

全国高技术重点图书

可编程逻辑器件设计

黄均鼎 俞承芳
蒋慧文 卢 烜 编著

复旦大学出版社

● C A D 应用系列丛书



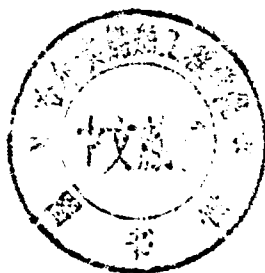
TP332.1
H87
全国高技术重点图书

493558

CAD应用系列丛书

可 编 程 逻 辑 器 件 设 计

黄均鼎 俞承芳 编著
蒋慧文 卢 烜



复旦大学出版社



责任编辑 林溪波

责任校对 韩向群

可编程逻辑器件设计

黄均翔、俞承芳、蒋慧文、卢烜

出 版 复旦大学出版社

(上海国权路 579 号 邮政编码 200433)

发 行 新华书店上海发行所

照 排 南京理工大学激光照排公司

印 刷 江苏句容市排印厂

开 本 787×1092 1/16

印 张 22.75

字 数 560 000

版 次 1997 年 3 月第 1 版 1997 年 3 月第 1 次印刷

印 数 1—1500

书 号 ISBN 7-309-01802-8/T·172

定 价 28.00 元

本版图书如有印订质量问题, 请向承印厂调换。

内 容 简 介

JS160/24

近 10 年来,可编程逻辑器件(包括 PLD 及 FPGA)的发展速度远比其他大规模集成电路为快,它已广泛应用于科研、新产品开发及工业生产中,并日益显示出它的优越性和强大生命力:它具有使用方便、新产品开发周期短、修改逻辑功能简便、成本较低以及允许在已投入运行的系统中修改逻辑功能等优点,特别适合于新产品研制及小批量生产。

本书系统地介绍了可编程逻辑器件的工作原理,着重介绍采用计算机辅助设计(CAD)技术使用上述器件设计数字电路(系统)的方法及技巧,阐述在完成设计后必须进行的功能模拟及同步模拟,简要介绍逻辑综合及划分的原理,对国际上主要生产厂商(Xilinx, Lattice 和 Altera 公司)的 CAD 软件的使用方法和操作步骤作了详细介绍。同时,书中列举很多实例,把设计原理和具体实践有机地结合起来,便于读者学习。

本书可以作为大专院校无线电、电子学、计算机、信息管理、通信、图象处理、自动控制等专业的教材,也可供有关专业的工程技术人员参考。

序

可编程逻辑器件(简称 PLD)及现场可编程门阵列(简称 FPGA)是近年来发展异常迅速的器件,最近 10 年中,每一芯片的门数已从几百门上升到十万门,从而构成了对 ASIC 及传统门阵列电路的很大挑战。

早期的可编程逻辑器件出现于 70 年代,第一个具有典型意义的可编程逻辑器件是 PAL。1984 年,Altera 公司采用 CMOS 工艺研制了首块可擦写的 PLD(简称 EPLD),一反常规的双极型及熔丝工艺制成的器件(这类器件只能编程一次),它可用紫外光擦除并反复编程,使出厂产品可百分之百测试,提高了编程合格率,同时使设计者易于改进设计,在原型机开发中使用极为方便。1985 年 Lattice 公司用 E²CMOS 工艺研制出电可擦除的可编程逻辑器件 GAL(通用阵列逻辑),它具有设计灵活、高速、低功耗、改写迅速方便的特点,现已成为最常用的工业标准器件之一。对于可编程器件的另一分支,Xilinx 公司于 1985 年推出了首块现场可编程门阵列,其后开发了多个系列的 FPGA,1995 年初推出的 XC4025,已达 25000 门。针对逻辑系统往往需要大量内部存储器,据 1995 年报道,Altera 公司已推出 FLEX10K 系列嵌入式 RAM 的 FPGA,这是一种 10 万门级的器件,在芯片内集成了很大容量的内部 SRAM,可用于存储器密集型应用的需要;Xilinx 公司也推出 XC6200 系列的类似规模的 FPGA,除芯片内部有大容量存储器外,其主要特点还在于可以按照数据总线的速度传输配置数据,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迅速作出响应的系统;1996 年初 Lattice 公司向市场推出了称为“子块化”(Cell-based)的可编程器件 ispLSI6192,在单一芯片中同时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,该公司于 1991 年研制出了大规模可编程逻辑器件 ispLSI,不仅可反复编程,而且首次使 PLD 彻底摆脱了编程器,用户可以在系统对安装在印刷电路板上的 LSI 器件进行编程和再编程。其后,Xilinx,Altera 也相继推出了自身的 ISP 器件,如 XC9500,MAX9000 系列即是。

