



手把手教你学系列丛书



手把手教你学 CPLD/FPGA 与单片机联合设计

周兴华 李玉丽 傅飞峰 编著



新增多媒体教学例程



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

手把手教你学系列丛书

手把手教你学 CPLD/FPGA 与单片机联合设计

周兴华 李玉丽 傅飞峰 编著

北京航空航天大学出版社

内 容 简 介

作者从2009年1月起,在《电子世界》杂志上连载了《手把手教你学 CPLD/FPGA 设计》讲座。本书以此为蓝本,并增加了大量的篇幅与实验例子进行充实。此外,为了帮助读者掌握 CPLD/FPGA 与单片机的联合设计,还介绍了 MCS-51 单片机的基本知识及单片机 C 语言编程的基础知识,并通过实例设计进行详解。

本书以实践(实验)为主线,以生动短小的实例为灵魂,穿插介绍了 Verilog HDL 语言的语法及 Altera 公司的 EPM7128S(或 Atmel 公司的 ATF1508AS)设计开发编程,理论与实践紧密结合,由浅入深、循序渐进地引导读者进行学习、实验,这样读者学得进、记得牢,不会产生畏难情绪,无形之中就掌握了 CPLD/FPGA 与单片机的联合设计。

本书贯彻《手把手教你学系列丛书》的教学方式。书中附有光盘,含本书所有的程序设计文件,可用作大学本科或专科、中高等职业技术学校、电视大学等的教学用书,也可作为 CPLD/FPGA 爱好者的自学用书。

图书在版编目(CIP)数据

手把手教你学 CPLD/FPGA 与单片机联合设计 / 周兴华,
李玉丽,傅飞峰编著. — 北京:北京航空航天大学出
版社,2010.11

ISBN 978-7-5124-0244-7

I. ①手… II. ①周… ②李… ③傅… III. ①可编程
序逻辑器件—系统设计②单片微型计算机—系统设计
IV. ①TP332.1②TP368.1

中国版本图书馆 CIP 数据核字(2010)第 207943 号

版权所有,侵权必究。

手把手教你学 CPLD/FPGA 与单片机联合设计

周兴华 李玉丽 傅飞峰 编著

责任编辑 杨 昕 刘爱萍

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:bhpress@263.net 邮购电话:(010)82316936

北京时代华都印刷有限公司印装 各地书店经销

*

开本:787×1092 1/16 印张:19.25 字数:493 千字

2010 年 11 月第 1 版 2010 年 11 月第 1 次印刷 印数:5 000 册

ISBN 978-7-5124-0244-7 定价:39.00 元(含光盘 1 张)

前言

CPLD/FPGA 是什么样的器件? 它起什么作用? 它与单片机是怎样的关系? 为什么学会了单片机的设计, 还要再学习 CPLD/FPGA 的设计? 刚接触本书的读者, 很可能会有这些疑问。

- CPLD(Complex Programmable Logic Device), 复杂可编程逻辑器件的英语缩写。
- FPGA(Field Programmable Gate Array), 现场可编程门阵列的英语缩写。

尽管 CPLD 与 FPGA 的结构不同, 但从应用的角度来看均属于可编程逻辑器件(Programmable Logic Device, PLD)的范畴。

接下来的问题是: 可编程逻辑器件 PLD 有什么用(或者起什么作用)? 它与单片机是怎样的关系?

为了说清楚这件事, 先举一个例子: 如果需要制作一个 50 MHz 的频率计, 仅使用单片机显然是不可能实现的, 因为单片机无法对高达 50 MHz 的频率信号进行计数及处理。

