



“十四五”职业教育国家规划教材

职业教育国家在线精品课程配套教材

EDA技术与应用 项目化教程 ——Verilog HDL版

主编◎张平华 谭立新



上海交通大学出版社
SHANGHAI JIAO TONG UNIVERSITY PRESS

内容提要

《EDA 技术与应用项目化教程——Verilog HDL 版》为职业教育国家在线精品课程“电子设计自动化技术”的配套用书。本书以提高工程设计能力为目的,选择电子设计自动化典型案例为主要载体,通过“项目导向、任务驱动”模式编撰,深入浅出地对 EDA 技术及相关知识做了系统和完整的介绍。本书的开发软件选用国内应用广泛的 Quartus II,硬件描述语言选用 Verilog HDL,依托三阶螺旋递进(初阶验证、进阶提高、高阶创新)设计思路组织相关项目。全书包括三人表决器设计、四位加法器设计、数字电子钟设计、声光警示器设计、交通灯控制器设计和数字温度计设计六个项目,每个项目从项目描述、知识准备、任务实现、考核评价、拓展提升和项目总结六个方面编写。本书取材广泛、内容新颖、重点突出,可作为应用电子技术、电子信息工程、通信工程等信息类专业及相关专业的教材,也可作为相关专业技术人员的参考书。

图书在版编目(CIP)数据

EDA 技术与应用项目化教程: Verilog HDL 版 / 张平
华, 谭立新主编. —上海: 上海交通大学出版社, 2023.8
ISBN 978-7-313-28074-9

I. ①E… II. ①张… ②谭… III. ①电子电路—电路
设计—计算机辅助设计—教材 IV. ①TN702.2

中国国家版本馆 CIP 数据核字(2023)第 162363 号

EDA 技术与应用项目化教程——Verilog HDL 版

EDA JISHU YU YINGYONG XIANGMUHUA JIAOCHENG——Verilog HDL BAN

主 编: 张平华 谭立新	地 址: 上海市番禺路 951 号
出版发行: 上海交通大学出版社	电 话: 021-6407 1208
邮政编码: 200030	
印 制: 北京荣玉印刷有限公司	经 销: 全国新华书店
开 本: 889mm × 1194mm 1/16	印 张: 15.5
字 数: 461 千字	
版 次: 2023 年 8 月第 1 版	印 次: 2023 年 8 月第 1 次印刷
书 号: ISBN 978-7-313-28074-9	
定 价: 56.00 元	

版权所有 侵权必究

告读者: 如发现本书有印装质量问题请与印刷厂质量科联系

联系电话: 010-6020 6144

目录

项目一 三人表决器设计 / 1

项目描述	2
一、EDA 技术及其发展	2
二、可编程逻辑器件	4
三、面向 CPLD/FPGA 的 EDA 设计流程	16
四、FPGA 生产厂商简介	19
五、基于 CPLD/FPGA 的常用 EDA 工具	21
六、硬件描述语言	25
七、任务实现——Quartus II 软件安装	34
八、任务实现——三人表决器设计	39
考核评价	58
拓展提升	59
项目总结	60

项目二 四位加法器设计 / 61

项目描述	62
一、四位加法器原理	62
二、Quartus II 原理图输入设计方法	64
三、任务实现——利用原理图输入法设计 3-8 译码器	64
四、任务实现——利用原理图输入法设计 四位加法器	79
考核评价	86
拓展提升	86
项目总结	88

项目三 数字电子钟设计 / 89

项目描述	90
一、Verilog 语言要素	91
二、Verilog 语句	105
三、任务实现——数字电子钟设计方案 分析	122
四、任务实现——数字电子钟的 Verilog 程序设计	123
考核评价	130
拓展提升	130
项目总结	132

项目四 声光警示器设计 / 133

项目描述	134
一、结构语句	134
二、Verilog 设计中 LPM 函数的应用	141
三、任务实现——声光警示器设计方案 分析	148
四、任务实现——声光警示器 Verilog 程序设计	150
考核评价	171
拓展提升	171
项目总结	173

项目五 交通灯控制器设计 / 174

项目描述 175

一、混合设计方式与数字系统验证 176

二、有限状态机设计 193

三、任务实现——交通灯控制器设计方案

 分析 201

四、任务实现——交通灯控制器设计 202

考核评价 210

拓展提升 210

项目总结 211

项目六 数字温度计设计 / 212

项目描述 213

一、温度传感器 214

二、单总线器件 214

三、任务实现——数字温度计设计方案

 分析 220

四、任务实现——数字温度计设计 222

考核评价 232

拓展提升 232

项目总结 234

附 录 本书所使用开发板功能简介 / 235

参考文献 / 237

项目一

三人表决器设计

学习目标

知识目标 >

- ① 了解 EDA 的概念、设计工具与开发流程；
- ② 了解 PLD 的概念、分类与基本结构；
- ③ 掌握 HDL 的种类、组成结构与设计示例；
- ④ 理解开发板硬件的结构与工作原理；
- ⑤ 掌握 Quartus II 软件的操作流程；
- ⑥ 了解 Verilog HDL 语言的特点、基本语法、基本单元及其构成。

能力目标 >

- ① 能说出 EDA 的内涵；
- ② 能进行 Quartus II 软件的安装；
- ③ 能用 Quartus II 进行文本设计方法设计；
- ④ 能用 Verilog HDL 语言设计简单的逻辑电路。

素质目标 >

- ① 树立公平公正民主意识，提升个人认同感；
- ② 培养尊重事实的实证精神；
- ③ 养成执着专注的工作习惯。

项目描述

情景引入



科技一小步
民主一大步

2022 年 3 月 10 日，中国人民政治协商会议第十三届全国委员会第五次会议闭幕。3 月 11 日，第十三届全国人民代表大会第五次会议闭幕。随着代表和委员们按下表决器，各项议程圆满完成。在代表和委员们按键表决的背后，稳定可靠的电子表决器为保障大会相关议程顺利进行立下了汗马功劳。代表们只需轻轻按下表决器上的签到、赞成、反对和弃权按钮即可快速完成签到和表决，签到和表决结果立即可以获得，大大缩短了会议时间，提高了会议效率，电子表决取代了举手或票箱表决，代表们可以更加方便地行使人民赋予的权力。

现某学校举行第六届“创意电子产品设计大赛”，大赛采用淘汰制，设置三名评委，当赞成评委超过半数时，参赛作品进入下一轮 PK 赛，现需设计制作一个三人表决器用于大赛评审结果表决。

任务分析



三人表决器原理
分析

本项目利用 Verilog HDL 语言，在 PLD（programmable logic device，可编程逻辑器件）开发平台上设计一个三人表决器，并用 FPGA（field programmable gate array，现场可编程门阵列）器件为载体实现最终的硬件电路。

三人表决器完成的逻辑功能：当同意人数为多数时（两人或两人以上），表决结果为通过；当不同意人数为多数时（两人或两人以上），表决结果为不通过。

其电路功能描述如下：三个人分别用拨码开关 KD1、KD2、KD3 来表达自己的意愿，如果对某项决议同意，对应的拨码开关拨到高电平（下方）；如果不同意，就把对应的拨码开关拨到低电平（上方）。表决结果用 LED 显示，如果决议通过，那么实验板上的 LED 点亮；如果决议不通过，那么实验板上的 LED 不亮；如果对某项决议有两到三人同意，那么此决议通过，LED 点亮；如果对某项决议只有一人同意或者没有人同意，那么此决议不通过，LED 不亮。

一、EDA 技术及其发展

（一）EDA 技术概念



EDA 概述

EDA 是电子设计自动化（electronic design automation）的缩写，它是在 90 年代初从 CAD（computer aided design，计算机辅助设计）、CAM（computer aided manufacturing，计算机辅助制造）、CAT（computer aided testing，计算机辅助测试）和 CAE（computer aided engineering，计算机辅助工程）的概念上发展而来的。

一般而言，EDA 技术的概念分狭义和广义两种。

狭义的 EDA 技术，就是以计算机为工作平台，以 EDA 软件工具为开发环境，以硬件描述语言为系统逻辑描述的主要表达方式，以 ASIC（application specific integrated circuit，专用集成电路）为实现载体的电子产品自动化设计的过程。狭义的 EDA 技术或称为 ASIC 自动设计技术，通过使用有关的开发软件，自动完成电子系统设计到硬件系统的逻辑编译、逻辑化简、逻辑分割、逻辑综合及优化、逻

辑布局布线、逻辑仿真，直至完成对于特定目标芯片的适配编译、逻辑映射、编程下载等工作，最终形成集成电子系统或专用集成芯片。本书讨论的对象专指狭义的 EDA 技术。

典型的 EDA 工具中必须包含两个特殊的软件包，即综合器和适配器。综合器的功能就是根据设计者在 EDA 平台上完成的针对某个系统项目的 HDL、原理图或状态图形描述，针对给定的硬件系统组件，进行编译、优化、转换和综合，最终获得可实现所需功能的描述文件。综合器在工作前，必须给定所要实现的硬件结构参数，它的功能就是将软件描述与给定的硬件结构用一定的方式联系起来。也就是说，综合器是软件描述与硬件实现的一座桥梁。综合过程就是将电路的高级语言描述转换为低级的、可与目标器件 FPGA/CPLD 相映射的网表文件。

适配器的功能是将综合器产生的网表文件配置到指定的目标器件中，产生最终的下载文件，如 JED 文件。适配所选定的目标器件（FPGA/CPLD 芯片）必须属于在综合器中已指定的目标器件系列。

