

第一章 FPGA/CPLD 概述

本章讲述 FPGA/CPLD 的基本概念、结构、发展以及如何进行 FPGA/CPLD 设计等基本问题，初学者仔细阅读，但只需要了解，有个概念就行，不需要深入研究。对 FPGA/CPLD 比较熟悉地可以跳过此页。

1.1 FPGA/CPLD 的基本概念

CPLD 是复杂可编程逻辑器件 (Complex Programmable Logic Device) 的简称，FPGA 是现场可编程门阵列 (Field Programmable Gate Array) 的简称，两者的功能基本相同，编程等过程也基本相同 (烧写文件不一样，但是是由软件自动产生)，只是芯片内部的实现原理和结构略有不同 (见 1.3)，所以对初学者，可以忽略这两者的区别，统称为可编程逻辑器件或 CPLD/FPGA 或 PLD，我们下面就这样称呼。

简单地讲，PLD 是这样一种 ASIC (专用集成电路)：内部有大量的门电路，通过软件编程可以实现这些门电路不同的连接关系，从而整个 PLD 对外就完成了不同的功能，并且这些门电路的连接关系可以不断用软件来改变。

和单片机比较，PLD 在时序和延迟的实现上不如单片机，但是 PLD 在芯片容量、组合逻辑、工作速度、编程难度以及可以擦写次数上 (特别是 FPGA) 远优于单片机；另外还有很重要地一点，PLD 加密性能很好，具我们所知，目前国内只有 7032LC 系列可以解密，并且费用达到 3 万，其他都不可以解密，而单片机几千到几万一般都可以解密了。

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 70 年代单片机的发明和使用。

PLD 能做什么呢？可以毫不夸张的讲，PLD 能完成任何数字器件的功能，上至高性能 CPU，下至简单的 74 电路，都可以用 PLD 来实现。PLD 如同一张白纸或是一堆积木，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的设计一个数字系统。通过软件仿真，我们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 90 年代以后得到飞速的发展，同时也大大推动了 EDA 软件和硬件描述语言（HDL）的进步。目前大容量地 FPGA 地出现，FPGA 又可以用着 IC 设计的前端验证。

1.2 PLD 发展历程及概述

发展历程：

当今社会是数字化的社会，是数字集成电路广泛应用的社会。数字集成电路本身在不断地进行更新换代。它由早期的电子管、晶体管、小中规模集成电路、发展到超大规模集成电路(VLSIC，几万门以上)以及许多具有特定功能的专用集成电路。但是，随着微电子技术的发展，设计与制造集成电路的任务已不完全由半导体厂商来独立承担。系统设计者们更愿意自己设计专用集成电路(ASIC)芯片，而且希望 ASIC 的设计周期尽可能短，最好是在实验室里就能设计出合适的 ASIC 芯片，并且立即投入实际应用之中，因而出现了现场可编程逻辑器件(FPLD)，其中应用最广泛的当属现场可编程门阵列(FPGA)和复杂可编程逻辑器件(CPLD)。早期的可编程逻辑器件只有可编程只读存储器

(PROM)、紫外线可擦除只读存储器 (EPROM) 和电可擦除只读存储器 (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。其后, 出现了一类结构上稍复杂的可编程芯片, 即可编程逻辑器件 (PLD), 它能够完成各种数字逻辑功能。典型的 PLD 由一个“与”门和一个“或”门阵列组成, 而任意一个组合逻辑都可以用“与-或”表达式来描述, 所以, PLD 能以乘积和的形式完成大量的组合逻辑功能。

这一阶段的产品主要有 PAL (可编程阵列逻辑) 和 GAL (通用阵列逻辑)。PAL 由一个可编程的“与”平面和一个固定的“或”平面构成, 或门的输出可以通过触发器有选择地被置为寄存状态。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 EEPROM 技术。还有一类结构更为灵活的逻辑器件是可编程逻辑阵列 (PLA), 它也由一个“与”平面和一个“或”平面构成, 但是这两个平面的连接关系是可编程的。PLA 器件既有现场可编程的, 也有掩膜可编程的。在 PAL 的基础上, 又发展了一种通用阵列逻辑 GAL (Generic Array Logic), 如 GAL16V8, GAL22V10 等。它采用了 EEPROM 工艺, 实现了电可擦除、电可改写, 其输出结构是可编程的逻辑宏单元, 因而它的设计具有很强的灵活性, 至今仍有许多人使用。这些早期的 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(Complex Programmable Logic Device)和与标准门阵列类似的 FPGA(Field Programmable Gate Array),它们都具有体系结构和逻辑单元灵活、集成度高以及适用范围宽等特点。这两种器件兼容了 PLD 和通用门阵列的优点,可实现较大规模的电路,编程也很灵活。与门阵列等其它 ASIC(Application Specific IC)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、质量稳定以及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生产(一般在 10,000 件以下)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应用 FPGA 和 CPLD 器件。