那怎么办呢? 比较可行的方法是: 先用数字逻辑电路对高达 50 MHz 的频率信号进行分频、计数、锁存, 然后将测得的信号再交给单片机进行运算处理、显示、输出控制等。因为数字逻辑电路的工作频率比较高, 可以满足几十至几百 MHz 的信号处理。但还有问题: 完成这些工作的数字逻辑电路, 需要十几片至几十片的通用数字逻辑集成电路芯片, 显然结构太复杂了, 可靠性也低。这个时候, 可编程逻辑器件(CPLD 或 FPGA)就可以大显身手了, 可以直接使用 PLD 芯片进行数字电路系统的设计, 将分频、计数、锁存等功能通过软件编程的方法设计在 PLD 的芯片内部, 使得从原来的印板级设计上升到芯片级设计, 大大缩小了印板的体积, 提高了可靠性。而单片机的特长是使用方便, 运算精确灵活, 控制能力强, 将 CPLD/FPGA 与单片机结合起来应用之后, 充分发挥了它们各自的特长, 使其优势互补。这样整个系统的结构简单、功能强大、性价比非常高。

刚才只是举了一个很小的例子, 实际上 PLD 所能完成的工作远不止这些, 小至各种门电路、计数器、触发器、锁存器, 大到雷达信号处理器、激光控制器, 都可以用 PLD 来实现, 甚至还可以用 PLD 直接构造出 CPU 内核。现在明白了吧, 可编程逻辑器件 PLD 主要应用于单片机无法胜任的高频数字逻辑领域。因此, 学会单片机之后, 还需要学习 CPLD/FPGA 的设计。

本书贯彻《手把手教你学系列丛书》的教学方式(本书为《手把手教你学系列丛书》之一), 由浅入深, 一步一步带领读者学会 CPLD/FPGA 与单片机的联合设计。

《手把手教你学系列丛书》是一套帮助初学者入门及学会真本事的丛书, 它以实践(实验)为主线, 理论与实践紧密结合, 使读者学得会、学得好、学得快, 深受国内外读者的喜爱, 出版 5 年来, 累计发行近 10 万册, 迄今已有数十万读者通过本系列丛书学会了单片机设计, 创造了很好的经济效益与社会效益。

本书的学习成本较低, 学习时采用“CPLD(或单片机)下载程序→试验板通电实验”的方

法,这样实验器材的配置只有 400 元左右。

随书所附的光盘中提供了本书的所有软件设计程序文件,可供读者朋友参考。

参与本书编写的主要工作人员有周兴华、李玉丽、吕超亚、傅飞峰、周济华、沈惠莉、周渊、周国华、丁月妹、周晓琼、钱真、周桂华、刘卫平、周军、李德英、朱秀娟、刘君礼、毛雪琴、邱华锋、胡颖静、吴辉东、冯骏、孔雪莲、王铨、方渝、刘郑州、王菲、付毛仙、吕丁才、唐群苗、吕亚波等,全书由周兴华统稿并审校。

本书的编写工作得到了我国单片机权威何立民教授的关注与鼓励,北京航空航天大学出版社嵌入式系统事业部主任胡晓柏老师也做了大量耐心细致的工作,使得本书得以顺利完成,在此表示衷心感谢。

由于作者水平有限,书中还存在不少缺点或漏洞,诚挚欢迎广大读者提出意见并不吝赐教。

周兴华

2010 年 9 月

一本优秀的入门书籍和一套与之相配的实验器材是学习的必要条件,在此前提下,加上自己的刻苦努力、持之以恒,才能在最短时间内学会、学好 CPLD/FPGA 与单片机联合设计。

如读者朋友自制或购买书中介绍的学习、实验器材有困难,则可与作者联系,咨询购买事宜。

本书所配的实验器材如下:

- Keil C51、MaxPlusII 10. 2、QuartusII8. 0 集成开发环境、Atmel ISP6. 7 下载软件、Pof2Jed 转换软件;
- MCU & CPLD DEMO 试验板(配 AT89S51 及 ATF1508AS);
- CPLD/FPGA JTAG 并口程序下载器;
- 单片机 USB 程序下载器;
- 16×2 字符型液晶显示模组;
- 9 V/800 mA 专用电源。

联系方式如下:

地址:上海市闵行区莲花路 2151 弄 57 号 201 室

邮编:201103

联系人:周兴华

电话(传真):021-64654216 13774280345 13044152947

技术支持 E-mail: zzh2151@sohu.com

zzh2151@yahoo.com.cn

笔者主页: <http://www.hlelectron.com>



第 1 章 可编程逻辑器件概述	1
1.1 可编程逻辑器件简介	1
1.1.1 可编程逻辑器件的基本结构	2
1.1.2 可编程逻辑器件的分类及特点	2
1.1.3 可编程逻辑器件的逻辑表示方法	4
1.2 CPLD/FPGA 的结构与特性	5
1.2.1 基于乘积项的 CPLD 原理与结构	6
1.2.2 基于乘积项的 CPLD 逻辑实现方式	7
1.2.3 基于查找表的 FPGA 原理与结构	8
1.2.4 基于查找表的 FPGA 逻辑实现方式	9
1.2.5 CPLD 与 FPGA 器件的差别	9
1.3 Altera 公司的 MAX7000 系列 CPLD 特性介绍	12
1.3.1 逻辑阵列块(LAB)	14
1.3.2 宏单元	15
1.3.3 扩展乘积项	16
1.3.4 可编程连线阵列	17
1.3.5 I/O 控制块	17
1.3.6 其他特性	18
第 2 章 可编程逻辑器件的设计流程及学习开发器材	20
2.1 可编程逻辑器件的设计流程	20
2.1.1 设计输入	20
2.1.2 综 合	22
2.1.3 CPLD/FPGA 器件适配	23
2.1.4 仿 真	23
2.1.5 编程下载	24
2.2 CPLD/FPGA 与单片机联合设计的学习器材介绍	24
2.2.1 Altera 公司的集成开发软件 MAX+plus II 及 Quartus II	24
2.2.2 Keil C51 Windows 集成开发环境	25
2.2.3 MCU & CPLD DEMO 综合试验板	26
2.2.4 ByteBlaster MV 并口下载器	29
2.2.5 单片机 USB 程序下载器	30
2.2.6 9 V 高稳定专用稳压电源	30

第 3 章 开发软件的安装	32
3.1 Keil C51 集成开发软件安装	32
3.2 MAX+plus II 集成开发软件安装	33
3.3 Quartus II 集成开发软件安装	38
3.4 USBasp 下载器的安装与使用	49
3.4.1 USBasp 下载器的安装	49
3.4.2 USBasp 下载器的使用	51
3.5 Atmel 并口下载软件 atmelisp 的安装	54
3.6 POF to JED 转换软件 Pof2jed 的安装	55
第 4 章 第一个 CPLD/FPGA 入门实验程序	56
4.1 使用 Max+plus II 集成开发软件进行入门实验	56
4.1.1 建立项目	56
4.1.2 设计输入(原理图或硬件描述语言)	56
4.1.3 选择器件并锁定引脚	59
4.1.4 编译器件	60
4.1.5 仿 真	62
4.1.6 编程下载	67
4.1.7 应 用	72
4.2 使用 Quartus II 集成开发软件进行入门实验	72
4.2.1 建立项目	73
4.2.2 设计输入(原理图或硬件描述语言)	76
4.2.3 设计编译	79
4.2.4 仿 真	79
4.2.5 引脚分配	91
4.2.6 编程下载	94
4.2.7 应 用	94
第 5 章 Verilog HDL 硬件描述语言	95
5.1 Verilog HDL 模块的基本结构	95
5.1.1 模块声明	95
5.1.2 端口定义	96
5.1.3 信号类型说明	96
5.1.4 逻辑功能描述	96
5.1.5 实验程序 1——缓冲器	98
5.1.6 实验程序 2——反相器(非门)	98
5.2 Verilog HDL 语法要素	99
5.2.1 标识符与关键字	99
5.2.2 常量、变量及数据类型	100
5.2.3 实验程序 3——与门	102
5.2.4 实验程序 4——与非门	103