硬件描述语言（hardware description language, HDL）是相对于一般的计算机软件语言（如 C、Python 等）而言的用于设计硬件电子系统的计算机语言，它能描述电子系统的逻辑功能、电路结构和连接方式。设计者可利用 HDL 程序来描述所希望得到的电路系统，规定器件结构特征和电路的行为方式，然后利用综合器和适配器将此程序编译成能控制 FPGA 和 CPLD 内部结构，并实现相应逻辑功能的门级或更底层的结构网表文件或下载文件。目前，就 FPGA/CPLD 开发来说，比较常用和流行的 HDL 主要有 Verilog HDL、VHDL、ABEL-HDL、AHDL 和 System Verilog 等。

广义的 EDA 技术，除了包括狭义的 EDA 技术外，还包括计算机辅助分析（computer aided analysis, CAA）技术（如 PSpice、EWB、MATLAB 等）和印制电路板计算机辅助设计 PCB-CAD 技术（如 Protel、OrCAD 等）。在广义的 EDA 技术中，CAA 技术和 PCB-CAD 技术不具备逻辑综合和逻辑适配的功能，因此它并不能称为真正意义上的 EDA 技术。

EDA 技术的 3 个层次：

- （1）EWB、PSpice、Protel 的学习作为 EDA 的最初级内容；
- （2）利用硬件描述语言完成对 CPLD / FPGA 的开发等作为中级内容；
- （3）ASIC 的设计作为最高级内容。

（二）EDA 技术的主要特征

EDA 技术主要有以下这些特征。

（1）高层次综合与优化的理论与方法取得了很大的进展，大大缩短了复杂的 ASIC 的设计周期，同时提高了设计质量。

（2）采用硬件描述语言来描述 10 万门以上的设计，形成了国际通用的 Verilog 等硬件描述语言。它们均支持不同层次的描述，使得复杂 IC 的描述规范化，便于传递、交流、保存和修改，并可建立独立的工艺设计文档，便于设计重用。

（3）开放式的设计环境（各厂家均适用）。

（4）“自顶向下”的算法。

（5）丰富的元器件模型库。

（6）建立并行设计工程框架结构的集成化设计环境，以适应当今 ASIC 的特点：规模庞大而复杂、数字与模拟电路并存、硬件与软件并存、产品上市更新快。

（三）EDA 技术发展历程

EDA 技术是伴随着计算机、集成电路和电子系统设计的兴起而发展起来的，至今已有 50 多年的

历史。EDA 技术的发展大致可以分为以下三个阶段。

第一阶段为 20 世纪 70 年代的 CAD（计算机辅助设计）阶段。这一阶段的主要特征是利用计算机辅助进行电路原理图编辑、PCB 布线，使得设计师从传统的高度重复繁杂的绘图劳动中解脱出来。

第二阶段为 20 世纪 80 年代的 CAED（计算机辅助工程设计）阶段。这一阶段的主要特征是以逻辑模拟、定时分析、故障仿真和自动布局布线为核心，重点解决电路设计的功能检测问题，使设计者能在产品制作之前预知产品的功能与性能。

第三阶段为 20 世纪 90 年代的 EDA（电子设计自动化）阶段。这一阶段的主要特征是以高级描述语言、系统仿真和综合技术为特点，采用自上而下的设计理念，将设计前期的许多高层次设计交给 EDA 工具来完成。

二、可编程逻辑器件

（一）可编程逻辑器件概述

1. 可编程逻辑器件的历史演变

可编程逻辑器件的发展主要可以分为以下五个阶段。

第一阶段：最早的可编程逻辑器件出现在 20 世纪 70 年代初，可编程器件只有简单的可编程只读存储器（programmable read only memory, PROM）、紫外线可擦除可编程只读存储器（erasable programmable read only memory, EPROM）和电可擦只读存储器（electrically erasable programmable read only memory, EEPROM）三种。由于结构的限制，它们只能完成简单的数字逻辑功能。

第二阶段：20 世纪 80 年代，出现了可编程阵列逻辑（programmable array logic, PAL）、可编程逻辑阵列（programmable logic array, PLA）和通用阵列逻辑（generic array logic, GAL）器件，可编程逻辑器件正式被称为 PLD。典型的 PLD 由与阵列和或阵列组成，用与或表达式来实现任意组合逻辑，所以 PLD 能以乘积和这一形式完成大量的逻辑组合。PAL 器件只能实现一次可编程，在编程以后无法修改。如果需要修改，则需要更换新的 PAL 器件，而 GAL 器件不需要进行更换，只要在原来器件上再次编程即可。

第三阶段：20 世纪 90 年代，众多可编程逻辑器件厂商推出了与标准门阵列类似的 FPGA 和类似于 PAL 结构的扩展性 CPLD。CPLD 提高了逻辑运算的速度，具有体系结构简单和逻辑单元灵活集成度高以及适用范围宽的特点，同时它兼容了 PLD 和通用门阵列的优点，能够实现超大规模的电路，编程方式灵活，成为产品原型设计和中小规模（一般小于 10000 门）产品生产的首选。

第四阶段：21 世纪初，将现场可编程门阵列和 CPU 融合，并且集成到一个单个的 FPGA 器件中，称为异构架构。Xilinx 推出了两种基于 FPGA 的嵌入式解决方案：内嵌 PowerPC 硬核微处理器、ARM Cortex-A9 双核硬核嵌入式处理器。Xilinx 还提供了低成本的嵌入式软核处理器，如 MicroBlaze、PicoBlaze 等。

第五阶段：现阶段，FPGA 朝着数模混合和异构架构的方向发展，真正成了“万能芯片”。在人工智能、云计算、物联网、通信和航空航天等方面都有着极其广泛的应用。



可编程逻辑器件
定义分类与简单
可编程逻辑器件
结构

2. 可编程逻辑器件的基本结构与分类

(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是由输入缓冲电路、与阵列、或阵列和输出缓冲电路组成，PLD 基础结构如图 1-1 所示。其中，与阵列和或阵列是核心，与阵列用来产生乘积项，或阵列用来产生乘积项之和形式的函数。输入缓冲电路可以产生输入变量的原变量和反变量，输出方式可以是组合输出、时序输出或者可编程输出，输出信号还可以通过内部通道反馈到输入端。



PLD 发展

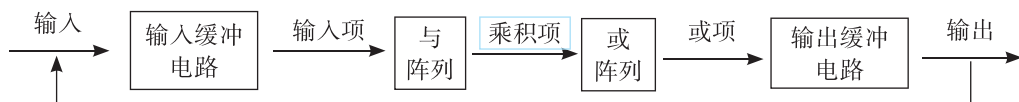


图 1-1 PLD 基本结构框图

(2) 可编程逻辑器件的分类。

可编程逻辑器件按集成度可分为简单 PLD 和复杂 PLD 两大类，PLD 发展结构如图 1-2 所示。

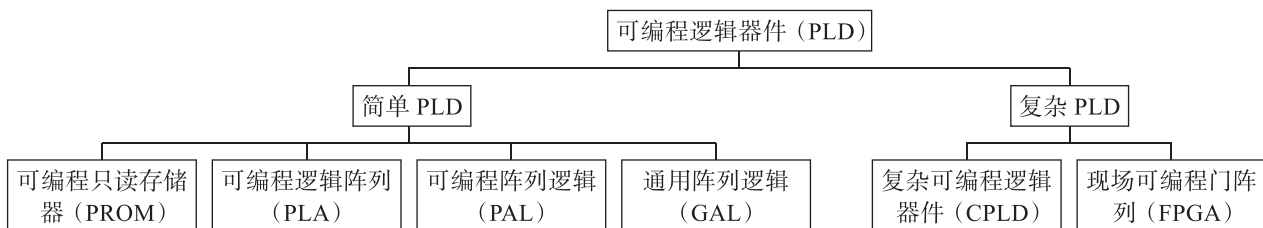


图 1-2 PLD 发展结构

① 简单可编程逻辑器件 (simple programmable logic device, SPLD)。

简单可编程逻辑器件是可编程器件的早期产品，它包括可编程只读存储器 (PROM)、可编程逻辑阵列 (PLA)、可编程阵列逻辑 (PAL) 和通用阵列逻辑 (GAL)。SPLD 的典型结构是由与门阵列和或门阵列组成，能够以“积之和”的形式实现布尔逻辑函数。因为任意一个组合逻辑都可以用与或表达式来描述，所以 SPLD 能够完成大量的组合逻辑功能，并且具有较高的运行速度和较好的性能。

最早的可编程逻辑器件是熔丝编程的只读存储器 PROM。当与阵列固定 (在结构图中常用 $\cdot$ 表示)，或阵列可编程时 (在结构图中常用 $\times$ 表示)，称之为可编程只读存储器 (PROM)，其结构如图 1-3 所示。这种可编程逻辑器件一般用作存储器，其输入为存储器的地址，输出为存储单元的内容。由于与阵列采用全译码器，随着输入的增多，阵列规模按输入的 2^n 倍增长。当输入的数目太大时，器件功耗增加，而较长的阵列开关时间也会导致其读存速度缓慢。但 PROM 价格低，易于编程，同时没有布局、布线问题，性能完全可以预测。此外，它不可擦除、不可重写的局限性也由于 EPROM (可擦除可编程只读存储器) 和 E^2 PROM (电可擦出可编程只读存储器) 的出现得到了解决，所以还是具有一定的应用价值。