(注:不同厂家的叫法不尽相同,Xilinx 把,基于查找表技术,SRAM 工艺,要外挂配置用的 EEPROM 的 PLD 叫 FPGA;把基于乘积项技术,Flash(类似 EEPROM 工艺)工艺的 PLD 叫 CPLD;Altera 把自己的 PLD 产品:MAX 系列(乘积项技术,EEPROM 工艺),FLEX 系列(查找表技术,SRAM 工艺)都叫作 CPLD,即复杂 PLD(Complex PLD),由于 FLEX 系列也是 SRAM 工艺,基于查找表技术,要外挂配置用的 EPROM,用法和 Xilinx 的 FPGA 一样,所以很多人把 Altera 的 FELX 系列产品也叫做 FPGA。

我们根据目前市场上各个厂家的芯片,我们描述如下:

FPGA: SRAM 工艺;直接烧写程序掉电后程序丢失;理论上擦写 100 万次以上;一般使用需要外挂 EEPROM,可以达到几百万门电路。比如 ALTERA 公司的 APEX、FLEX、ACEX、STRATIX、CYCLONE 系列。

CPLD: EEPROM 或 FLASH 工艺;直接烧写程序调电后程序不会消失;一般可以擦写几百次,并且一般宏单元在 512 以下。比如 ALTERA 的 MAX3000/5000/7000/9000 和 CLASSIC 系列)

FPGA / CPLD 概述

FPGA(现场可编程门阵列)与 CPLD(复杂可编程逻辑器件)都是可编程逻辑器件,它们是在 PAL, GAL 等逻辑器件的基础之上发展起来的。同以往的 PAL, GAL 等相比较, FPGA / CPLD 的规模比较大,它可以替代几十甚至几千块通用 IC 芯片。这样的 FPGA / CPLD 实际上就是一个子系统部件。这种芯片受到世界范围内电子工程设计人员的广泛关注和普遍欢迎。经过了十几年的发展,许多公司都开发出了多种可编程逻辑器件。比较典型的就 Xilinx 公司的 FPGA 器件系列和 Altera 公司的 CPLD 器件系列,它们开发较早,占用了较大的 PLD 市场。通常来说,在欧洲用 Xilinx 的人多,在日本和亚太地区用 ALTERA 的人多,在美国则是平分秋色。全球 PLD/FPGA 产品 60%以上是由 Altera 和 Xilinx 提供的。可以讲 Altera 和 Xilinx 共同决定了 PLD 技术的发展方向。当然还有许多其它类型器件,如:Lattice, Vantis, Actel, Quicklogic, Lucent 等。(99 年 Lattice 收购了 Vantis,成为第三大 PLD 供应商;同年 Xilinx 收购了 Philips 的 PLD 部门)

表 1.2 - 1 1998 年世界十大 PLD 公司

排名	公司	销售额 (亿美金)	市场占有率
1	Altera	5.96	30.1
2	Xilinx	5.74	29.0
3	Vantis	2.20	11.1
4	Lattice	2.18	11.0
5	Actel	1.39	7.0
6	Lucent	0.85	4.3

