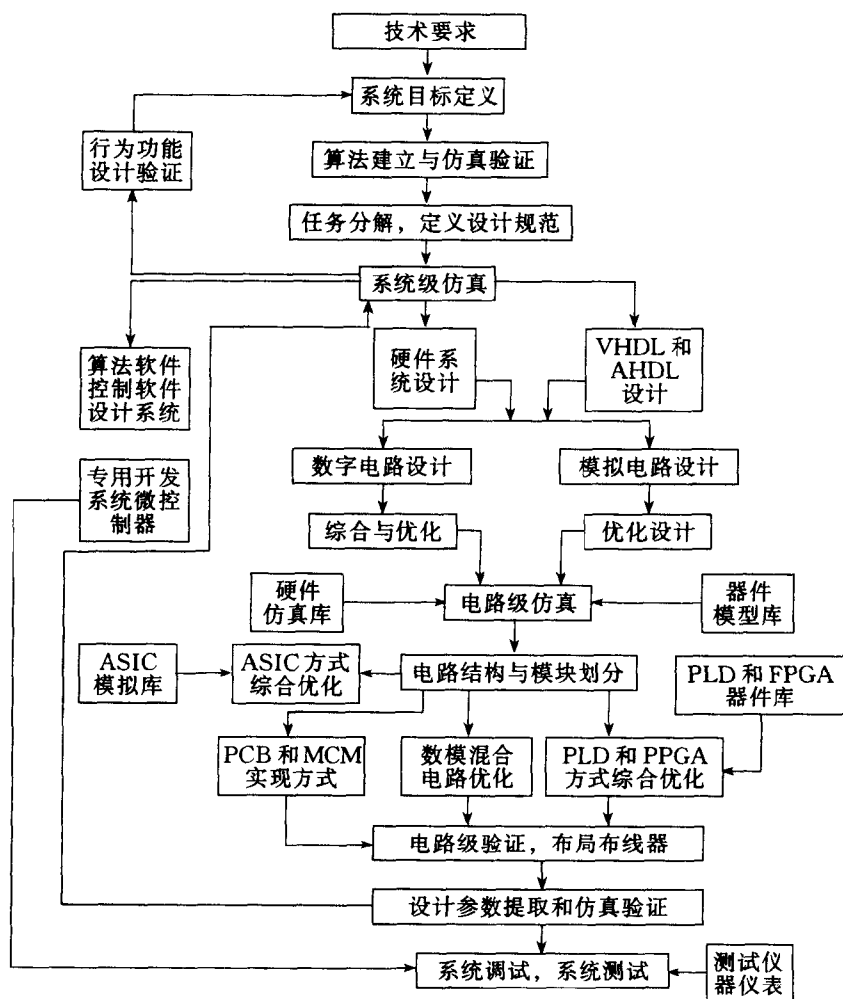


图 1-4 ESDA 软件集成系统的构成和设计、仿真过程

CE 要求从管理层次上把工艺、工具、任务、智力和时间安排协调好；它使用统一的集成化设计环境，由若干个相关的设计小组共享数据库和知识库，同步地进行设计。CE 改变了在传统的电子集成系统设计研制过程中，过分依赖专业分工和设计人员、过分强调所学专业知识的状况。

图 1-4 给出了一个 ESDA 软件集成系统的构成和设计、仿真过程。

本书将分别介绍 Xilinx 公司的 FPGA 现场可编程逻辑器件、EDA 设计工具和 VHDL 设计语言。

第二章 可编程逻辑器件基础

大规模可编程器件是数字系统设计的基础之一。本章介绍目前常用的几类具有代表性的可编程逻辑器件的结构和原理，以及相关的工艺、编程和测试技术。可编程逻辑器件的发展变化迅速，在学习这些内容时，应注意掌握基本的概念、结构和分析方法。

第一节 可编程逻辑器件的分类

常见的可编程逻辑器件有 FPGA, CPLD, EPLD, GAL, PAL, PLA 和 PROM 等。由于历史的原因，对可编程逻辑器件的命名不很规范，一种器件往往具备几种器件的特征，不能够严格地进行分类，所以可编程逻辑器件有多种分类方法，没有统一的分类标准。本节介绍其中几种比较通行的分类方法。