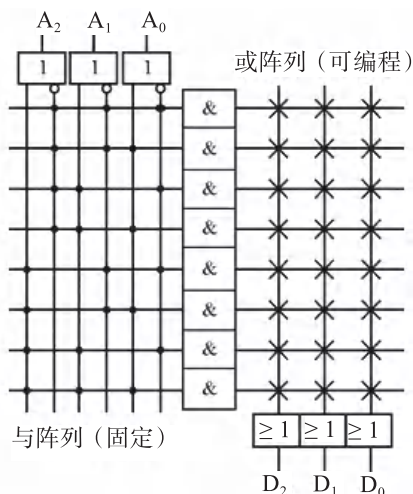


图 1-3 PROM 基本结构

当与阵列和或阵列都可编程时，称为可编程逻辑阵列 (PLA)，

其结构如图 1-4 所示。由于与阵列可编程，使得 PROM 中由于输入增加而导致规模增加的问题不复存在，从而有效地提高了芯片的利用率。PLA 用于含有复杂的随机逻辑置换的场合是较为理想的，PLA 器件既有现场可编程的，也有掩膜可编程的，虽然利用率高，但运行速度较慢，仅适用于小规模逻辑，同时相对高的价格也妨碍了它的广泛使用。

随后出现了可编程阵列逻辑（PAL）。当或阵列固定，与阵列可编程时，称之为可编程阵列逻辑，其结构如图 1-5 所示，其或门的输出可以通过触发器有选择地被设置为寄存状态。与阵列的可编程特性可以增加输入项，而固定的或阵列又使器件得到简化。在这种结构中，每个输出都是若干乘积项之和，其中乘积项的数目是固定的。PAL 的这种基本门阵列结构对于大多数逻辑函数是很有效的，因为大多数逻辑函数都可以方便地化简为若干个乘积项之和，即与或表达式，同时这种结构也提供了较高的性能和较快的运行速度，一度成为 PLD 发展史上的主流。PAL 有几种固定的输出结构，不同的输出结构对应不同的型号，设计时可以根据实际需要进行选择。PAL 器件是现场可编程的，它的实现工艺有反熔丝技术、EPROM 技术和 E²PROM 技术，但由于市场上类型和结构较多，反而不方便用户的使用和修改，目前已被淘汰。

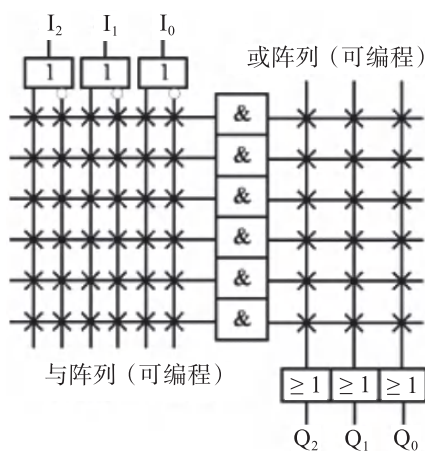


图 1-4 PLA 结构

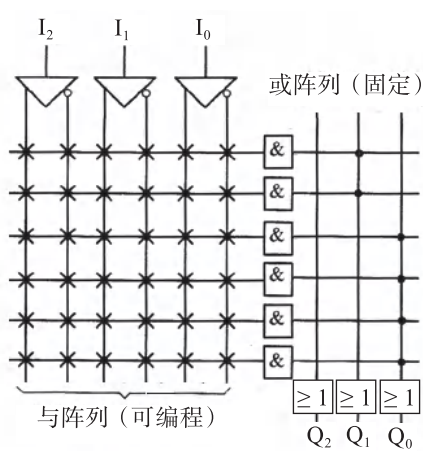


图 1-5 PAL 结构

在 PAL 的基础上，随后发展出了一种通用阵列逻辑（GAL）。它采用 E²PROM 工艺，彻底解决了熔丝型可编程器件的一次可编程问题，实现了电可擦除、电可改写、可重复编程和可设置加密的功能。GAL 的输出可由用户来定义，它的每个输出端都集成着一个可编程的输出逻辑宏单元（output logic macro cell, OLMC）。GAL16V8 的逻辑框图中，在 12-19 号管脚内就各有一个 OLMC，如图 1-6 所示。

GAL22V10 的 OLMC 内部结构如图 1-7 所示。从图 1-7 中可以看出，OLMC 中除了包含或门阵列和 D 触发器之外，还有两个多路选择器（MUX）。其中 4 选 1 MUX 用来选择输出方式和输出极性，2 选 1 MUX 用来选择反馈信号。这些选择器的状态都是可编程控制的，通过编程改变其连线可以使 OLMC 配置成多种不同的输出结构，这些输出结构完全包含了 PAL 的几种输出结构。普通 GAL 器件只有少数几种基本型号就可以取代数十种 PAL 器件，因而 GAL 是名副其实的通用可编程逻辑器件。由于 GAL 的设计具有很强的灵活性，所以至今仍被许多人使用。GAL 的缺点是规模较小，对于较为复杂的逻辑电路显得力不从心。

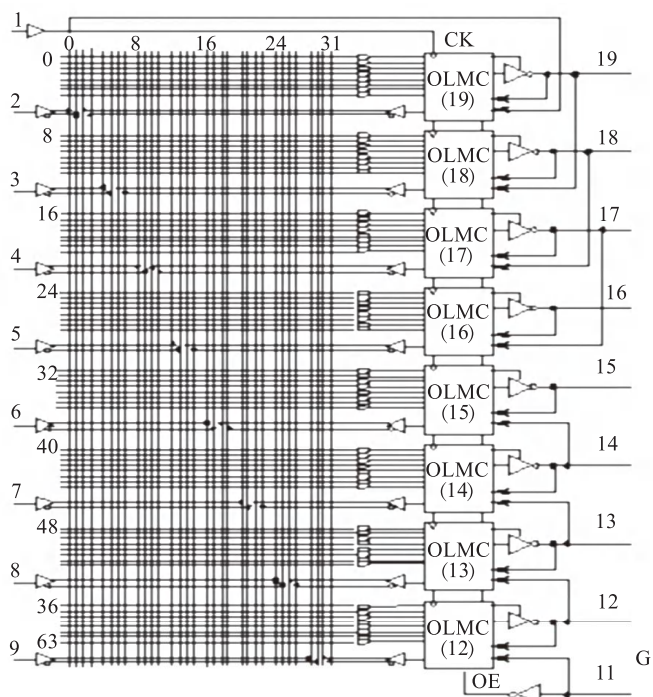


图 1-6 GAL16V8 逻辑框图

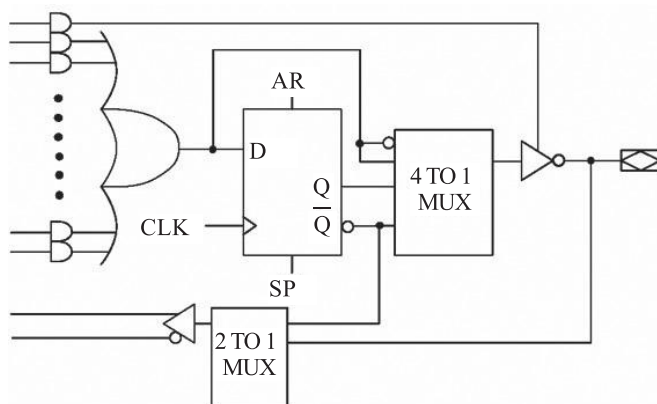


图 1-7 GAL22V10 的 OLMC 内部结构

② 复杂可编程逻辑器件 (complex programmable logic device, CPLD)。

CPLD 出现在 20 世纪 80 年代末期，其结构区别于早期的简单 PLD，最基本的特点在于：简单 PLD 是逻辑门编程，而复杂 PLD 为逻辑板块编程，即以逻辑宏单元为基础，加上内部的与或阵列和外围的输入 / 输出模块，不但实现了除简单逻辑控制之外的时序控制，又扩大了在整个系统中的应用范围和扩展性。

③ 现场可编程门阵列 (field programmable gate array, FPGA)。

FPGA 是一种可由用户自行定义配置的高密度专用集成电路，它将定制的超大规模集成电路 (very large scale integrated circuits, VLSI) 的单片逻辑集成优点和用户可编程逻辑器件的设计灵活、工艺实现方便、产品上市快捷等长处结合起来，采用逻辑单元阵列结构和静态随机存储工艺，设计灵活，集成度高，可重复编程，并可现场模拟调试验证。

3. PLD 相对于 MCU 的优势所在

(1) MCU 经常面临的难题。

MCU 逻辑行为上的普适性常常会让人们误以为 MCU 是无所不能的。在任何一个电子系统设计项目中, MCU 成了无可置疑的主角。但不深入考察 MCU 的优势和弱点, 任何项目都使用 MCU, 将严重影响系统设计的最佳选择和性价比的提高。

① 运行速度。

从理论上讲, MCU 几乎可以解决任何逻辑的实现, 但 MCU 是通过内部的 CPU 逐条执行软件指令来完成各种运算和逻辑功能的, 无论多么高的工作时钟频率和多么优秀的指令时序方式, 在排队式串行指令执行方式 (DSP 处理器也不能逃避这种工作方式) 面前, 其工作速度和效率必将大打折扣。因此, MCU 在实时仿真、高速工控或高速数据采样等许多领域尤显力不从心, 工作速度是 MCU 及其系统面临的巨大挑战。

② 复位。

复位工作方式是 MCU 的另一致命弱点, 任何 MCU 工作初始时都必须经历一个复位过程, 否则将无法进行正常工作。MCU 的复位必须满足一定的电平条件和时间条件 (长达毫秒级)。在工作电平有某种干扰性突变时, MCU 的复位设置将成为系统不可工作的重要因素。而且这种产生于复位的不完全性构成了系统不可靠工作的隐患, 其出现方式极为随机, 一般难以检测。一些系统在工作中出现的“假复位”和不可靠性复位带来的后果是十分严重的。尽管人们提出了种种改善复位的方法及可靠复位的电路, 市场上也有层出不穷的 MCU 复位监控专用器件, 但到目前为止, 复位的可靠性问题仍然未能得到根本解决。