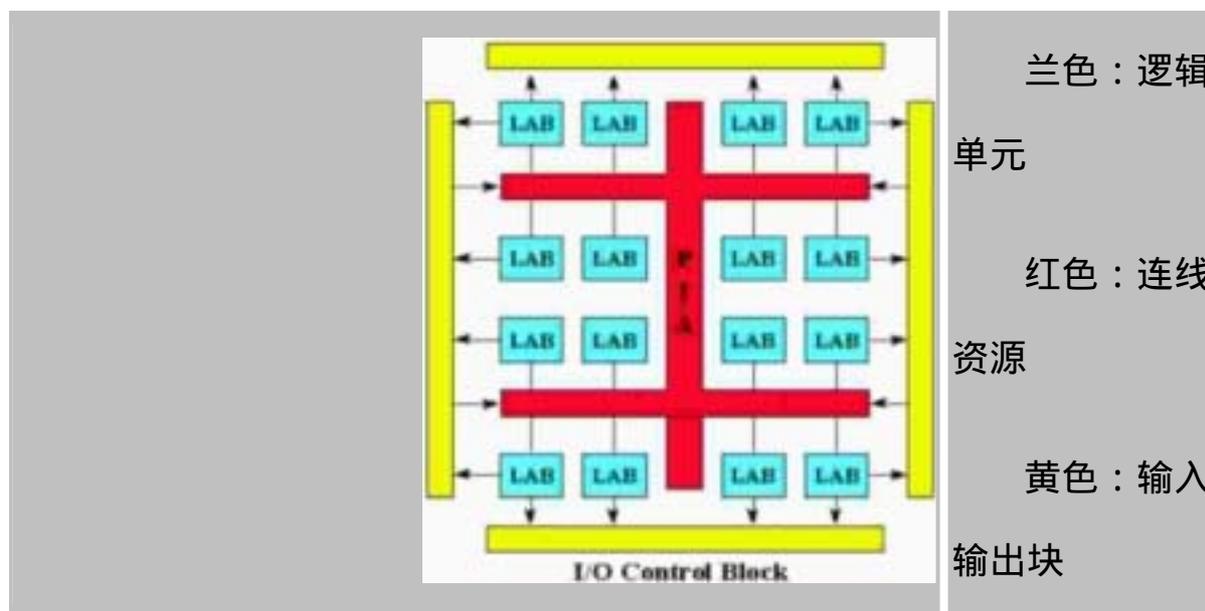
7	Cypress	0.44	2.2
8	Atmel	0.42	2.1
9	Philips	0.28	1.4
10	Quicklogic	0.24	1.2

资料来源：99年4月《电子产品世界》

尽管 FPGA, CPLD 和其它类型 PLD 的结构各有其特点和长处，但概括起来，它们是由三大部分组成的，

- 一个二维的逻辑块阵列，构成了 PLD 器件的逻辑组成核心。
- 输入 / 输出块：
- 连接逻辑块的互连资源。连线资源：由各种长度的连线线段组成，其中也有一些可编程的连接开关，它们用于逻辑块之间、逻辑块与输入 / 输出块之间的连接。

图：1.2 - 2 典型的 PLD 的框图



对用户而言，CPLD 与 FPGA 的内部结构稍有不同，但用法一样，所以多数情况下，不加以区分。

FPGA / CPLD 芯片都是特殊的 ASIC 芯片，它们除了具有 ASIC 的特点之外，还具有以下几个优点：

- 随着 VLSI (Very Large Scale IC, 超大规模集成电路) 工艺的不断提高单一芯片内部可以容纳上百万个晶体管，FPGA / CPLD 芯片的规模也越来越大，其单片逻辑门数已达到上百万门，它所能实现的功能也越来越强，同时也可以实现系统集成。

- FPGA / CPLD 芯片在出厂之前都做过百分之百的测试，不需要设计人员承担投片风险和费用，设计人员只需在自己的实验室里就可以通过相关的软硬件环境来完成芯片的最终功能设计。所以，FPGA / CPLD 的资金投入小，节省了许多潜在的花费。

- 用户可以反复地编程、擦除、使用或者在外围电路不动的情况下用不同软件就可实现不同的功能。所以，用 FPGA / PLD 试制样片，能以最快的速度占领市场。FPGA / CPLD 软件包中有各种输入工具和仿真工具，及版图设计工具和编程器等全线产品，电路设计人员在很短的时间内就可完成电路的输入、编译、优化、仿真，直至最后芯片的制作。当电路有少量改动时，更能显示出 FPGA / CPLD 的优势。电路设计人员使用 FPGA / CPLD 进行电路设计时，不需要具备专门的 IC(集成电路)深层次的知识，FPGA / CPLD 软件易学易用，可以使设计人员更能集中精力进行电路设计，快速将产品推向市场。

(以上部分内容内容参照 WWW.FPGA.COM.CN)

1.3 PLD 的结构和原理

1.3.1 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) 我们先看一下这种

PLD 的总体结构(以 MAX7000 为例,其他型号的结构与此都非常相似):