一、可编程逻辑器件的集成度分类

集成度是集成电路一项很重要的指标，如果按集成密度来分类，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)。GAL22V10 是简单 PLD 和复杂 PLD 的分水岭。一般也按照 GAL22V10 芯片的容量区分为 LDPLD 和 HDPLD。GAL22V10 的集成密度根据制造商的不同，大致在 500~750 门之间。按照这个标准，PROM, PLA, PAL 和 GAL 器件属于低密度可编程逻辑器件 (LDPLD)，而 EPLD, CPLD 和 FPGA 属于高密度可编程逻辑器件 (HDPLD)。如图 2-1 所示。

1. 低密度可编程逻辑器件 (LDPLD)

低密度可编程逻辑器件包括 PROM, PLA, PAL 和 GAL 四种。

PROM 即可编程只读存储器。它的基本结构是：与阵列固定和或阵列可编程的与或阵列。PROM 采用熔丝工艺编程，只能写一次，不可以擦除或重写。随着技术的发展和应用方面的要求，后来又出现了 EPROM（紫外线擦除可编程只读存储器）和 EEPROM（电擦写可编程只读存储器）。由于 PROM 具有价格低、易于编程的特点，适合于

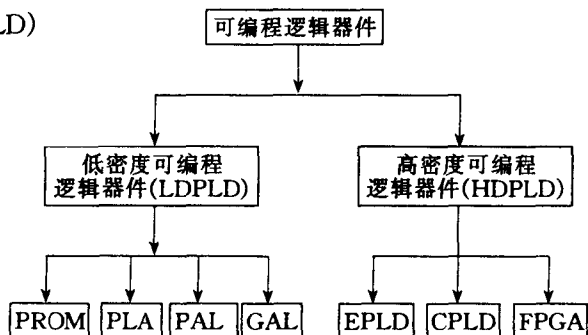


图 2-1 按集成密度分类的可编程逻辑器件

存储函数和数据表格，在某些场合有一定的用途。

PLA (Programmable Logic Array) 也是基于与或阵列的器件。它的与阵列和或阵列都是可编程的。PLA 曾经被认为是极有发展前途的可编程逻辑器件，但由于器件的资源利用率低，现在已经不常使用，只在一些传统的场合还有应用。

PAL (Programmable Array Logic) 由 AMD 公司发明，也是与或阵列结构的器件。在结构上，这类器件包括一个可编程的与阵列和一个固定的或阵列，其中与阵列的编程特性使输入项可以增多，而或阵列固定使器件得以简化。PAL 具有多种输出结构形式。因而其型号最多。

GAL (Generic Array Logic) 器件是 Lattice 公司于 20 世纪 80 年代发明的一种可电擦写、可重复编程、可设置加密位的 PLD 器件。GAL 器件和 PAL 器件相比，增加了一个可编程的逻辑宏单元 (OMLC) 输出，通过对 OLMC 进行配置可以得到多种形式的输出和反馈。具有代表性的 GAL，芯片有 GAL16V8 (如图 2-2 所示) 和 GAL20V8。这两种 GAL 器件几乎能够仿真所有类型的 PAL 器件。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，所以 GAL 几乎完全代替了 PAL 器件。

低密度可编程逻辑器件易于编程，对开发软件的要求低，故在 20 世纪 80 年代得到了广泛应用。但随着技术的发展，低密度可编程逻辑器件在集成密度和性能方面的局限性也暴露了出来。低密度可编程逻辑器件的寄存器、I/O 引脚和时钟资源的数目有限，没有内部互连，使设计的灵活性受到明显的限制。

2. 高密度可编程逻辑器件 (HDPLD)