③ 程序“跑飞”。

在强干扰或某种偶然因素下, 任何 MCU 的程序指针都极可能越出正常的程序流程“跑飞”, 这已是不争的事实。事实证明, 无论多么优秀的 MCU, 无论多么良好的抗干扰措施, 包括设置任何方式的内外硬件“看门狗”, 在受到强干扰特别是强电磁干扰的情况下, MCU 都无法保证仍能正常工作而不进入不可挽回的“死机”状态。尤其是当程序指针“跑飞”与复位不可靠因素互相交错时, 情况将变得尤为复杂。

(2) CPLD/FPGA 的优势。

基于 CPLD/FPGA 器件的开发应用可以从根本上解决 MCU 所遇到的问题。与 MCU 相比, CPLD/FPGA 在某些领域的优势是多方面的和根本性的。

① 高速性。

CPLD/FPGA 的时钟延迟仅为纳秒级, 结合其并行工作方式, 它在超高速应用领域和实时测控方面有非常广阔的应用前景。

② 高可靠性。

在高可靠应用领域, MCU 的缺憾为 CPLD/FPGA 的应用留下了很大的用武之地。除了不存在 MCU 所特有的复位不可靠与程序“跑飞”等固有缺陷外, CPLD/FPGA 的高可靠性还表现在几乎可将整个系统下载到同一芯片中, 从而大大缩小了芯片体积, 易于管理和屏蔽。

③ 编程方式。

采用 JTAG 在系统配置编程方式, 可对正在工作的系统上的 CPLD/FPGA 进行编程, 这在工控、智能仪器仪表、通信和军事上有特殊用途, 同时为系统的调试带来极大的方便。



FPGA 单片机和
嵌入式的区别

④ 标准化设计语言。

CPLD/FPGA 的设计开发工具，通过符合国际标准的硬件描述语言（如 VHDL 或 Verilog HDL）来进行电子系统设计和产品开发。由于开发工具的通用性、设计语言的标准化以及设计过程几乎与所用的 CPLD/FPGA 器件的硬件结构没有关系，所以设计的各类逻辑功能块软件都有很好的兼容性和可移植性。

在电子应用系统设计中，设计者应该充分了解系统的需求，认识 MCU 与 CPLD/FPGA 各自的优势，利用 MCU 与 CPLD/FPGA 在功能和性能上的互补性，在构成系统的功能模块中合理选择 MCU 或 CPLD/FPGA，充分发挥 MCU 与 CPLD/FPGA 各自的长处，使应用系统实现最佳的技术配合。

(二) FPGA 和 CPLD

早期的 PLD 器件的一个共同特点是可以实现速度特性较好的逻辑功能，但其过于简单的结构也使它们只能实现规模较小的电路。为了弥补这一缺陷，20 世纪 80 年代中期，Altera 和 Xilinx 分别推出了类似于 PAL 结构的扩展型 CPLD 以及标准门阵列类似的 FPGA，它们都具有体系结构完善、逻辑单元灵活、集成度高和适用范围宽等特点。这两种器件兼容了 PLD 的通用门阵列的优点，可实现较大规模的电路，编程也很灵活。与门阵列其他 ASIC 相比，它们又具有设计开发周期短、设计制造成本低、开发工具先进、标准产品无需测试、质量稳定和可实时在线检验等优点，因此被广泛应用于产品的原型设计和产品设计之中。几乎所有应用门阵列、PLD 和中小规模通用数字集成电路的场合均可以应用 FPGA 和 CPLD 器件。



复杂可编程逻辑器件结构

不过，不同厂家对 PLD 的叫法不尽相同，Xilinx 把基于查找表技术和 SRAM 工艺，需要外挂配置用的 EEPROM 的 PLD 叫做 FPGA，把基于乘积项技术 Flash 工艺（类似 EEPROM 工艺）的 PLD 叫做 CPLD。Altera 把自己的 PLD 产品的 MAX 系列（乘积项技术，EEPROM 工艺）和 FLEX 系列（查找表技术，SRAM 工艺）都叫做 CPLD。由于 FLEX 系列也是采用 SRAM 工艺，基于查找表技术，需要外挂配置用的 EPROM，并且它的用法和 Xilinx 的 FPGA 相同，所以很多人把 Altera 的 FLEX 系列产品也叫 FPGA。CPLD 是复杂可编程逻辑器件的简称，FPGA 是现场可编程门阵列的简称，两者的功能基本相同，只是实现原理略有不同。对用户而言，CPLD 与 FPGA 的内部结构稍有不同，但用法一样，所以大多数情况下不加以区分，把它们统称为可编程逻辑器件或 CPLD/FPGA。

PLD 是电子设计领域中最具活力和发展前途的一项技术，它的影响丝毫不亚于 20 世纪 70 年代单片机的发明和使用。PLD 能做什么呢？可以毫不夸张地说，PLD 能完成任何数字器件的功能，上至高性能的 CPU，下至简单的 74 电路，都可以用 PLD 来实现。PLD 如同一张白纸或者一堆积木，工程师可以通过传统的原理图输入法，或是硬件描述语言自由地设计一个数字系统。通过软件仿真，设计者可以事先验证设计的正确性。在 PCB 完成以后，还可以利用 PLD 的在线修改能力，随时修改设计而不必改动硬件电路。使用 PLD 来开发数字电路，可以大大缩短设计时间，减少 PCB 面积，提高系统的可靠性。PLD 的这些优点使得 PLD 技术在 20 世纪 90 年代以后得到飞速的发展，同时也推动了 EDA 软件和硬件描述语言（HDL）的进步。如何使用 PLD 呢？其实 PLD 的使用很简单，学习 PLD 要比学习单片机容易得多，有数字电路基础，会使用计算机，就可以进行 PLD 的开发。

1. 基于乘积项 (Product-Term) 的 PLD 结构

采用这种结构的 PLD 芯片有 Altera 的 MAX7000、MAX3000 系列（EEPROM 工艺），Xilinx 的 XC9500 系列（Flash 工艺）和 Lattice, Cypress 的大部分产品（EEPROM 工艺）。先看一下这种 PLD 的总体结构（以 MAX7000 为例，其他型号的结构与此类似），基于乘积项的 PLD 内部结构如图 1-8 所示。

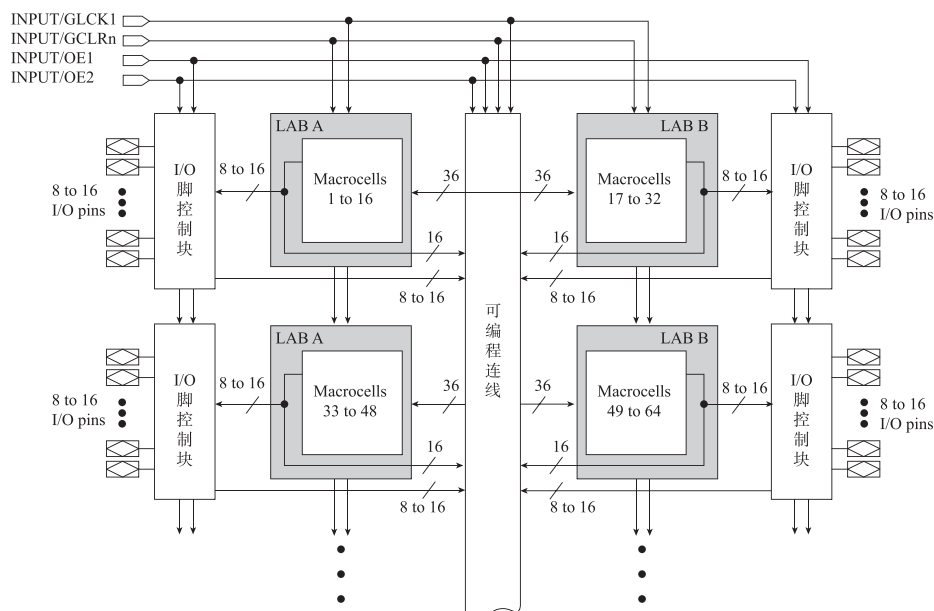


图 1-8 基于乘积项的 PLD 内部结构

这种 PLD 可分为三部分：宏单元 (macrocell)，可编程连线阵列 (programmable interconnect array, PIA) 和 I/O 控制块 (I/O control blocks)。宏单元是 PLD 的基本结构，由它来实现基本的逻辑功能。图 1-8 中深色部分是多个宏单元的集合 (因为宏单元较多，没有一一画出)。可编程连线阵列负责信号传递，连接所有的宏单元。I/O 脚控制块负责输入 / 输出的电气特性控制，比如可以设定集电极开路输出、三态输出等。图 1-8 中左上角的 INPUT/GCLK1、INPUT/GCLRn、INPUT/OE1 和 INPUT/OE2 是全局时钟、清零和输出使能信号，这几个信号有专用连线与 PLD 中的每个宏单元相连，信号到每个宏单元的延时相同并且延时最短。

宏单元结构如图 1-9 所示，左侧是乘积项逻辑阵列，实际就是一个与阵列，每一个交叉点都是一个可编程熔丝，如果导通就是实现与逻辑。后面的乘积项选择矩阵是一个或阵列，两者一起完成组合逻辑。图 1-9 右侧是一个可编程 D 触发器，它的时钟和清零输入都可以编程选择，可以使用专用的全局清零和全局时钟，也可以使用内部逻辑 (乘积项阵列) 产生的清零和时钟。如果不需要触发器，也可以将 D 触发器旁路，信号直接输给 PIA 或输出到 I/O 脚。