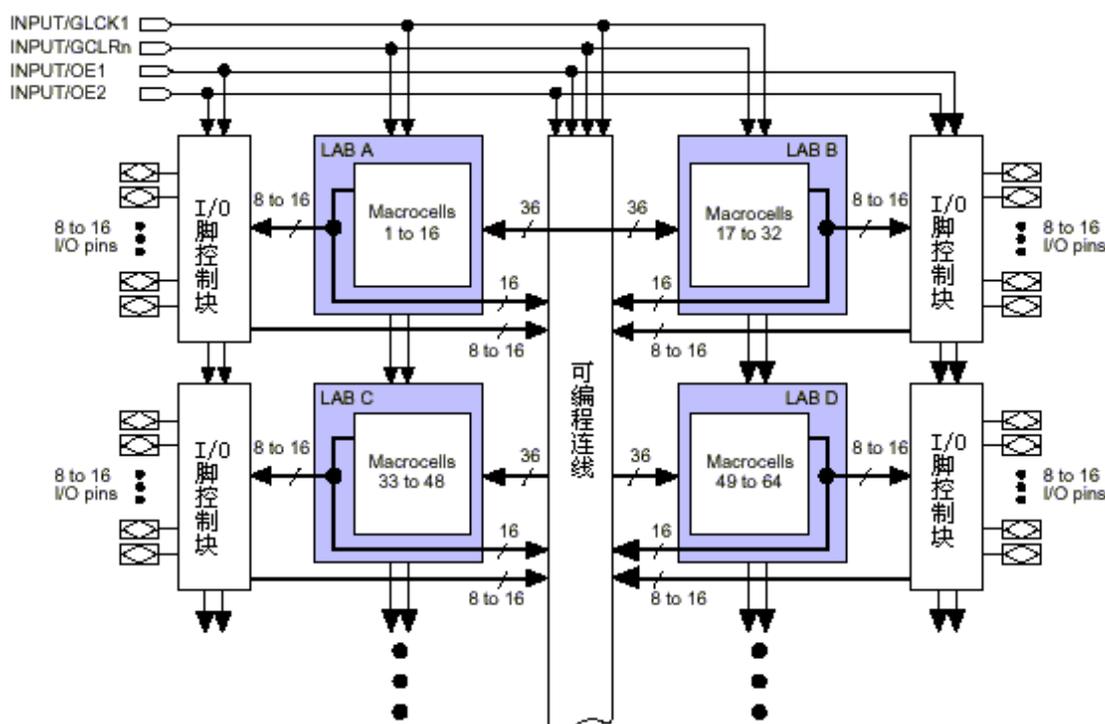


图 1.3.1 - 1 基于乘积项的 PLD 内部结构

这种 PLD 可分为三块结构：宏单元 (Macrocell)，可编程连线 (PIA) 和 I/O 控制块。宏单元是 PLD 的基本结构，由它来实现基本的逻辑功能。图 1 中蓝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)。可编程连线负责信号传递，连接所有的宏单元。I/O 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，

摆率控制，三态输出等。上图的 INPUT/GCLK1，INPUT/GCLRn, INPUT/OE1, 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。宏单元的具体结构见下图：