高密度可编程逻辑器件包括

EPLD, CPLD 和 FPGA 三种。

20 世纪 80 年代中期，Altera 公司推出一种新型的、可擦除的可编程逻辑器件，称为 EPLD (Erasable Programmable Logic Device)，它是一种基于 EPROM 和 CMOS 技术的可编程逻辑器件。EPLD 器件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。宏单元和整个器件的逻辑功能均由 EPROM 来定义和规划。从某种意义上讲，EPLD 是改进了的 GAL。EPLD 的特点是大量增加了输出宏单元的数目，提供了更大的与阵列。由于特有的宏单元结构，使设计的灵活性较 GAL 有较大改善，集成密度提高，在一片芯片内能够实现较多的逻辑功能。EPLD 由于保留了逻辑块的结构，内部连线相对固定，即使是大规模集成容量的器件，其内部延时也很小，有利于器件在高频下工作。

世界著名的半导体器件公司如 Altera, Xilinx, AMD, Lattice 和 Atmel 均生产 EPLD

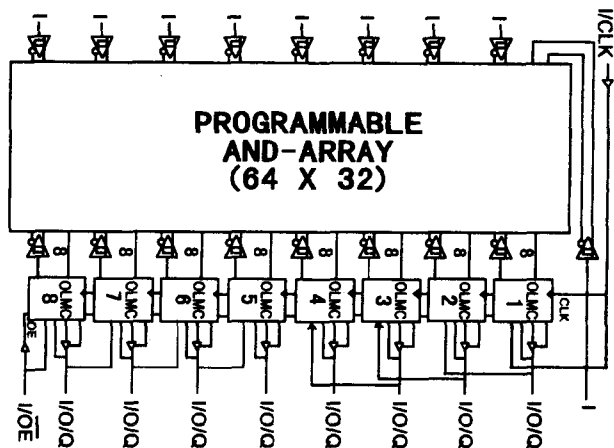


图 2-2 GAL16V8 原理图

产品，但各公司产品的结构差异较大。

由于 EPLD 内部互连能力十分弱，故在 20 世纪 80 年代末受到另一种新兴的可编程逻辑器件 FPGA 的冲击，直到 20 世纪 90 年代 EPLD 的改进器件 CPLD (Complex PLD) 器件出现后，这种情况才有所改观。

CPLD 器件为复杂可编程逻辑器件。同 EPLD 相比，CPLD 增加了内部连线，对逻辑宏单元和 I/O 单元也有重大改进。一般情况下，CPLD 器件中至少包含三种结构：可编程逻辑宏单元，可编程 I/O 单元和可编程内部连线。部分 CPLD 器件内部还集成了 RAM，FIFO 或双口 RAM 等存储器。以适应 DSP 应用设计的要求。

典型的 CPLD 器件有 Lattice 的 pLSI/ispLSI 系列器件、Xilinx 的 7000 和 9500 系列器件、Altera 的 MAX9000 系列器件和 AMD 的 MACH 系列器件。

图 2-3 是 Lattice 公司生产的 ispLSI1016 芯片的结构框图。ispLSI 器件还有一个特性，即具备在系统编程能力 (ISP, In System Programmable)。所谓“在系统编程”，简单讲就是对焊到电路板上的器件进行编程和再配置。具有 ISP 能力的器件 (即 ISP 器件) 在使用上同 FPGA 一样方便。