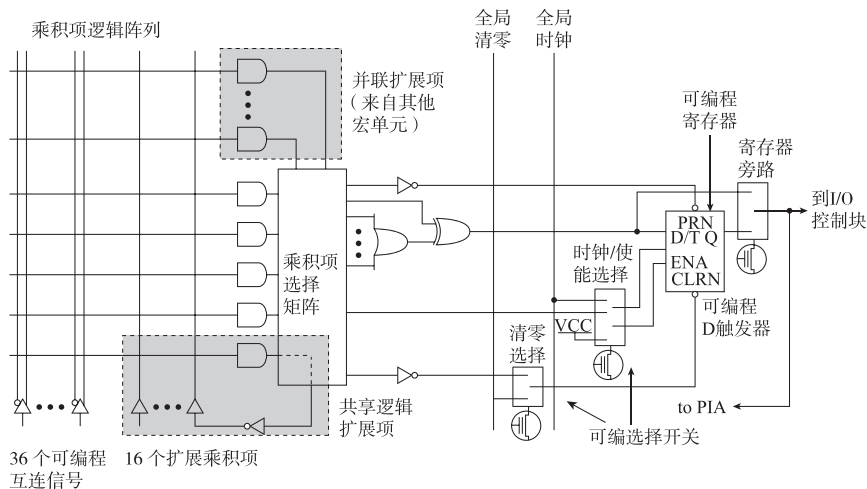


图 1-9 宏单元结构

2. 乘积项结构 PLD 的逻辑实现原理

下面以一个简单的电路为例，具体说明 PLD 是如何利用以上结构实现逻辑的，电路如图 1-10 所示。

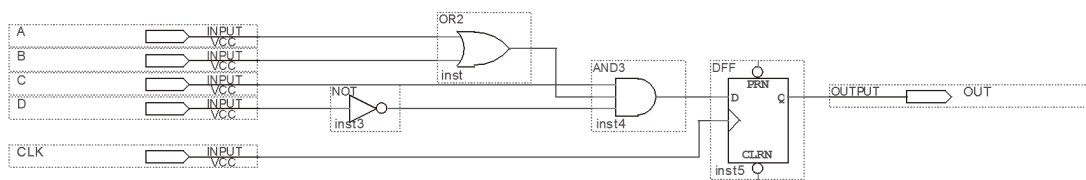


图 1-10 电路 1

假设组合逻辑的输出（AND3 的输出）为 f ，则 $f = (A+B) * C * (\bar{D}) = A * C * \bar{D} + B * C * \bar{D}$ 。PLD 将以如图 1-11 所示的方式来实现组合逻辑 f 。

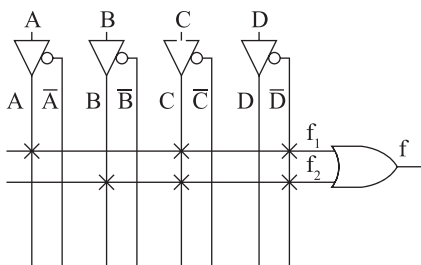


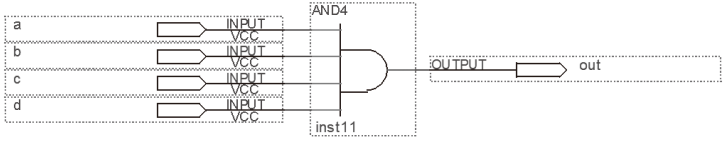
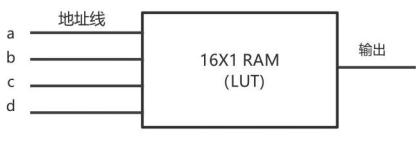
图 1-11 电路图 2

A 、 B 、 C 、 D 由 PLD 芯片的管脚输入后进入可编程连线阵列（PIA），在内部会产生 A 、 A 反、 B 、 B 反、 C 、 C 反、 D 、 D 反 8 个输出。图 1-11 中每一个叉相连（可编程熔丝导通），所以得到 $f = f_1 + f_2 = A * C * \bar{D} + B * C * \bar{D}$ ，这样组合逻辑就实现了。图 1-10 中 D 触发器的实现比较简单，直接利用宏单元中的可编程 D 触发器来实现。时钟信号 CLK 由 I/O 管脚输入后进入芯片内部的全局时钟专用通道，直接连接到可编程 D 触发器的时钟端。可编程 D 触发器的输出与 I/O 脚相连，把结果输出到芯片管脚。这样 PLD 就完成了图 1-10 所示电路的功能。以上这些步骤都是由软件自动完成，不需要人工干预。图 1-10 的电路是一个很简单的例子，只需要一个宏单元就可以完成，但对于一个复杂的电路，一个宏单元是不能完成的，这时就需要通过并联扩展项和共享扩展项将多个宏单元相连，宏单元的输出也可以连接到可编程连线阵列，再作为另一个宏单元的输入。这样 PLD 就可以实现更复杂的逻辑。这种基于乘积项的 PLD 基本都是由 EEPROM 和 FLASH 工艺制造的，一上电就可以工作，无需其他芯片配合。

3. 查找表 (Look-Up-Table) 的原理与结构

采用查找表结构的 PLD 芯片也可以称之为 FPGA，如 Altera 的 ACEX、APEX 系列，Xilinx 的 Spartan、Virtex 系列等。查找表（Look-Up-Table）简称 LUT，LUT 本质上就是一个 RAM（random access memory，随机存取存储器）。目前 FPGA 中多使用 4 输入的 LUT，所以每一个 LUT 可以看成是一个有 4 位地址线的 16×1 的 RAM。当用户通过原理图或 HDL 语言描述了一个逻辑电路以后，CPLD/FPGA 开发软件会自动计算逻辑电路所有可能的结果，并把结果事先写入 RAM。这样，每输入一个信号进行逻辑运算就等于输入一个地址进行查表，找出地址对应的内容，然后输出即可。表 1-1 是一个 4 输入与门的例子。

表 1-1 4 输入与门

实际逻辑电路		LUT 实现方式	
			
0000	0	0000	0
0001	0	0001	0
.....	0		0
1111	1	1111	1

4. 基于查找表 (LUT) 的 FPGA 结构

Xilinx Spartan-II 芯片的主要结构如图 1-12 所示。

Spartan-II 芯片主要包括 CLB (configurable logic block, 可配置逻辑块)、I/O 块、RAM 块和可编程连线 (未表示出)。在 Spartan-II 中, 一个 CLB 包括两个 Slices, 每个 Slices 包括两个 LUT、两个触发器和相关逻辑。Slices 可以看成是 Spartan-II 实现逻辑的最基本结构 (Xilinx 的其他系列, 如 SpartanXL、Virtex 的结构与此稍有不同, 具体结构请参阅数据手册), Slices 结构如图 1-13 所示。