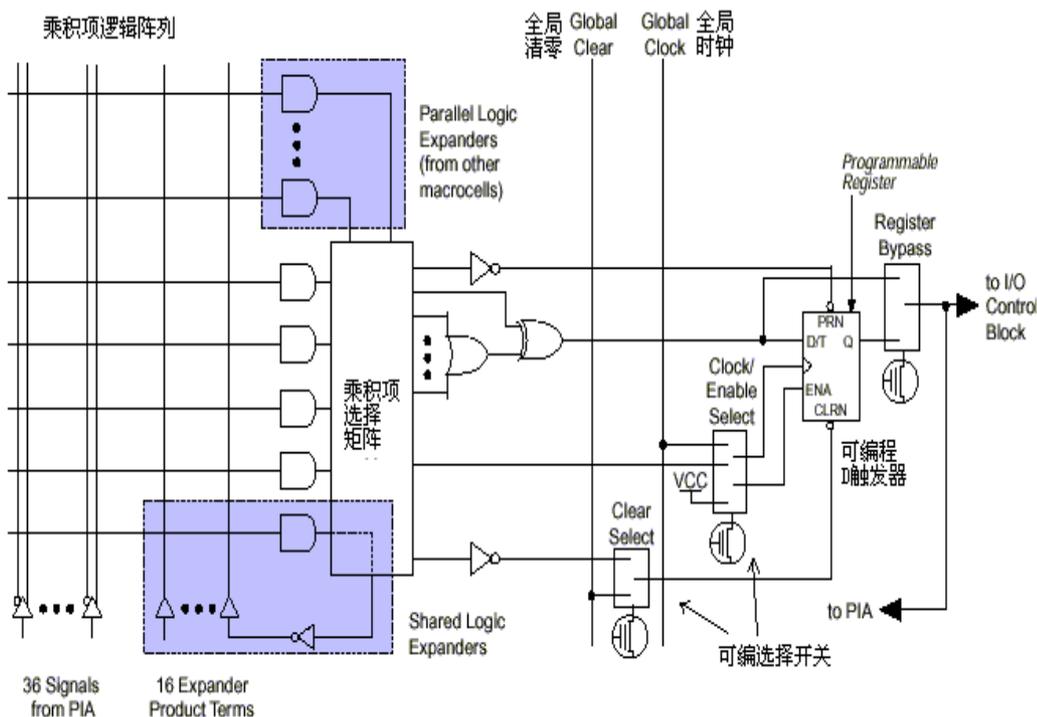


图 1.3.1 - 2 宏单元结构

左侧是乘积项阵列，实际就是一个与或阵列，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。图右侧是一个可编程 D 触发器，它的时钟，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）产生的时钟和清零。如果不需要触发器，也可以将此触发器旁路，信号直接输给 PIA 或输出到 I/O 脚。

1.3.2 乘积项结构 PLD 的逻辑实现原理

下面我们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,电路如下图:

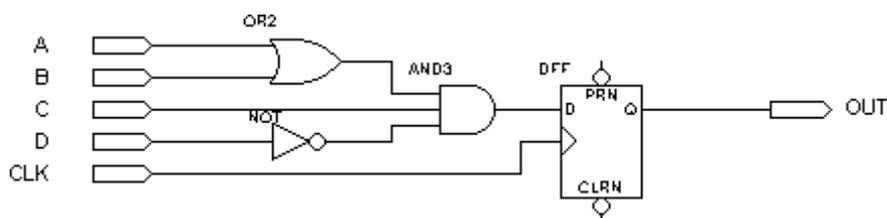


图 1.3.2 - 1

假设组合逻辑的输出(AND3 的输出)为 f , 则

$$f = (A+B) * C * (!D) = A * C * !D + B * C * !D \quad (\text{我们以} !D \text{表示} D \text{的“非”})$$

PLD 将以下面的方式来实现组合逻辑 f :

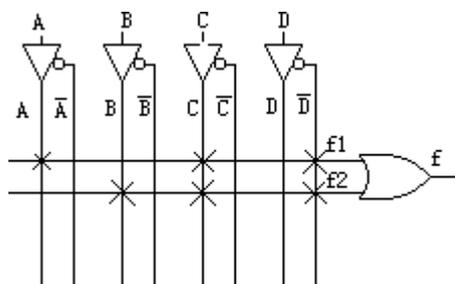


图 1.3.2-2

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。图中每一个叉表示相连 (可编程熔丝导通), 所以得到: $f = f1 + f2 = (A * C * !D) + (B * C * !D)$ 。这样组合逻辑就实现了。图 3 电路中 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 脚输入后进入芯片内部的全局时钟专用通道,直接连接到可编程触发器的时钟端。可编程触发器的输出与 I/O 脚相连,把结果输出到芯片管脚。这样 PLD 就完成了图 3 所示电路的功能。(以上这些步骤都是由软件自动完成的,不需要人为干预)

图 1.3.2-1 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,一个宏单元是不能实现的,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,再做为另一个宏单元的输入。这样 PLD 就可以实现更复杂逻辑。

这种基于乘积项的 PLD 基本上都是由 EEPROM 和 Flash 工艺制造的,一上电就可以工作,无需其他芯片配合。

1.3.3 查找表 (Look-Up-Table) 的原理与结构

采用这种结构的 PLD 芯片我们也可以称之为 FPGA:如 altera 的 ACEX, APEX 系列, xilinx 的 Spartan, Virtex 系列等。