5.2.5 实验程序 5——LED 的闪烁	103
5.2.6 运算符	104
5.2.7 运算符的优先级	108
5.2.8 实验程序 6——或门	109
5.2.9 实验程序 7——或非门	109
5.2.10 实验程序 8——异或门	110
5.2.11 实验程序 9——异或非门	111
5.2.12 实验程序 10——三态门	111
5.3 Verilog HDL 的行为语句	112
5.3.1 赋值语句	113
5.3.2 过程语句	113
5.3.3 块语句	115
5.3.4 条件语句	117
5.3.5 循环语句	118
5.3.6 编译预处理	119
5.3.7 任务和函数	121
5.4 Verilog HDL 数字逻辑单元结构的设计	122
5.4.1 结构描述方式	122
5.4.2 实验程序——门级结构描述设计的基本门电路	125
5.4.3 数据流描述方式	127
5.4.4 行为描述方式	127
第 6 章 组合逻辑电路的设计实验	129
6.1 2 选 1 数据选择器	129
6.1.1 2 选 1 数据选择器简介	129
6.1.2 采用数据流描述方式的设计	130
6.1.3 采用行为描述方式的设计	131
6.2 4 选 1 数据选择器	131
6.2.1 4 选 1 数据选择器简介	131
6.2.2 采用数据流描述方式的设计	132
6.2.3 采用行为描述方式的设计	133
6.3 3 位二进制优先编码器(8-3 优先编码器)	134
6.3.1 3 位二进制优先编码器简介	134
6.3.2 3 位二进制优先编码器的设计	135
6.4 3 位二进制译码器(3-8 译码器)	136
6.4.1 3 位二进制译码器简介	136
6.4.2 3 位二进制译码器的设计	137
6.5 BCD-7 段译码器	139
6.5.1 BCD-7 段译码器简介	139
6.5.2 BCD-7 段译码器的设计	139

6.6 半加器	141
6.6.1 半加器简介	141
6.6.2 采用门级描述方式的半加器设计	142
6.6.3 采用数据流描述方式的半加器设计	142
6.6.4 采用行为描述方式的半加器设计	143
6.7 全加器	144
6.7.1 全加器简介	144
6.7.2 全加器的设计	144
第7章 触发器的设计实验	146
7.1 RS 触发器	146
7.1.1 RS 触发器简介	146
7.1.2 RS 触发器的设计	146
7.2 JK 触发器	148
7.2.1 JK 触发器简介	148
7.2.2 JK 触发器的设计	148
7.3 带有复位的 JK 触发器	150
7.3.1 带有复位的 JK 触发器简介	150
7.3.2 带有复位的 JK 触发器的设计	150
7.4 D 触发器	152
7.4.1 D 触发器简介	152
7.4.2 D 触发器的设计	153
7.5 带有复位的 D 触发器	154
7.5.1 带有复位的 D 触发器简介	154
7.5.2 带有复位的 D 触发器的设计	154
7.6 带有复位的异步 T 触发器	156
7.6.1 带有复位的异步 T 触发器简介	156
7.6.2 带有复位的异步 T 触发器的设计	156
7.7 带有复位的同步 T 触发器	158
7.7.1 带有复位的同步 T 触发器简介	158
7.7.2 带有复位的同步 T 触发器的设计	158
第8章 时序逻辑电路的设计实验	160
8.1 寄存器	160
8.1.1 寄存器简介	160
8.1.2 寄存器的设计	161
8.2 锁存器	162
8.2.1 锁存器简介	162
8.2.2 锁存器的设计	163
8.3 移位寄存器	164
8.3.1 移位寄存器简介	164

8.3.2 移位寄存器的设计	165
8.4 计数器	167
8.4.1 4 位二进制异步加法计数器简介	167
8.4.2 4 位二进制异步加法计数器的设计	168
8.4.3 十进制(任意进制)同步加法计数器简介	170
8.4.4 十进制同步加法计数器的设计	170
第 9 章 CPLD/FPGA 的设计应用	172
9.1 跑马灯实验	172
9.1.1 实验要求	172
9.1.2 实现方法	172
9.1.3 程序设计	172
9.2 多位数码管的动态扫描显示	174
9.2.1 实验要求	174
9.2.2 实现方法	174
9.2.3 程序设计	174
9.3 蜂鸣器发声实验	177
9.3.1 实验要求	177
9.3.2 实现方法	177
9.3.3 程序设计	177
9.4 简易电子琴实验	178
9.4.1 实验要求	178
9.4.2 实现方法	178
9.4.3 程序设计	179
9.5 驱动字符型液晶显示器实验	180
9.5.1 实验要求	180
9.5.2 字符型液晶控制器的指令简介	180
9.5.3 字符型液晶控制器的工作时序	183
9.5.4 时序参数	184
9.5.5 实现方法	184
9.5.6 程序设计	184
9.6 串口接收实验	188
9.6.1 实验要求	188
9.6.2 实现方法	188
9.6.3 程序设计	188
9.7 串口发送实验	192
9.7.1 实验要求	192
9.7.2 实现方法	193
9.7.3 程序设计	193
9.8 RS232 收发实验	197