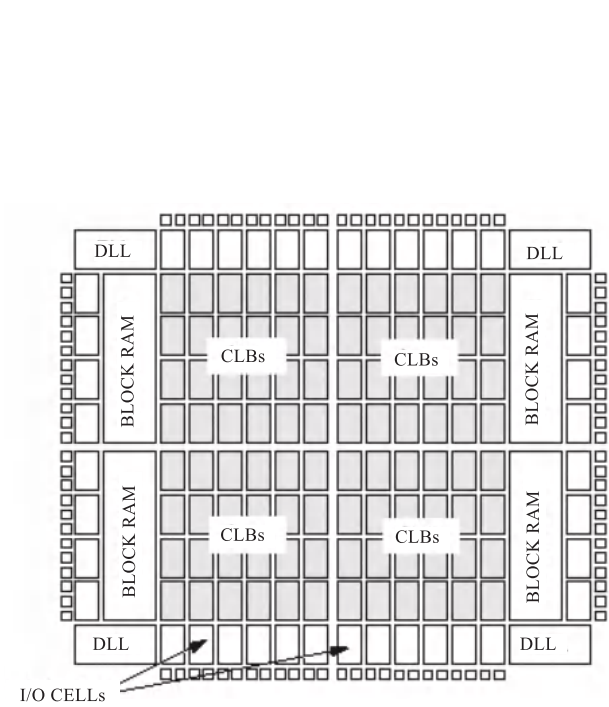


图 1-12 Xilinx Spartan-II 芯片的主要结构

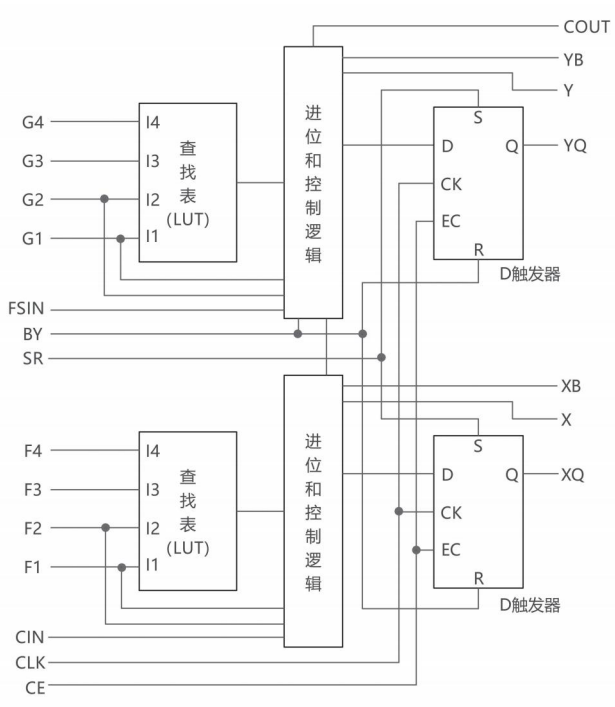


图 1-13 Slices 结构

Altera 的 FLEX/ACEX 芯片的内部结构如图 1-14 所示。

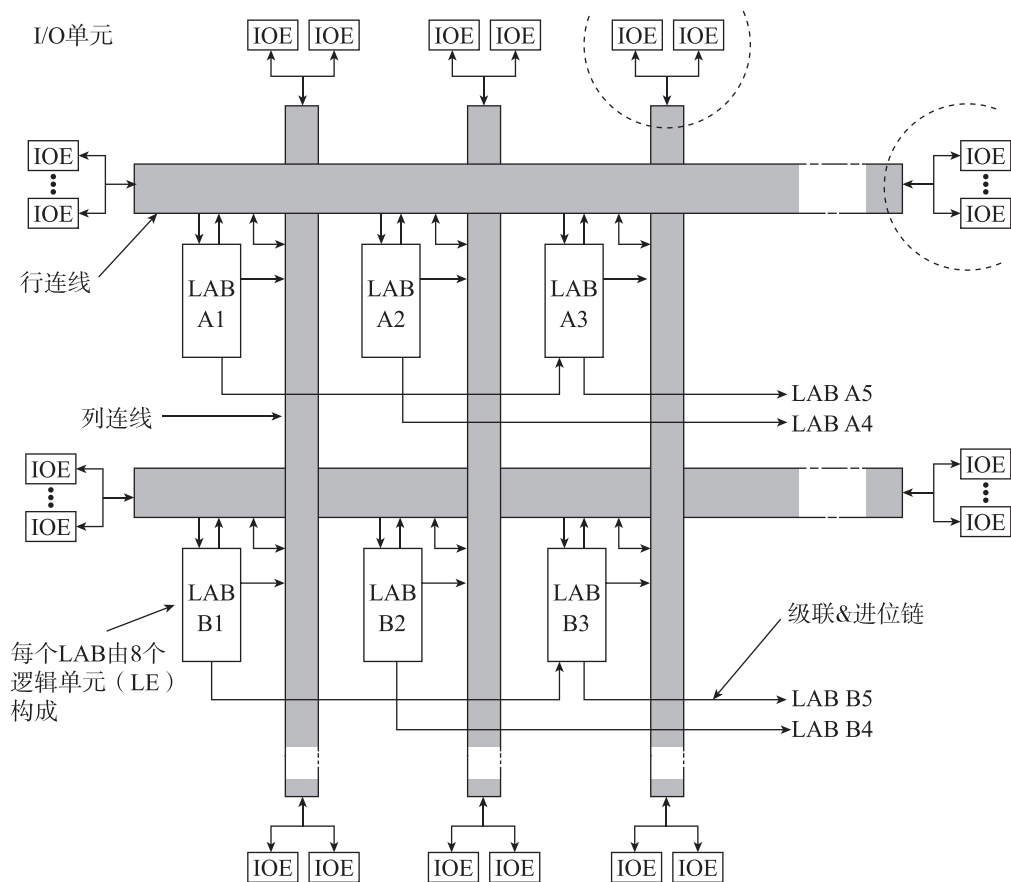


图 1-14 Altera 的 FLEX/ACEX 芯片的内部结构

逻辑单元 (LE) 内部结构如图 1-15 所示。

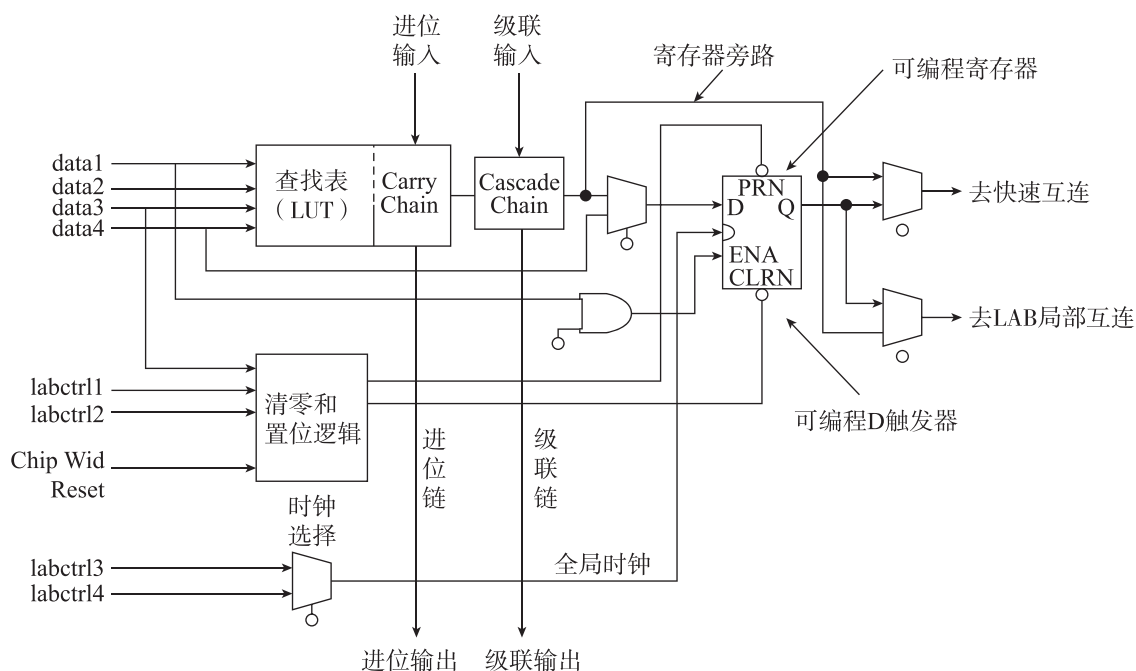


图 1-15 逻辑单元 (LE) 内部结构

FLEX/ACEX 的结构主要包括 LAB、I/O 块、RAM 块（未表示出）和可编程行 / 列连线。在 FLEX/ACEX 中，一个 LAB 包括 8 个逻辑单元（LE），每个 LE 包括一个 LUT、一个触发器和相关逻辑。LE 是 FLEX/ACEX 芯片实现逻辑的最基本结构（Altera 其他系列，如 APEX 的结构与此基本相同，具体结构请参阅数据手册）。

5. 查找表结构的 FPGA 逻辑实现原理

还是以图 1-10 所示的逻辑电路为例进行介绍。

A、B、C、D 由 FPGA 芯片的管脚输入后进入可编程连线，然后作为地址线连到 LUT，LUT 中已经事先写入了所有可能的逻辑结果，通过地址查找到相应的数据然后输出，这样就实现了组合逻辑。该电路中 D 触发器直接利用 LUT 后面的 D 触发器来实现。时钟信号 CLK 由 I/O 脚输入后进入芯片内部的时钟专用通道，直接连接到触发器的时钟端。触发器的输出与 I/O 脚相连，把结果输出到芯片管脚。这样 PLD 就完成了如图 1-10 所示电路的功能。（以上这些步骤都是由软件自动完成，不需要人工干预）

这个电路是一个很简单的例子，只需要一个 LUT 加上一个触发器就可以完成。对于一个 LUT 无法完成的电路，就需要通过进位逻辑将多个单元相连，这样 FPGA 就可以实现复杂逻辑。由于 LUT 主要适合 SRAM 工艺生产，所以目前大部分 FPGA 都是基于 SRAM 工艺的。而 SRAM 工艺的芯片在掉电后信息就会丢失，一定需要外加一片专用配置芯片，在上电的时候，由这个专用配置芯片把数据加载到 FPGA 中，然后 FPGA 就可以正常工作，由于配置时间很短，不会影响系统正常工作。也有少数 FPGA 采用反熔丝或 Flash 工艺，对于这种 FPGA，就不需要外加专用的配置芯片。

6. 其他类型的 FPGA 和 PLD

随着技术的发展，在 2004 年以后，一些厂家推出了一些新的 PLD 和 FPGA，这些产品模糊了 PLD 和 FPGA 的区别。如 Altera 的 MAXII 系列 PLD，这是一种基于 FPGA（LUT）结构，集成配置芯片的 PLD，它在本质上就是一种在内部集成了配置芯片的 FPGA。由于配置时间极短，它在上电后就可以工作，所以对用户来说，感觉不到配置过程，可以像传统的 PLD 一样使用它，加上其容量和传统 PLD 类似，所以 Altera 也把它归为 PLD。还有像 Lattice 的 XP 的系列 FPGA 也是使用了同样的原理，它将外部配置芯片集成到内部，在使用方法上与 PLD 类似，但是因为容量大，性能和传统 FPGA 相同，也是 LUT 架构，所以 Lattice 仍把它归为 FPGA。

（三）可编程逻辑器件的应用

CPLD、FPGA 是近几年集成电路中发展最快的产品。由于 PLD 性能的提高以及设计人员自身能力的提高，可编程逻辑器件供应商将进一步扩大可编程芯片的领地，将复杂的专用芯片推向高端和超复杂应用。可编程逻辑器件依然是集成电路中最具活力和前途的产业。

1. PLD 在专用集成电路设计中的应用

传统的专用集成电路（ASIC）设计通常采用全定制电路设计方法或半定制电路设计方法进行检验，当不满足要求时，就需要重新设计再进行验证，这就使得设计开发费用较高，研发周期较长，因而使得设计出来的产品性价比不高，缺乏市场竞争力，进而无法满足市场的需求。随着微电子技术的高速发展，特别是 ASIC 的市场需求，迫使产品越来越需要降低系统体积和成本，提高系统可靠性，缩短研发周期。在这种形势下，一种共同性强、用量大、成本也不高的且具有一定连线的结构和已封装好的全功能的标准电路诞生了，这就是 PLD，它使用某种编程技术自己“烧制”，使内部电路结构实现再连接，PLD 的引入就形成了半定制电路设计方法的可编程 ASIC。通信和网络产品市场是 PLD 最大