可编程器件在逻辑系统领域的广泛应用,已引起模拟领域技术界的广泛注意,美国的 INTERNATIONAL MICROELECTRONIC PRODUCTS 公司已开发了电可编程模拟电路(EPAC),作为取代试验电路板的方案推向模拟系统设计者,尽管刚刚起步,但发展潜力还是相当广阔。

PLD 及 FPGA 的开发商,都针对本公司的器件开发了整套软件,提供用户进行设计、模拟、并把设计结果配置到相应的器件中去,提供用户的输入手段,包括语言(如 ABEL, HDL 等)、布尔表达式、真值表等,还有易为用户接受的输入简图的方式,尽管有些公司未开发简图输入的工具,但与国际上常用的工具,如 Workview, Or CAD, Candence 等有接口,仍可采用输入简图的方式进行设计。Altera 公司自行开发了简图输入工具,使它的开发系统自成

体系,独具一格。

在大学的电子及电机类专业的本科生中,不懂得可编程器件,不会使用这类器件将是一件憾事,我们希望尽早把可编程器件介绍给学生,不仅给学生有听课的机会,而且还有设计和实践的机会,这是至关重要的。我们希望本书的出版有助于学生学习和使用可编程逻辑器件,有助于这些器件的应用在国内推广。

唐璞山

1996. 4

前 言

可编程逻辑器件(PLD)及现场可编程门阵列(FPGA)已在通信、计算机、图象处理等多个领域得到广泛应用,近年来已成为集成电路发展的最快品种之一。对于这种颇有前途而国内在应用方面还处于发展阶段的器件,应尽快引入大学电子类专业的课程中去,再配以必要的实验,使学生能掌握这种器件的工作原理、设计方法、编程技术及实验技巧,基于这一想法,我们编写了这本书。

国际上生产 PLD 及 FPGA 的公司很多,我们选择了 Xilinx, Lattice 及 Altera 三家公司的产品介绍给读者,这不仅是因为上述厂商是世界上生产 PLD 及 FPGA 的主要公司,在国内也用得较多,还因为作者使用过这些器件,有较丰富的实践经验,因而有利于把使用这些器件的经验介绍给读者。

本书第一、第二章由黄均甬编写,介绍逻辑代数及器件的基本原理;第三、四、五章由俞承芳编写,介绍 Lattice 公司的产品特点及性能、编程语言、编程技术及设计实例;第六、第七章由卢烜编写,介绍 Xilinx 公司的产品特点及性能、编程技术和设计实例;第八、九、十章由蒋慧文编写,介绍 Altera 公司的产品特点及性能、设计系统特点、模拟技术及设计实例。全书由黄均甬统稿。

在本书的编写过程中,科汇(亚太)有限公司、上海莱迪思半导体有限公司及北京集成电路设计中心提供了许多实用的和最新的资料,同时得到了章开和教授、钱学俭先生和姜世平先生的大力帮助和支持;复旦大学电子工程系谢悦华女士协助绘制了书中部分插图,在此一并致谢。

本书在编写出版过程中得到了复旦大学出版社和电子工程系的 CAD 研究室、微机应用实验室及专用集成电路与系统国家重点实验室的大力支持,抽调人员编写,特此表示感谢。还要感谢复旦大学微电子研究所所长唐璞山教授的关怀,拨冗作序推荐本书。