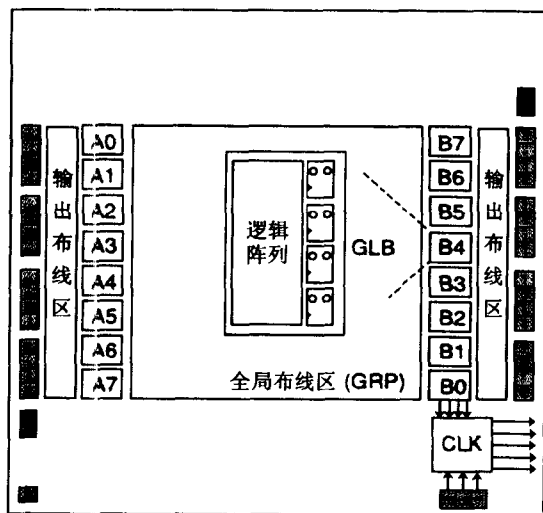


图 2-3 ispLSI1016 芯片的结构图

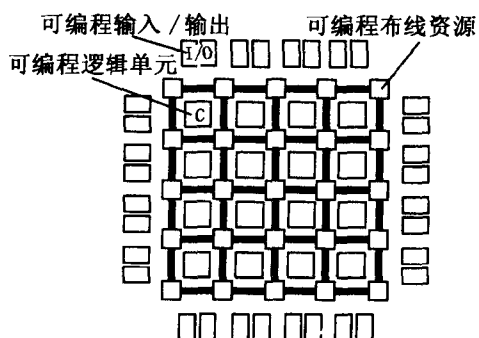


图 2-4 FPGA 的基本结构

CPLD 器件由于具有同 FPGA 器件相似的集成度和易用性，且在速度上还有一定的优势，故能够在可编程逻辑器件技术的竞争中与 FPGA 并驾齐驱，成为两支领导可编程器件技术发展的力量之一。

FPGA (Field Programmable Gate Array) 器件为现场可编程门阵列器件，是最近 10 年发展起来的新型可编程逻辑器件。在结构上，FPGA 器件由逻辑功能块排列为阵列，并由可编程的内部连线连接这些功能块来实现一定的逻辑功能。

以 Xilinx 的 FPGA 器件为例，它的结构可以分为三个部分：可编程逻辑块 CLB (Configurable Logic Blocks)、可编程 I/O 模块 IOB (Input/Output Block) 和可编程内部连接 PI (Programmable Interconnect)，如图 2-4 所示。CLB 在器件中排列为阵列，周围环境

内部连线, IOB 分布在四周的管脚上。Xilinx 的 CLB 功能很强, 不仅能够实现逻辑函数, 还可以配置成 RAM 等复杂的形式。

FPGA 器件的功能由逻辑结构的配置数据决定。工作时, 这些配置数据存放在片内的 SRAM 或者熔丝图上。使用 SRAM 的 FPGA 器件, 在工作前需要从芯片外部加载配置数据。配置数据可以存储在片外的 EPROM 或其他存储体上, 人们可以控制加载过程, 在现场修改器件的逻辑功能, 即所谓现场编程。

FPGA 的发展十分迅速, 目前已达到 25 万门/片的集成度和 3 ns 的内部门延时。除 Xilinx 公司外, Altera 和 Actel 等公司也提供高性能的 FPGA 芯片。

二、可编程逻辑器件的其他分类方法

目前常用的可编程逻辑器件都是从与或阵列和门阵列两类基本结构发展起来的, 所以又可从结构上将其分为两大类器件:

- (1) PLD (Programmable Logic Device) 器件;
- (2) FPGA (Field Programmable Array) 器件。

这种分类方法将基本结构为与或阵列的器件, 称作 PLD; 将基本结构为门阵列的器件, 称作 FPGA。

PLD 器件是最早的可编程逻辑器件。它的基本逻辑结构由与阵列和或阵列组成, 称为与或阵列。它能有效地实现“积之和”形式的布尔逻辑函数。PLD 器件包括简单 PLD 器件 (PROM, PLA, PAL 和 GAL)、EPLD 和 CPLD。PLD 通过修改内部电路的逻辑功能来编程, FPGA 主要通过改变内部连线的布线来编程。

也有人将可编程逻辑器件分为简单 PLD 和复杂 PLD (CPLD), 将 FPGA 也划分到 CPLD 的范围之内。

所有的 CPLD 器件和 FPGA 器件均采用 CMOS 技术, 但它们在编程工艺上有很大的差别。如果按编程工艺划分, 可编程逻辑器件又可分为四种。

(1) 熔丝 (Fuse) 或反熔丝 (Antifuse) 编程器件。PROM 器件、Xilinx 的 XC5000 系列器件和 Actel 的 FPGA 器件等采用这种编程工艺。