的应用市场。因 PLD 器件工艺复杂且产品利润高，所以一直以来 PLD 被认为是只能应用于高档产品。但是随着半导体工艺的发展，PLD 芯片的成本已越来越低，甚至已经可以和 ASIC 芯片和标准集成电路相互竞争，这使得 PLD 的应用领域不断扩大，这反过来进一步加速了 PLD 产品的发展。

2. 基于 EDA 工具的 PLD 应用

美国 Altera 公司开发的 EDA 软件具有功能强大、易学易懂等特点，Maxplus II 和 Quartus II 分别作为其第三代和第四代开发系统的代表，应用广泛。Xilinx 公司的开发工具功能也十分强大，应用广泛，其中 Foundation 系列及 ISE (integrated synthesis environment) 都曾是非常优秀的开发系统。Vivado 是 Xilinx 公司于 2012 年推出的新一代集成设计环境，Vivado 的推出是为了提高设计者的效率，它能显著增加 Xilinx 的 28 nm 工艺的可编程逻辑器件的设计、综合与实现效率。CPLD/FPGA 的设计开发采用这些功能强大的 EDA 工具，通过符合国际标准的硬件描述语言（如 VHDL 或 Verilog HDL）来进行电子系统设计和产品开发。开发工具的通用性、设计语言的标准化以及设计过程几乎与所用的 CPLD/FPGA 器件的硬件结构没有关系等特点，使得设计成功的逻辑功能软件有很好的兼容性和可移植性、开发周期短、易学易用并且开发便捷。可编程逻辑器件最初主要用于通信领域，其优势在于缩短开发生产周期，现场灵活性好，适于少量生产，但价格稍微昂贵，随着微细化的进步，芯片面积缩小，价格迅速下降，市场发展加快。Actel 公司对 CPLD/FPGA 产品的发展蓝图显示，在未来的几年内，中国的可编程逻辑器件增幅最大的行业是航空航天、通信和消费电子领域，特别是消费电子中的游戏机、机顶盒、数字相机和 PDA 等产品。目前我国已成为世界最主要的 PLD 制造基地。

3. FPGA 和 CPLD 开发应用选择

尽管 FPGA 和 CPLD 都是可编程 ASIC 器件，都具有现场可编程特性，都支持边界扫描技术，但由于 CPLD 和 FPGA 结构上的差异，决定了 CPLD 和 FPGA 在性能上各有特点。

(1) FPGA 是细粒结构，这意味着每个单元间存在细粒延迟。如果将少量逻辑紧密排列在一起，FPGA 的速度相当快。然而，随着设计密度的增加，信号不得不通过许多开关，路由延迟也快速增加，从而削弱了整体性能。FPGA 是“寄存器丰富”型（即其寄存器与逻辑门的比例高），而 CPLD 正好相反，它是粗粒结构，是“逻辑丰富”型，这意味着进出器件的路径经过较少的开关，相应的延迟也小。因此，与等效的 FPGA 相比，CPLD 可工作在更高的频率，具有更好的性能。

(2) CPLD 的连续式布线结构决定了它的时序延迟是均匀的和可预测的，而 FPGA 的分段式布线结构决定了其延迟的不可预测性。

(3) CPLD 使用起来比 FPGA 更方便。CPLD 的编程采用 E2PROM 或 Fast Flash 技术，无需外部存储器芯片，使用简单。而 FPGA 的编程信息需存放在外部存储器上，使用方法复杂。

(4) FPGA 的集成度比 CPLD 高，具有更复杂的布线结构和逻辑实现。

(5) 一般情况下，CPLD 的功耗要比 FPGA 大，并且集成度越高越明显。

(6) CPLD 的速度比 FPGA 快，并且具有较大的时间可预测性。这是由于 FPGA 是门级编程，并且 CLB 之间采用分布式互联，而 CPLD 是逻辑块级编程，并且其逻辑块之间的互联是集总式的。

(7) CPLD 更适合完成各种算法和组合逻辑，FPGA 更适合于完成时序逻辑。换句话说，FPGA 更适合于触发器丰富的结构，而 CPLD 更适合于触发器有限而乘积项丰富的结构。

(8) FPGA 在编程上比 CPLD 具有更大的灵活性。CPLD 通过修改具有固定内连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；FPGA 可在逻辑门下编程，而 CPLD 是在逻辑块下编程的。

(9) 在编程方式上，CPLD 主要是基于 E²PROM 或 Flash 存储器编程，编程次数可达 1 万次，优

点是系统断电时编程信息也不丢失。FPGA 大部分是基于 SRAM 编程的，编程信息在系统断电时丢失，每次上电时，需从器件外部将编程数据重新写入 SRAM 中。FPGA 的优点是可以编程任意次，可在工作中快速编程，从而实现板级和系统级的动态配置。

(10) CPLD 保密性好，FPGA 保密性差（因为其编程信息需存放在外部存储器上）。

总之，CPLD 与 FPGA 由于各自的特点与优势，使得二者在可编程逻辑器件技术的竞争中并驾齐驱，成为两支领导可编程器件技术发展的主要力量。在选择使用 CPLD 还是 FPGA 时，可根据不同的技术要求和设计环境做出最佳选择。

三、面向 CPLD/FPGA 的 EDA 设计流程

基于 EDA 工具的 CPLD/FPGA 开发流程如图 1-16 所示。

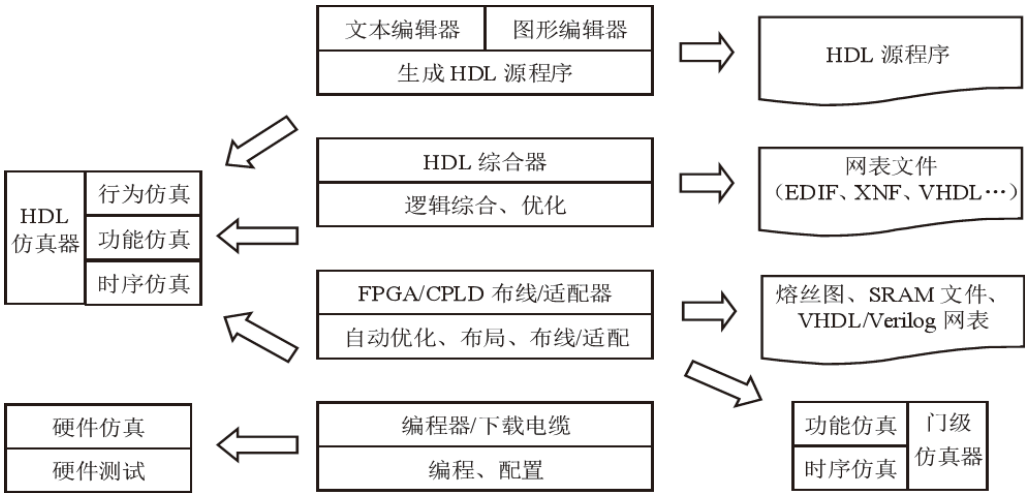


图 1-16 基于 EDA 工具的 CPLD/FPGA 开发流程

第一，需要进行“源程序的编辑和编译”——用一定的逻辑表达手段将设计表达出来；第二，要进行“逻辑综合”——将用一定的逻辑表达手段表达出来的设计，经过一系列的操作分解成一系列的基本逻辑电路及对应关系（电路分解）；第三，要进行“目标器件的布线 / 适配”——在选定的目标器件中建立这些基本逻辑电路及对应关系（逻辑实现）；第四，“目标器件的编程 / 下载”——将前面的软件设计经过编程变成具体的设计系统（物理实现）；最后，要进行“硬件仿真 / 硬件测试”——验证所设计的系统是否符合设计要求。同时在设计过程中要进行有关“仿真”——模拟有关设计结果与设计构想是否相符。

（一）源程序的编辑与编译

利用 EDA 工具进行一项工程设计，首先需要利用 EDA 工具的文本编辑器或图形编辑器将它用文本方式或图形方式表达出来，进行排错编译，变成硬件描述语言文件格式，为进一步的编译综合做准备。

常用的源程序输入方式有以下三种。

1. 原理图输入方式

利用 EDA 工具提供的图形编辑器以原理图的方式进行输入。原理图输入方式比较容易掌握，输入

方式直观便捷，所画的电路原理图（请注意，这种原理图与利用 Protel 画的原理图有本质的区别）与传统的器件连接方式完全一样，很容易被人接受，而且编辑器中有很多现成的单元器件可以利用，用户也可以根据需要设计元件。然而原理图输入法的优点也是它的缺点：

（1）随着设计规模增大，设计的易读性迅速下降，对于图中密密麻麻的电路连线，极难弄清电路的实际功能；

（2）一旦完成，电路结构的改变将十分困难，因而几乎没有可再利用的设计模块；

（3）移植困难、入档困难、交流困难、设计交付困难。因为不可能存在一个标准化的原理图编辑器。

2. 状态图输入方式

以图形的方式表示状态图进行输入。当填好时钟信号名、状态转换条件和状态机类型等要素后，就可以自动生成 Verilog 或 VHDL 程序。这种设计方式简化了状态机的设计，比较流行。