由于时间紧,各公司产品更新快,加之编者水平有限,疏漏或错误之处,敬请读者批评指正。

编 者

1996.4

目 录

序

前言

第一章 数字逻辑设计基础——布尔代数	1
1.1 布尔代数	1
1.2 开关代数	2
参考文献	6
第二章 可编程逻辑器件原理	7
2.1 只读存储器(ROM)介绍	8
2.1.1 双极型 ROM	8
2.1.2 MOS 型 ROM	8
2.2 可编程只读存储器的基本结构	9
2.2.1 二极管型可编程只读存储器	10
2.2.2 双极型晶体管为存储单元的 PROM	12
2.2.3 MOS 型 PROM	14
2.3 现场可编程逻辑阵列	17
2.4 可编程阵列逻辑	18
2.4.1 基本原理	18
2.4.2 PAL 编号	19
2.4.3 PAL 的优点	21
2.5 通用阵列逻辑	22
2.5.1 GAL 的开关参数	22
2.5.2 GAL 16V8/20V8 的结构	24
2.5.3 GAL 16V8/20V8 中的宏单元	24
2.5.4 GAL 16V8/20V8 宏单元的工作方式	27
2.5.5 OLMC 的控制字	29
2.5.6 应用举例	30
2.6 可编程大规模集成及在系统可编程大规模集成的 PLD	31
2.7 现场可编程门阵列	41
2.7.1 器件结构	41
2.7.2 开发系统	47
参考文献	49
第三章 Lattice 通用可编程阵列逻辑器件 GAL	50
3.1 GAL 器件及开发	50

3.1.1	GAL 器件介绍	50
3.1.2	利用 GAL 器件进行逻辑设计	54
3.1.3	GAL 器件的开发	62
3.2	可编程逻辑器件设计语言 ABEL	66
3.2.1	可编程逻辑器件设计语言 ABEL 概述	66
3.2.2	ABEL-HDL 硬件描述语言的基本语法	67
3.2.3	ABEL-HDL 语言源文件的基本结构	70
3.2.4	逻辑描述	76
3.3	ABEL 集成设计环境及处理模块	82
3.3.1	ABEL 集成设计环境的启动	82
3.3.2	ABEL 语言的处理模块	86
3.4	设计实例	86
3.4.1	四位同步计数器的设计	86
3.4.2	计算机接口电路	90
	参考文献	91
第四章	Lattice 可编程大规模集成电路 pLSI 和 ispLSI	92
4.1	pLSI 和 ispLSI 器件及开发	92
4.1.1	pLSI 和 ispLSI 器件介绍	92
4.1.2	利用 pLSI 和 ispLSI 器件进行逻辑设计	97
4.1.3	pLSI 和 ispLSI 器件的开发	99
4.2	pLSI 和 ispLSI 可编程逻辑器件设计软件 pDS	101
4.2.1	pLSI 和 ispLSI 设计软件 pDS 概述	101
4.2.2	pDS 软件语言的基本语法	102
4.2.3	pDS 软件语言源文件的基本结构	104
4.3	pDS 集成设计环境	110
4.3.1	pDS 集成设计环境的启动	111
4.3.2	pDS 设计软件的处理功能	113
4.4	设计实例	117
4.4.1	四位同步计数器的设计	118
4.4.2	计算机接口电路的设计	121
	参考文献	123
第五章	在系统编程技术	124
5.1	在系统编程技术概述	124
5.1.1	可编程逻辑器件的编程技术	124
5.1.2	在系统编程技术	125
5.1.3	在系统可编程逻辑器件	126
5.2	在系统编程原理	130
5.2.1	在系统编程概述	130
5.2.2	ispGDS 的编程原理	133