9.8.1 实验要求	197
9.8.2 实现方法	197
9.8.3 程序设计	198
9.9 RS232 收发不同内容的实验	204
9.9.1 实验要求	204
9.9.2 实现方法	205
9.9.3 程序设计	205
9.10 简易数字电子钟	212
9.10.1 实验要求	212
9.10.2 实现方法	212
9.10.3 程序设计	212
第 10 章 51 单片机的基本知识	217
10.1 51 单片机的基本结构	217
10.2 80C51 基本特性及引脚定义	218
10.2.1 80C51 的基本特征	218
10.2.2 80C51 的引脚定义及功能	219
10.3 80C51 的内部结构	220
10.4 80C51 的存储器配置和寄存器	222
第 11 章 单片机 C 语言基础知识	225
11.1 C 语言的标识符与关键字	225
11.2 数据类型	227
11.3 常量、变量及存储类型	227
11.4 数 组	230
11.4.1 一维数组的定义	231
11.4.2 二维及多维数组的定义	231
11.4.3 字符数组	232
11.4.4 数组元素赋初值	232
11.4.5 数组作为函数的参数	233
11.5 C 语言的运算	233
11.5.1 算术运算符	233
11.5.2 关系运算符	234
11.5.3 逻辑运算符	234
11.5.4 赋值运算符	235
11.5.5 自增和自减运算符	235
11.5.6 逗号运算符	236
11.5.7 条件运算符	236
11.5.8 位运算符	236
11.5.9 sizeof 运算符	236
11.6 流程控制	237

11.6.1	条件语句与控制结构	237
11.6.2	循环语句	239
11.7	函 数	241
11.7.1	函数定义的一般形式	241
11.7.2	函数的参数和函数返回值	242
11.7.3	函数调用的三种方式	242
11.8	指 针	243
11.8.1	指针与地址	244
11.8.2	指针变量的定义	244
11.8.3	指针变量的引用	244
11.8.4	数组指针与指向数组的指针变量	245
11.8.5	指针变量的运算	246
11.8.6	指向多维数组的指针和指针变量	246
11.9	结构体	247
11.9.1	结构体的概念	247
11.9.2	结构体类型变量的定义	247
11.9.3	结构体类型需要注意的地方	249
11.9.4	结构体变量的引用	249
11.9.5	结构体变量的初始化	250
11.9.6	结构体数组	250
11.9.7	指向结构体类型数据的指针	250
11.9.8	用指向结构体变量的指针引用结构体成员	251
11.9.9	指向结构体数组的指针	251
11.9.10	将结构体变量和指向结构体的指针作函数参数	251
11.10	共用体	252
11.10.1	共用体类型变量的定义	252
11.10.2	共用体变量的引用	253
11.11	中断函数	253
11.11.1	什么是中断	253
11.11.2	中断响应及 C51 编程	254
11.11.3	51 单片机的常用中断源和中断向量	255
11.11.4	编写 51 单片机中断函数时应严格遵循的规则	255
第 12 章	CPLD/FPGA 与单片机的接口及数据传输	257
12.1	CPLD/FPGA 与单片机 AT89S51 的接口连接及数据传输实验	257
12.1.1	实验要求	257
12.1.2	实现方法	257
12.1.3	CPLD/FPGA 程序设计	258
12.1.4	单片机程序设计	260
12.2	单片机直接访问方式驱动液晶	267