(2) UVEPROM 编程器件, 即紫外线擦除/电气编程器件。大多数的 EPROM 和 CPLD 采用这种方式编程。

(3) EEROM 编程器件, 即电擦写编程器件。GAL、ispLSI 器件属于这类器件。

(4) SRAM 器件。Xilinx 的 FPGA 是这一类器件的代表。

在可编程逻辑器件的术语中, 将第 1 类、第 2 类和第 3 类器件称为非易失性器件, 它们在编程后, 配置数据保持在器件上; 第 4 类器件称易失性器件, 这类器件每次掉电后配置数据会丢失, 在每次上电时需要进行重新配置。由于熔丝或反熔丝器件只能写一次, 所以称为 OTP 器件 (一次性编程器件)。其他种类的器件均可以多次编程。

第二节 PLD 类器件的基本结构

PLD 类器件包括 PROM, PLA, PAL, GAL, EPLD 和 CPLD 器件。由于 PLD 器件

种类繁多,不同厂商生产的 PLD 器件的结构差别较大,不能够逐一介绍,本节中选择 PLD 器件中一些具有代表性的结构说明其实现逻辑函数的原理和分析方法。

一、与或阵列

所有 PLD 器件的共同之处是包含一个由与门和或门构成的“与或阵列”。与或阵列在硅片上非常容易实现,对逻辑设计也非常方便,它可以直接实现与或表达形式的逻辑函数。

与或表达式是布尔代数的常用表达形式。根据布尔代数原理,所有的逻辑函数均可以用与或表达式描述。在数字电路课程中,大家已经学过如何利用卡诺图、摩根定理和 Q-M 表,将真值表或其他形式的逻辑函数转换成与或表达式。

可以通过改变与或阵列的连接实现不同的逻辑功能。无论改变与阵列的连接,还是改变或阵列的连接,都可以使所实现的逻辑函数发生变化。根据这一区别,可以将与或阵列划分为如下三种形式。

(1) 与阵列固定、或阵列可编程的与或阵列。PROM 器件采用这种形式。

(2) 与阵列可编程、或阵列固定的与或阵列。PAL, GAL, EPLD 和 CPLD 器件采用这种形式。

(3) 与阵列和或阵列均可编程的与或阵列。PLA 器件采用这种形式。

随着 PLD 器件研究的深入。第 1 种和第 3 种形式的与或阵列结构暴露出了一定的缺陷。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 的级数倍增;第 3 种结构的器件制造工艺复杂,器件工作速度慢。基于这两类与或阵列结构的器件处于被淘汰的边缘,目前只有很少的制造商在继续生产。相对而言,第 2 种形式具备一定的技术优势,是目前 PLD 发展的主流。

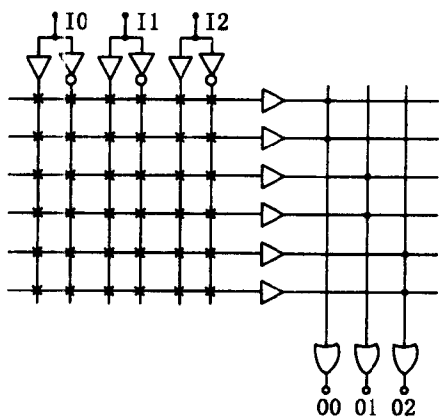


图 2-5 与阵列可编程或阵列固定的与或阵列结构

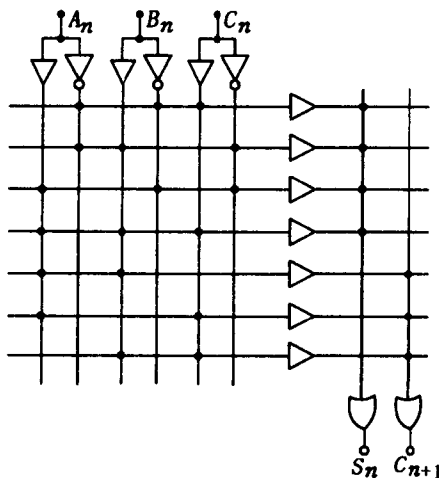


图 2-6 与或阵列实现的全加器

与阵列可编程/或阵列固定的与或阵列结构如图 2-5 所示。图中左边部分为与阵列,右边部分为或阵列,与门采用线与与门形式,交叉点上的“*”表示可编程连接,实点表示固定连接。输入信号通过互补缓冲器输入,提供输入信号的原变量和反变量,对应的输入