5.2.3	ispGAL 的编程原理	138
5.2.4	ispLSI 的编程原理	140
5.3	在系统编程的硬件构造	145
5.3.1	并行结构	145
5.3.2	串行结构	147
5.4	在系统编程器件的编程	148
5.4.1	利用 PC 机对在系统编程器件编程	148
5.4.2	利用微处理器对在系统编程器件编程	151
5.4.3	ipCODE 软件	152
	参考文献	154
第六章	Xilinx 现场可编程门阵列	155
6.1	概 述	155
6.1.1	Xilinx 公司可重复编程的 FPGA	155
6.1.2	Xilinx 公司一次性可编程 FPGA	159
6.2	基本结构	161
6.2.1	Xilinx 公司可重复编程 FPGA 的结构	161
6.2.2	Xilinx 公司一次性可编程 FPGA 结构	186
6.3	电路设计	192
6.3.1	组合电路设计	192
6.3.2	时序电路设计	200
6.3.3	利用 IOB 的设计	207
6.3.4	电路设计小结	217
6.4	转换程序及布局布线	218
6.4.1	原理图到 XNF 文件的生成	218
6.4.2	XNF 文件转换成 MAP 文件	219
6.4.3	MAP 文件转换成 LCA 文件	219
6.4.4	使用 LCA 文件进行布局布线	220
6.4.5	由原理图自动生成 LCA 的 Xmake 软件	227
6.5	同步模拟	228
6.6	Xilinx FPGA 器件的电路载入	229
6.6.1	MakeBits 程序	230
6.6.2	MakePROM 程序	232
6.6.3	下载和配置具体使用过程介绍	233
6.7	设计过程实例	240
6.7.1	电路设计	240
6.7.2	电路输入和逻辑模拟	241
6.7.3	电路转换为 LCA 文件	244
6.7.4	布局布线	244
6.7.5	逆转换过程	244

6.7.6 同步模拟	244
6.7.7 电路载入	245
6.7.8 整个电路实现	245
参考文献.....	245
第七章 Xilinx 可擦除式可编程逻辑器件和复杂可编程逻辑器件	246
7.1 Xilinx 可擦除式可编程逻辑器件	246
7.1.1 概 述	246
7.1.2 XC7200A 系列	247
7.1.3 XC7300 系列	252
7.2 Xilinx 复杂可编程逻辑器件	260
7.2.1 概 述	260
7.2.2 结构简介	261
7.2.3 应用特性简介	261
参考文献.....	262
第八章 Altera 公司可编程逻辑器件.....	263
8.1 概 述	263
8.1.1 Altera 公司 PLD 器件简介	263
8.1.2 EPLD 基本结构及时间特性	264
8.2 Altera 系列器件	268
8.2.1 Altera 各种器件描述	268
8.2.2 六种 Altera 系列器件	268
参考文献.....	299
第九章 Altera 软件设计系统.....	300
9.1 概 述	300
9.2 系统设计	300
9.2.1 启动系统	300
9.2.2 逻辑设计输入	302
9.2.3 编 译	316
9.2.4 模 拟	327
9.2.5 编程烧录	332
参考文献.....	333
第十章 Altera 系统编程硬件及系统文件.....	334
10.1 概 述.....	334
10.2 编程硬件.....	334
10.2.1 逻辑编程卡.....	334
10.2.2 主编程器.....	335
10.2.3 适配器.....	335
10.2.4 FLEX8000 系列器件的编程烧录.....	335
10.2.5 在电路配置 ICR	342

10.2.6 在系统编程 ISP	342
10.3 系统文件.....	344
10.3.1 设计文件.....	344
10.3.2 接口文件.....	344
10.3.3 编译文件.....	346
10.3.4 模拟文件.....	347
10.3.5 配置和编程文件.....	348
参考文献.....	349

第一章 数字逻辑设计基础——布尔代数

可编程逻辑器件的设计,离不开逻辑代数的应用,它的数学基础就是布尔代数。布尔代数首先由乔治·布尔(George Boole)提出,它的作用,特别是二元布尔代数——开关代数,随着数字电子计算机的问世而突出出来,它是开关理论和逻辑设计的数学基础。

1.1 布尔代数

布尔代数是一代数($B; \cdot, +, /$ 或 $-; 0, 1$),它由一布尔变量集 B (B 至少包含 $0, 1$ 两个元素)和三个操作(与,用“ $\cdot$ ”来表示;或,用“ $+$ ”来表示;非,用“ $/$ ”或“ $-$ ”来表示)组成,且对 B 中的任何元素 X, Y ,它们的 $X \cdot Y$ (积), $X + Y$ (和)和 $\bar{X}$ (非)都仍在 B 内。

布尔代数的公理有:

① 幂等律: $X \cdot X = X, X + X = X$;

② 交换律: $X \cdot Y = Y \cdot X, X + Y = Y + X$;

③ 结合律: $X \cdot (Y \cdot Z) = (X \cdot Y) \cdot Z, X + (Y + Z) = (X + Y) + Z$;