12.2.1 实验要求	267
12.2.2 实现方法	267
12.2.3 CPLD/FPGA 程序设计	268
12.2.4 单片机程序设计	269
12.3 单片机间接控制方式驱动液晶	273
12.3.1 实验要求	273
12.3.2 实现方法	273
12.3.3 CPLD/FPGA 程序设计	274
12.3.4 单片机程序设计	274
第 13 章 CPLD/FPGA 与单片机的联合设计实例——液晶显示频率计	279
13.1 设计要求	279
13.2 实现方法	279
13.2.1 CPLD/FPGA 的功能设计	279
13.2.2 单片机的功能设计	280
13.3 CPLD/FPGA 程序设计	280
13.4 单片机程序设计	286
参考文献	293

第 1 章

可编程逻辑器件概述

多年来,人们设计数字电路系统都是使用标准的数字集成电路芯片,如 74/54 系列(TTL)、4000/4500 系列(CMOS)等,根据设计的功能从这些标准的芯片中进行选择,然后搭建成一个完整的数字电路应用系统。使用这种方法设计出来的系统,不仅芯片数量多、印板面积大,而且可靠性差,毫无设计的灵活性可言。

可编程逻辑器件 PLD(Programmable Logic Device)出现后,改变了人们的传统设计方法,可以直接使用 PLD 芯片进行数字电路系统的设计。例如,可以直接设计芯片内部的数字逻辑并定义输入/输出引脚等,从原来的印板级设计上升到芯片级设计。由于 PLD 设计时引脚定义非常灵活,不仅降低了电路原理和印板设计的难度,提高了设计效率,而且大大减少了芯片的数量和种类,缩小了印板面积,降低了功耗,并极大地提高了系统工作的可靠性。

1.1 可编程逻辑器件简介

可编程逻辑器件是 20 世纪 70 年代发展起来的一种新型器件,它给数字系统的设计方式带来了革命性的变化。

PLD 器件最早是 20 世纪 70 年代中期出现的可编程逻辑阵列 PLA (Programmable Logic Array),PLA 在结构上由可编程的与阵列和可编程的或阵列构成,阵列规模比较小,编程也很烦琐,并没有得到广泛应用。随后出现了可编程阵列逻辑 PAL(Programmable Array Logic),PAL 由可编程的与阵列和固定的或阵列组成,采用熔丝编程方式,它的设计比较灵活,器件速度快,因而成为第一个得到普遍应用的 PLD 器件。

20 世纪 80 年代初,Lattice 公司发明了通用阵列逻辑 GAL(Generic Array Logic)。GAL 器件采用了输出逻辑宏单元(OLMC)的结构和 EEPROM 工艺,具有可编程、可擦除、可长期保持数据的优点,使用灵活,所以 GAL 得到了极为广泛的应用,迄今还在大量使用。

80 年代中期,Altera 公司推出了一种新型的可擦除、可编程的逻辑器件 EPLD(Erasable Programmable Logic Device),EPLD 采用 CMOS 和 UVEPROM 工艺制成,集成度更高,设计也更灵活,但它的内部连线功能并不是很强。

EPLD 经 Lattice 公司改进后就成为 CPLD(Complex Programmable Logic Device),即复杂可编程逻辑器件,采用 EEPROM 工艺制作。与 EPLD 相比,CPLD 增强了内部连线,对逻

辑宏单元和 I/O 单元也有重大的改进,它的性能更好,使用更方便。并且,现在的大部分 CPLD 都具备在系统编程(ISP)功能。CPLD 是当前的主流 PLD 器件之一。

1985 年,Xilinx 公司推出了现场可编程门阵列 FPGA (Field Programmable Gate Array),这是一种采用单元型结构的新型 PLD 器件。它采用 CMOS 的 SRAM 工艺制作,在结构上和阵列型 PLD 不同,它的内部由许多独立的可编程逻辑单元构成,各逻辑单元之间可以灵活地相互连接,具有密度高、速度快、编程灵活、可重新配置等优点,FPGA 也是当前主流的 PLD 器件之一。