3. 硬件描述语言软件程序的文本方式

硬件描述语言软件程序的文本方式是最一般化、最具普遍性的输入方式，任何支持 Verilog 或 VHDL 的 EDA 工具都支持文本方式的编辑与编译。

（二）逻辑综合和优化

要想把 Verilog 或 VHDL 的软件设计与硬件的可实现性挂钩，需要利用 EDA 软件系统的综合器进行逻辑综合。所谓逻辑综合，就是将电路的高级语言描述（如 HDL、原理图或状态图的描述）转换成低级的可与 CPLD/FPGA 或构成 ASIC 的门阵列基本结构相映射的网表文件。逻辑映射的过程，就是针对给定硬件结构组件，将电路的高级描述进行编译、优化、转换和综合，最终获得门级电路甚至更低层的电路描述文件。而网表文件就是按照某种规定描述电路的基本组成及如何相互连接的关系的文件。由于 Verilog/VHDL 仿真器的行为仿真功能是面向高层次的系统仿真，只能对 Verilog/VHDL 的关系描述做可行性的评估测试，不针对任何硬件系统，因此基于这一仿真层次的许多 Verilog/VHDL 语句不能被综合器所接受。这就是说，这类语句的描述无法在硬件系统中实现（至少是现阶段），这时，综合器不支持的语句在综合中将被忽略掉。综合器对 Verilog/VHDL 源文件的综合是针对某一 PLD 供应商的产品系列，因此，综合后的结果是可以被硬件系统所接受的，具有硬件可实现性。

（三）目标器件的布线与适配

所谓逻辑适配，就是将综合器产生的网表文件针对某一具体的目标器件进行逻辑映射操作，其中包括底层器件配置、逻辑分割、逻辑优化、布线与操作等，把它们配置到指定的目标器件中，产生最终的下载文件，如 JEDEC 格式的文件。

适配所选定的目标器件（如 CPLD/FPGA 芯片）必须属于原综合器指定的目标器件系列。对于一般的可编程模拟器件所对应的 EDA 软件来说，一般仅需包含一个适配器就可以了，如 Lattice 公司的 PAC-DESIGNER。通常 EDA 软件中的综合器可由专业的第三方 EDA 公司提供，而适配器则需由 CPLD/FPGA 供应商自己提供，因为适配器的适配对象直接与器件结构相对应。

（四）目标器件的编程与下载

如果编译、综合、布线器或适配和行为仿真、功能仿真、时序仿真等过程都没有发现问题，即满足了原设计的要求，此时可以将由 CPLD/FPGA 布线或适配器产生的配置和下载文件通过编程器或下载电缆载入目标芯片 CPLD 或 FPGA 中。

（五）设计过程中的有关仿真

设计过程中的仿真有行为仿真、功能仿真和时序仿真三种。

1. 行为仿真

行为仿真，就是将 Verilog/VHDL 设计源程序直接送到仿真器中所进行的仿真。

该仿真只是根据 Verilog/VHDL 的语义进行的，与具体电路没有关系。在这个阶段的仿真中，可以充分发挥 Verilog/VHDL 中适用于仿真控制的语句及其有关的预定义函数和库文件。

2. 功能仿真

功能仿真，就是将综合后的 Verilog/VHDL 网表文件再送到仿真器中所进行的仿真。

这时的仿真仅对 Verilog/VHDL 描述的逻辑功能进行测试模拟，以了解其实现的功能是否满足原设计的要求，仿真过程不涉及具体器件的硬件特性，如延时特性。该仿真的结果与门级仿真器所做的功能仿真结果基本一致。综合之后的 Verilog/VHDL 网表文件采用 Verilog/VHDL 语法，首先描述了最基本的门电路，然后将这些门电路用例化语句连接起来。描述的电路与生成的 EDIF/XNF 等网表文件一致。

3. 时序仿真

时序仿真，就是将布线器或适配器所产生的 Verilog/VHDL 网表文件发送到仿真器中所进行的仿真。

该仿真已将器件特性考虑进去，因此可以得到精确的时序仿真结果。布线器或适配处理后生成的 Verilog/VHDL 网表文件包含了较为精确的延时信息，网表文件中描述的电路结构与布线后、适配后的结果是一致的。

需要注意的是，EDA 工程设计流程图中有两个仿真器，一个是 Verilog/VHDL 仿真器，另一个是门级仿真器，它们都能进行功能仿真和时序仿真。所不同的是仿真用的文件格式不同，即网表文件不同。这里所谓的网表（Netlist），特指电路网络。网表文件描述了一个电路网络。目前流行多种网表文件格式，其中最通用的是 EDIF 格式的网表文件，Xilinx 公司的 XNF 网表文件格式也很流行，不过一般只在使用 Xilinx 公司的 CPLD/FPGA 时才会用到 XNF 格式。Verilog/VHDL 文件格式也可以用来描述电路网络，即采用 Verilog/VHDL 语法描述各级电路互连，称之为 Verilog/VHDL 网表。

（六）硬件仿真和硬件测试

所谓硬件仿真，就是在 ASIC 设计中利用 FPGA 对系统的设计进行功能检测，功能检测通过后再将其 Verilog/VHDL 设计以 ASIC 形式实现，这一过程称之为硬件仿真。

所谓硬件测试，就是把 FPGA 或 CPLD 直接用于应用系统的设计中，将下载文件下载到 FPGA 后，对系统的设计进行的功能检测，这一过程称之为硬件测试。

硬件仿真和硬件测试的目的是在更真实的环境中检测 Verilog/VHDL 设计的运行情况，特别是对于一些设计不是十分规范、语义上含有一定歧义的程序。一般的仿真器包含 Verilog/VHDL 行为仿真器和 Verilog/VHDL 功能仿真器，它们对于同一 Verilog/VHDL 设计的“理解”（即仿真模型的产生）与 Verilog/VHDL 综合器的“理解”（即综合模型的产生）常常是不一致的。此外，由于目标器件功能的可行性约束，综合器对于设计的“理解”常在一个有限范围内选择，而 Verilog/VHDL 仿真器的“理解”是纯软件行为，其“理解”的选择范围要宽得多，这种“理解”的偏差势必导致仿真结果与综合后实现的硬件电路在功能上不一致。当然，还有许多其他的因素也会产生这种不一致，由此可见，Verilog/VHDL 设计的硬件仿真和硬件测试是十分必要的。

四、FPGA 生产厂商简介

（一）全球 FPGA 主要厂商

目前全球 FPGA 市场常年被美国四大巨头垄断，全球 FPGA 主要生产商目前主要有美国硅谷这四大公司：Xilinx、Altera、Lattice、Microsemi。Xilinx 是全球领先的可编程逻辑完整解决方案的供应商，也是目前排名第一的 FPGA 解决方案提供商。2022 年 2 月，Xilinx 被 AMD 以 350 亿美元收购，AMD 成了全球唯一能够同时提供 CPU、GPU 和 FPGA 三种产品的厂商。Altera 是在业界与 Xilinx 齐名的 FPGA 供应商，2015 年被英特尔收购了，因此现在也称为英特尔的 FPGA。Xilinx 与 Altera 这两家公司共占近 90% 的市场份额。Lattice（莱迪思）是著名的可编程逻辑解决方案供应商，实力仅次于赛灵思和阿尔特拉。Microsemi（美高森美）专注于美国军工和航空领域，产品以反熔丝结构 FPGA 和基于 Flash 的 FPGA 为主，具有抗辐射和可靠性高的优势，2018 年 Microsemi 被 Microchip（微芯科技）并购。

数字集成电路本身在不断地进行更新换代，市场需求越来越多样化。为了满足这样的需求，必然不能单单只通过简单的方案组合、设计架构来实现。

为了适应时代发展和满足市场需求，技术、芯片、IP、软件开发、人才培养以及健康的生态等多个方面都是 FPGA 不断进步、不断发展的方向。这也意味着各大厂商唯有不断追求创新、弥补短板、发挥优势，才能有更好的产品 and 市场。

（二）国内主要 FPGA 企业

FPGA 芯片的市场前景广阔，但全球 FPGA 市场多年来一直被 Xilinx、Altera 等四大巨头垄断。近年来，国内 FPGA 产业已经取得了很大的发展，涌现了一大批优秀的国产 FPGA 企业，正在不断缩小与国际巨头的差距。国内十大 FPGA 厂商包括紫光同创、安路科技、高云半导体、复旦微电子等。



1. 紫光同创

紫光同创是紫光集团旗下紫光国微的子公司，成立于 2013 年，具备大规模 FPGA 全流程开发设计能力，布局覆盖高中低端 FPGA 产品，与安路科技和高云半导体曾被称为“国内 FPGA 三驾马车”。在 2015 年，紫光同创成功推出国内第一款千万门级的全自主知识产权高性能 FPGA 芯片 Titan 系列；2019 年 10 月，紫光同创高性能 FPGA 芯片荣获“中国芯”年度重量级奖项优秀市场表现产品奖。

2. 安路科技

安路科技成立于 2011 年，隶属于上海安路信息科技股份有限公司，专注于 FPGA 芯片设计领域，以成为国产 FPGA 芯片的产业创新者和国际市场 FPGA 芯片的重要竞争者为发展愿景，通过多年的技术积累，在 FPGA 芯片设计技术、SoC（system on a chip，片上系统）系统集成技术、FPGA 专用 EDA 软件技术、FPGA 芯片测试技术和 FPGA 应用解决方案等领域均有技术突破。

3. 高云半导体

广东高云半导体科技股份有限公司是一家专业从事国产现场可编程逻辑器件研发与产业化为核心，旨在推出具有核心自主知识产权的民族品牌 FPGA 芯片，提供集设计软件、IP 核、参照设计、开发板、定制服务等一体化完整解决方案的高科技企业。作为国内知名 FPGA 厂商，高云半导体于 2016 年顺利推出国内首颗 55nm 嵌入式 Flash SRAM 的非易失性 FPGA 芯片、于 2018 年成功研发出国内首款