④ 吸收律: $X \cdot (X + Y) = X, X + (X \cdot Y) = X$;

⑤ 分配律: $X \cdot (Y + Z) = X \cdot Y + X \cdot Z, X + (Y \cdot Z) = (X + Y) \cdot (X + Z)$;

⑥ 对于 0 (也称无效,最小)元素和 1 (也称有效,最大)元素,其公理如下:

对于每个 $X \in B$,存在一个唯一元素(1 元素) $1 \in B$,使

$$X \cdot 1 = 1 \cdot X = X$$

对于每个 $X \in B$,存在一个唯一元素(0 元素) $0 \in B$,使

$$X + 0 = 0 + X = X$$

⑦ 对于每个 $X \in B$,存在一个叫 X 非的唯一元素 $\bar{X} \in B$,使得

$$X \cdot \bar{X} = 0$$

$$X + \bar{X} = 1$$

从上述公理的表述中,读者定会注意到, $X \cdot Y$ 或 $X + Y$ 不是普通的代数运算,元素 0

和 1 也没有普通代数中的 0 和 1 的意义。

我们先考察二元布尔代数 $B_2 = (\{0, 1\}; \cdot, +, /; 0, 1)$ 的三个操作, 如图 1-1 所示。

$\cdot$	0	1
0	0	0
1	0	1

与操作

+	0	1
0	0	1
1	1	1

或操作

/	0	1
0		1
1		0

非操作

图 1-1 二元布尔代数的三种操作

下面再看单元素集 S 构成的幂集 $P(S)$ 。显然, $P(S)$ 只包括 Φ 和 S 两个元素, 与二元布尔代数相类似, 对于 Φ 和 S 也可列出它们的交集、和集及补集三个表, 具体如图 1-2 所示。

$\cap$	Φ	S
Φ	Φ	Φ
S	Φ	S

交集

$\cup$	Φ	S
Φ	Φ	S
S	S	S

和集

/	Φ	S
Φ		S
S		Φ

补集

图 1-2 集的三种操作

上述两个布尔代数同构。

* 如果两个代数存在一一对应和从一个向另一个映射的关系, 这两个代数同构。

1.2 开关代数

开关代数就是二元布尔代数 B_2 , 它是分析和设计数字系统中开关电路的数学基础, 它的两个元素就是由 1 表示的大数和由 0 表示的小数, 是“真”和“假”, 或“是”和“非”在数学上的表述。我们可以用日常生活中的开关接通或断开来说明: 设有两个开关和一个灯泡构成一回路, 两个开关的连接可以是串联, 也可以是并联, 分别如图 1-3 中的 (a) 和 (b) 所示。设用

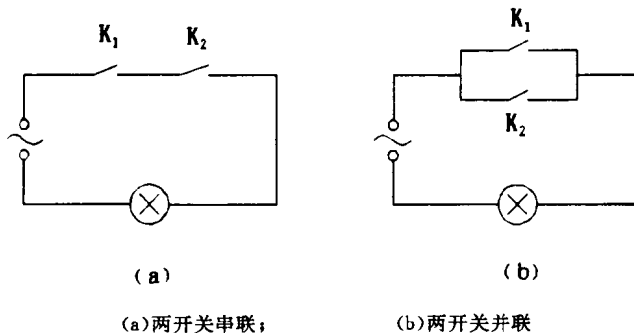


图 1-3 说明开关代数的实例

“1”表示开关接通, “0”表示开关断开; 又用“1”表示灯泡亮, “0”表示灯泡不亮。那末上述开关 K_1 及 K_2 的串、并联关系的真值表分别示于图 1-4 的 (a) 和 (b), 由这两真值表可知, 对 K_1 , K_2 串联的情况, 相当于布尔代数中的与操作, K_1 与 K_2 并联的情况, 相当于布尔代数中的或