1.1.1 可编程逻辑器件的基本结构

可编程逻辑器件的基本结构如图 1-1 所示,由输入控制电路、与阵列、或阵列以及输出控制电路组成。在输入控制电路中,输入信号经过输入缓冲单元产生每个输入变量的原变量和反变量,并作为与阵列的输入项。与阵列由若干个与门组成,输入缓冲单元提供的各输入项被有选择地连接到各个与门输入端,每个与门的输出则是部分输入变量的乘积项。各与门输出又作为或阵列的输入,这样或阵列的输出就是输入变量的与或形式。输出控制电路将或阵列输出的与或式通过三态门、寄存器等电路,一方面产生输出信号,另一方面作为反馈信号送回输入端,以便实现更复杂的逻辑功能。因此,利用可编程逻辑器件可以方便地实现各种逻辑功能。

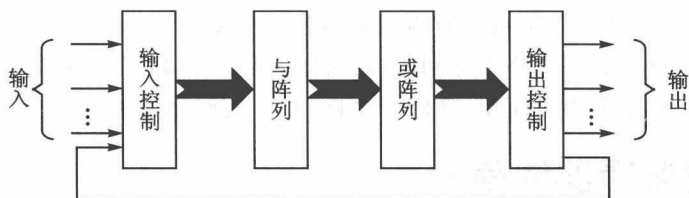


图 1-1 可编程逻辑器件的基本结构

1.1.2 可编程逻辑器件的分类及特点

表 1-1 可编程逻辑器件按集成度分类示意

PLD	LDPLD	PROM
		PLA
		PAL
		GAL
	HDPLD	CPLD
		FPGA

可编程逻辑器件按照不同的类型和标准,可以有多种分类方法。如按器件的集成度划分,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)。常见的低密度可编程逻辑器件有 PROM、PLA、PAL 和 GAL 等,通常称为简单 PLD 器件;常见的高密度可编程逻辑器件有 CPLD 以及 FPGA 等。表 1-1 所列为可编程逻辑器件按集成度分类示意。

1. 低密度可编程逻辑器件

根据可编程逻辑器件的“与”阵列和“或”阵列的编程情况以及输出形式,低密度可编程逻辑器件(LDPLD)通常可分为四类。

(1) 与阵列固定、或阵列可编程的 PLD 器件

这类 PLD 器件以可编程只读存储器 PROM 为代表。可编程只读存储器 PROM 是组合逻辑阵列,它包含一个固定的与阵列和一个可编程的或阵列。PROM 中的与阵列是全译码形式,它产生 n 个输入变量的所有最小项。PROM 的每个输出端通过或阵列将这些最小项有选择地进行或运算,即可实现任何组合逻辑函数。由于与阵列能够产生输入变量的全部最小项,所以用 PROM 实现组合逻辑函数不需要进行逻辑化简。但是随着输入变量数的增加,与阵列的规模会迅速增大,其价格也随之大大提高。而且与阵列越大,译码开关时间就越长,相应的工作速度也越慢。因此,实际上只有规模较小的 PROM 可以有效地实现组合逻辑函数,而大规模的 PROM 价格高,工作速度低,一般只做存储器使用。

(2) 与阵列和或阵列均可编程的 PLD 器件

以可编程逻辑阵列 PLA 为代表。PLA 和 PROM 一样也是组合型逻辑阵列,与 PROM 不同的是它的两个逻辑阵列均可编程。PLA 的与阵列不是全译码形式,它可以通过编程控制只产生函数最简与或式中所需要的与项。因此 PLA 器件的与阵列规模较小,集成度相对高一些。

但是,由于 PLA 只产生函数最简与或式中所需要的与项,因此 PLA 在编程前必须先进行函数化简。另外,PLA 器件需要对两个阵列进行编程,编程难度较大。而且,PLA 器件的开发工具应用不广泛,编程一般只能由生产厂家完成。

(3) 以可编程阵列逻辑 PAL 为代表的与阵列可编程、或阵列固定的 PLD 器件

这种器件的每个输出端是若干个乘积项之或,其中乘积项的数目固定。通常 PAL 的乘积项数允许达到 8 个,而一般逻辑函数的最简与或式中仅需要完成 3~4 个乘积项或运算。因此,PAL 的这种阵列结构可以满足大多数逻辑函数的设计要求。

PAL 有几种固定的输出结构,如专用输出结构、可编程 I/O 结构、带反馈的寄存器输出结构以及异或型输出结构等。一定的输出结构只能实现一定类型的逻辑函数,因此,PAL 的通用性较差。