操作。

K_1	K_2	L	K_1	K_2	L
0	0	0	0	0	0
0	1	0	0	1	1
1	0	0	1	0	1
1	1	1	1	1	1

(a) K_1 与 K_2 串联 (b) K_1 与 K_2 并联

图 1-4 图 1-3 电路的真值表

根据所含的变量,开关代数的基本性质可以分为三类,如表 1-1 所示。

表 1-1 逻辑方程式

$n=1$	
1.1 $X_1 + X_1 = X_1$	1.1' $X_1 \cdot X_1 = X_1$
1.2 $X_1 + \overline{X_1} = 1$	1.2' $X_1 \cdot \overline{X_1} = 0$
1.3 $1 + X_1 = X_1 + 1 = 1$	1.3' $1 \cdot X_1 = X_1$
1.4 $0 + X_1 = X_1 + 0 = X_1$	1.4' $0 \cdot X_1 = 0$
$n=2$	
2.1 $X_1 + X_2 = X_2 + X_1$	2.1' $X_1 \cdot X_2 = X_2 \cdot X_1$
2.2 $X_1 + X_1 \cdot X_2 = X_1$	2.2' $X_1 \cdot (X_1 + X_2) = X_1$
2.3 $(X_1 + \overline{X_2}) \cdot X_2 = X_1 \cdot X_2$	2.3' $X_1 \cdot \overline{X_2} + X_2 = X_1 + X_2$
$n=3$	
3.1 $X_1 + X_2 + X_3 = (X_1 + X_2) + X_3 = X_1 + (X_2 + X_3)$	3.1' $X_1 \cdot X_2 \cdot X_3 = (X_1 \cdot X_2) \cdot X_3 = X_1 \cdot (X_2 \cdot X_3)$
3.2 $X_1 \cdot X_2 + X_1 \cdot X_3 = X_1 \cdot (X_2 + X_3)$	3.2' $(X_1 + X_2) \cdot (X_1 + X_3) = X_1 + X_2 \cdot X_3$
3.3 $(X_1 + X_2) \cdot (X_2 + X_3) \cdot (X_3 + \overline{X_1}) = (X_1 + X_2)(X_3 + \overline{X_1})$	3.3' $X_1 \cdot X_2 + X_2 \cdot X_3 + X_3 \cdot \overline{X_1} = X_1 \cdot X_2 + X_3 \cdot \overline{X_1}$
3.4 $(X_1 + X_2) \cdot (\overline{X_1} + X_3) = X_1 \cdot X_3 + \overline{X_1} \cdot X_2$	

*n 代表变量数。

下面介绍开关函数的一般定理。

1. 狄·莫根(De Morgan)定理

(a) $\overline{(X_1 + X_2 + \cdots + X_n)} = \overline{X_1} \cdot \overline{X_2} \cdot \cdots \cdot \overline{X_n}$

(b) $\overline{(\overline{X_1} \cdot \overline{X_2} \cdot \cdots \cdot \overline{X_n})} = \overline{X_1} + \overline{X_2} + \cdots + \overline{X_n}$

2. 香农(Shannon)定理

由于狄·莫根定理在说明互补函数之间的关系不完善,香农作了推广。

$\overline{f}(X_1, X_2, \cdots, X_n, +, \cdot) = f(\overline{X_1}, \overline{X_2}, \cdots, \overline{X_n}, \cdot, +)$ 此定理用文字来表达即为:任何函数之非,可以通过函数变量求反,并使与操作代替或操作,或操作代替与操作来求得。

3. 展开定理

(a) $f(X_1, X_2, \cdots, X_n) = X_1 \cdot f(1, X_2, \cdots, X_n) + \overline{X_1} \cdot f(0, X_2, \cdots, X_n)$

(b) $f(X_1, X_2, \cdots, X_n) = [X_1 + f(0, X_2, \cdots, X_n)] \cdot [\overline{X_1} + f(1, X_2, \cdots, X_n)]$

这是函数对变量 X_1 的展开,对其他变量的展开式与此类似。

例 1-1 函数 $f(X_1, X_2, X_3) = \overline{X_1} \cdot \overline{X_2} \cdot \overline{X_3} + X_1 \cdot (\overline{X_2} + X_3)$ (1-1)

按展开定理(a),可写为

$$f(X_1, X_2, X_3) = X_1 \cdot (\overline{X_2} \cdot \overline{X_3} + \overline{X_2} + X_3) + \overline{X_1}$$

或按展开定理(b), 可写为

$$f(X_1, X_2, X_3) = (X_1 + 1) \cdot (\overline{X_1} + \overline{X_2} \cdot \overline{X_3} + \overline{X_2} + X_3)$$

从展开定理又可得以下定理:

$$4. (a) \quad X_1 \cdot f(X_1, X_2, \dots, X_n) = X_1 \cdot f(1, X_2, \dots, X_n)$$

$$\text{或} \quad X_1 + f(X_1, X_2, \dots, X_n) = X_1 + f(0, X_2, \dots, X_n)$$

$$(b) \quad \overline{X_1} \cdot f(X_1, X_2, \dots, X_n) = \overline{X_1} \cdot f(0, X_2, \dots, X_n)$$

$$\text{或} \quad \overline{X_1} + f(X_1, X_2, \dots, X_n) = \overline{X_1} + f(1, X_2, \dots, X_n)$$

应用表 1-1 的开关函数性质及上述定理, 就可把 n 个变量的 2^n 个开关函数化简为积-和范式。当 $n=2$ 时, 所有开关函数的积-和范式列于表 1-2。

表 1-2 二变量开关函数简化公式

函数数	积-和 范式项数	$\overline{X_1}\overline{X_2}$	$\overline{X_1}X_2$	$X_1\overline{X_2}$	X_1X_2	积-和范式	简化函数
1	0	0	0	0	0	0	0
2	1	0	0	0	1	X_1X_2	X_1X_2
3	1	0	0	1	0	$X_1\overline{X_2}$	$X_1\overline{X_2}$
4	1	0	1	0	0	$\overline{X_1}X_2$	$\overline{X_1}X_2$
5	1	1	0	0	0	$\overline{X_1}\overline{X_2}$	$\overline{X_1}\overline{X_2}$
6	2	0	0	1	1	$X_1\overline{X_2} + X_1X_2$	X_1
7	2	1	1	0	0	$\overline{X_1}\overline{X_2} + \overline{X_1}X_2$	$\overline{X_1}$
8	2	0	1	0	1	$\overline{X_1}X_2 + X_1X_2$	X_2
9	2	1	0	1	0	$\overline{X_1}\overline{X_2} + X_1\overline{X_2}$	$\overline{X_2}$
10	2	1	0	0	1	$\overline{X_1}\overline{X_2} + X_1X_2$	$X_1X_2 + \overline{X_1}\overline{X_2}$
11	2	0	1	1	0	$\overline{X_1}X_2 + X_1\overline{X_2}$	$X_1\overline{X_2} + \overline{X_1}X_2$
12	3	0	1	1	1	$\overline{X_1}X_2 + X_1\overline{X_2} + X_1X_2$	$X_1 + X_2$
13	3	1	0	1	1	$\overline{X_1}\overline{X_2} + X_1\overline{X_2} + X_1X_2$	$X_1 + \overline{X_2}$
14	3	1	1	0	1	$\overline{X_1}\overline{X_2} + \overline{X_1}X_2 + X_1X_2$	$\overline{X_1} + X_2$
15	3	1	1	1	0	$\overline{X_1}\overline{X_2} + \overline{X_1}X_2 + X_1\overline{X_2}$	$\overline{X_1} + \overline{X_2}$
16	4	1	1	1	1	$\overline{X_1}\overline{X_2} + \overline{X_1}X_2 + X_1\overline{X_2} + X_1X_2$	1

如果用真值表来表示某开关函数, 那么由下述规则可以写出积-和范式: 真值表映射等于 1 的各行可以得到开关函数的积-和范式的最小项; 变量为 0, 表示该变量的非, 变量为 1 则表示该变量本身。

类似地, 一个开关函数的和-积范式的最大项, 可从其真值表映射为 0 的各行得到, 变量值为 0, 表示变量本身, 变量值为 1 表示该变量的非。

例 1-2 某开关函数的真值表为:

表 1-3 真 值 表

X_1	X_2	X_3	F	X_1	X_2	X_3	F	X_1	X_2	X_3	F
0	0	0	0	0	1	1	1	1	1	0	0
0	0	1	0	1	0	0	1	1	1	1	1
0	1	0	1	1	0	1	0				

写出它的积-和范式及和-积范式。

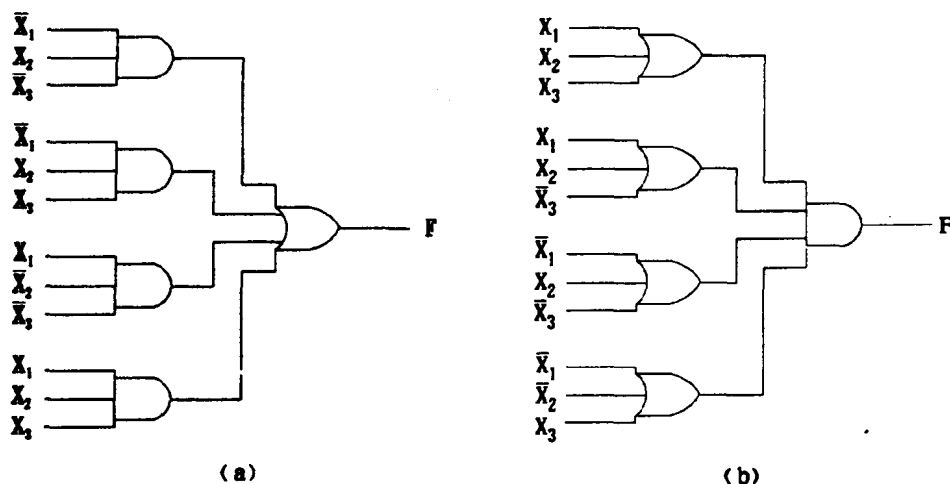
积-和范式为：

$$F = \bar{X}_1 \cdot X_2 \cdot \bar{X}_3 + \bar{X}_1 \cdot X_2 \cdot X_3 + X_1 \cdot \bar{X}_2 \cdot \bar{X}_3 + X_1 \cdot X_2 \cdot X_3 \quad (1-2a)$$

和-积范式为：

$$F = (X_1 + X_2 + X_3) \cdot (X_1 + X_2 + \bar{X}_3) \cdot (\bar{X}_1 + X_2 + \bar{X}_3) \cdot (\bar{X}_1 + \bar{X}_2 + X_3) \quad (1-3a)$$

如果用与门及或门来实现上述开关函数,那么例 1-2 中积-和范式与和-积范式分别可用图 1-5(a)及(b)来实现。



(a)与-或逻辑对应于积-和范式；

(b)或-与逻辑对应于和-积范式

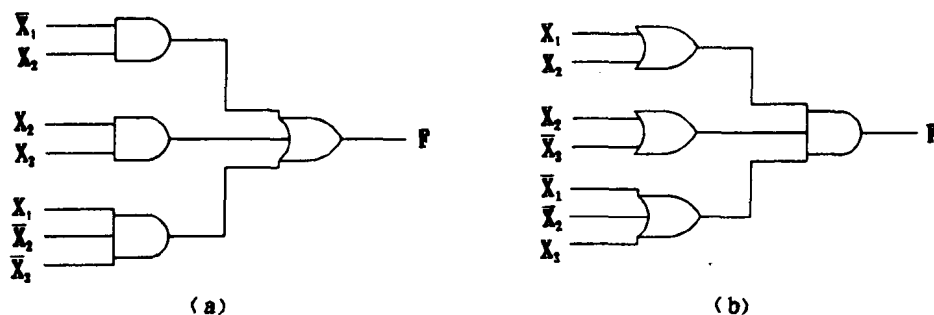
图 1-5 例 1-2 所示真值表的二级门实现

为减少门数及门的输入端个数,在用门实现开关函数时,还必须把函数化简,例 1-2 的开关函数经化简后可得：

$$F(X_1, X_2, X_3) = \bar{X}_1 \cdot X_2 + X_1 \cdot \bar{X}_2 \cdot \bar{X}_3 + X_2 \cdot X_3 \quad (\text{积-和范式}) \quad (1-2b)$$

$$F(X_1, X_2, X_3) = (X_1 + X_2) \cdot (X_2 + \bar{X}_3) \cdot (\bar{X}_1 + \bar{X}_2 + X_3) \quad (\text{和-积范式}) \quad (1-3b)$$

简化后的电路如图 1-6 所示。



(a)简化后的与-或逻辑；

(b)简化后的或-与逻辑

图 1-6 图 1-5 的简化电路