(4) 具有可编程输出逻辑宏单元的通用 PLD 器件

以通用型可编程阵列逻辑 GAL 器件为主要代表,GAL 器件的阵列结构与 PAL 相同,都是采用与阵列可编程及或阵列固定的形式,两者的主要区别是输出结构不同:PAL 的输出结构是固定的,一种结构对应一种类型芯片,如果系统中需要几种不同的输出形式,就必须选择多种芯片来实现;GAL 器件的每个输出端都集成有一个输出逻辑宏单元 OLMC(Out Logic Macro Cell),输出逻辑宏单元是可编程的,通过编程可以决定该电路是完成组合逻辑还是时序逻辑,是否需要产生反馈信号,并能实现输出使能控制以及输出极性选择等。因此,GAL 器件通过对输出逻辑宏单元 OLMC 的编程可以实现 PAL 的各种输出结构,使芯片具有很强的通用性和灵活性。

2. 高密度可编程逻辑器件

高密度可编程逻辑器件(HDPLD)主要包括 CPLD 和 FPGA 两类器件,这两类器件也是当前 PLD 的主流应用器件。

CPLD 是基于乘积项(Product - Term)技术,采用 Flash(或 EEPROM)工艺制作的 PLD 器件,配置数据掉电后不会丢失,一般多用于 5 000 门以下的中小规模设计,适合做复杂的组

合逻辑,如译码器等。

FPGA 采用静态存储器(SRAM)结构,属于单元型的 PLD 器件,它的基本结构是可编程逻辑块,由许多这样的逻辑块排列成阵列状,逻辑块之间由水平连线和垂直连线通过编程连通。FPGA 器件采用查找表(Look - Up Table)技术及 SRAM 工艺,配置数据易失,需要外挂非易失性器件进行配合。FPGA 的集成度高(其密度远高于 CPLD),触发器多,多用于较大规模的设计,适合做复杂的时序逻辑,如数字信号处理和各种算法等。

1.1.3 可编程逻辑器件的逻辑表示方法

由于可编程逻辑器件的阵列结构特点,现在广泛采用如下的逻辑表示方法。

1. PLD 输入缓冲单元

PLD 的输入缓冲单元由若干个缓冲器组成,每个缓冲器产生该输入变量的原变量和反变量,其逻辑表示方法如图 1-2 所示,表 1-2 是它所对应的真值表。

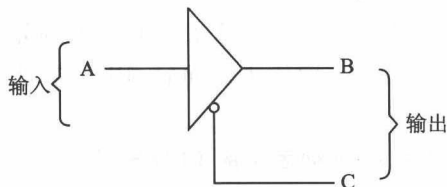


图 1-2 PLD 输入缓冲单元逻辑表示方法

表 1-2 PLD 输入缓冲单元真值表

A	B	C
0	0	1
1	1	0

2. PLD 与门

以三输入与门为例,其 PLD 表示法如图 1-3 所示。A、B、C 为输入项,D 为输出项。其中 $D=A \times B \times C$

3. PLD 或门

以三输入或门为例,其 PLD 表示法如图 1-4 所示。A、B、C 为输入项,D 为输出项。其中 $D=A+B+C$

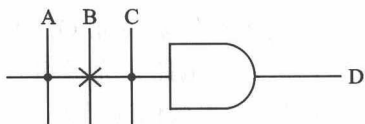


图 1-3 PLD 三输入与门

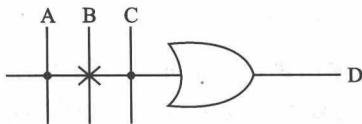


图 1-4 PLD 三输入或门

4. PLD 连接方式

PLD 有 3 种不同的连接方式:固定连接、可编程连接以及断开,其表示方法如图 1-5 所示。

图 1-5(a)表示的固定连接是厂家在生产芯片时连好的,是不可改变的。图 1-5(b)表示可编程连接,在熔丝编程工艺的 PLD 中(如 PAL),接通对应于熔丝未熔断;断开对应于熔丝