查找表 (Look-Up-Table) 简称为 LUT, LUT 本质上就是一个 RAM。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 16x1 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, PLD/FPGA 开发软件会自动计算逻辑电路的所有可能的结果, 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。

下面是一个 4 输入与门的例子 :

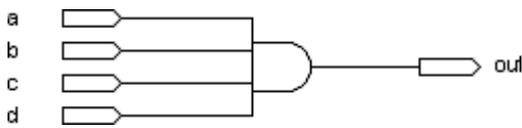
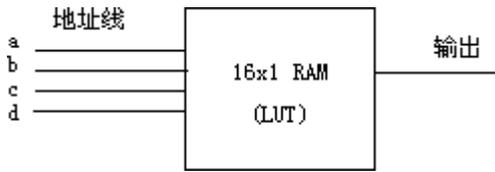
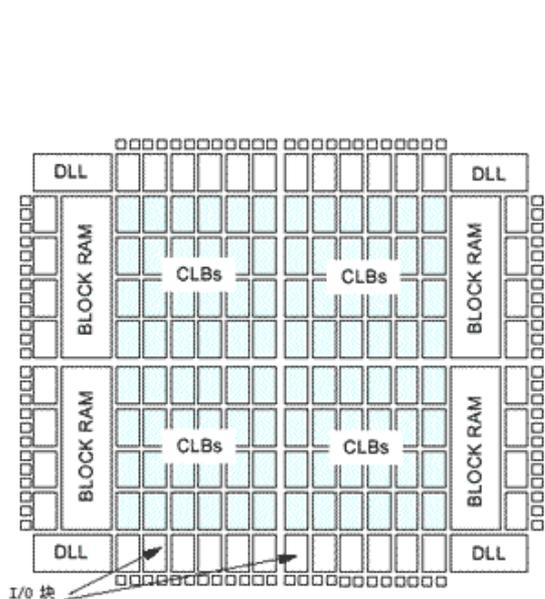
实际逻辑电路		LUT 的实现方式	
			
a, b, c, d 输入	逻辑输出	地址	RAM 中存储的内容
0000	0	0000	0
0001	0	0001	0
....	0	...	0
1111	1	1111	1

图 1.3.3 - 1 基于查找表 (LUT) 的 FPGA 的结构

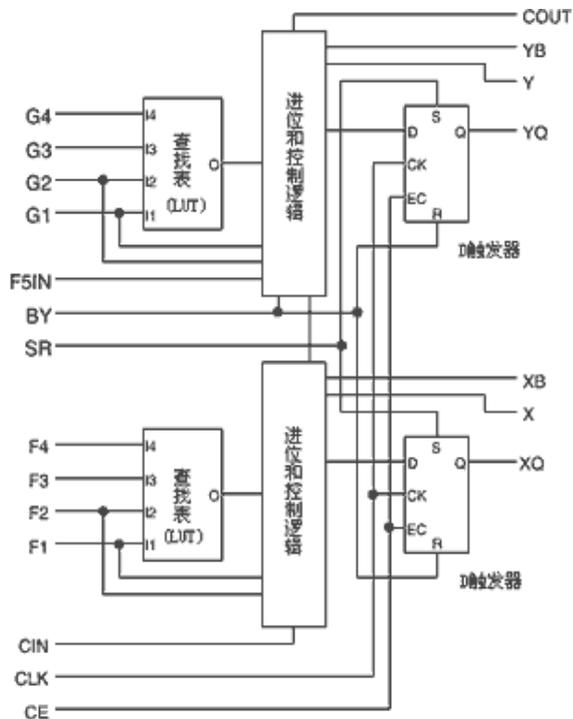
我们看一看 xilinx Spartan-II 的内部结构,如图 1.3.3 - 2 :

Spartan-II 主要包括 CLBs ,I/O 块 ,RAM 块和可编程连线(未表示出)。在 spartan-II 中,一个 CLB 包括 2 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 SpartanII 实现

逻辑的最基本结构 (xilinx 其他系列, 如 SpartanXL, Virtex 的结构与此稍有不同, 具体请参阅数据手册)



xilinx Spartan-II 芯片内部结构



Slices 结构

图 1.3.3 - 2

al tera 的 FLEX/ACEX 等芯片的结构如下图：

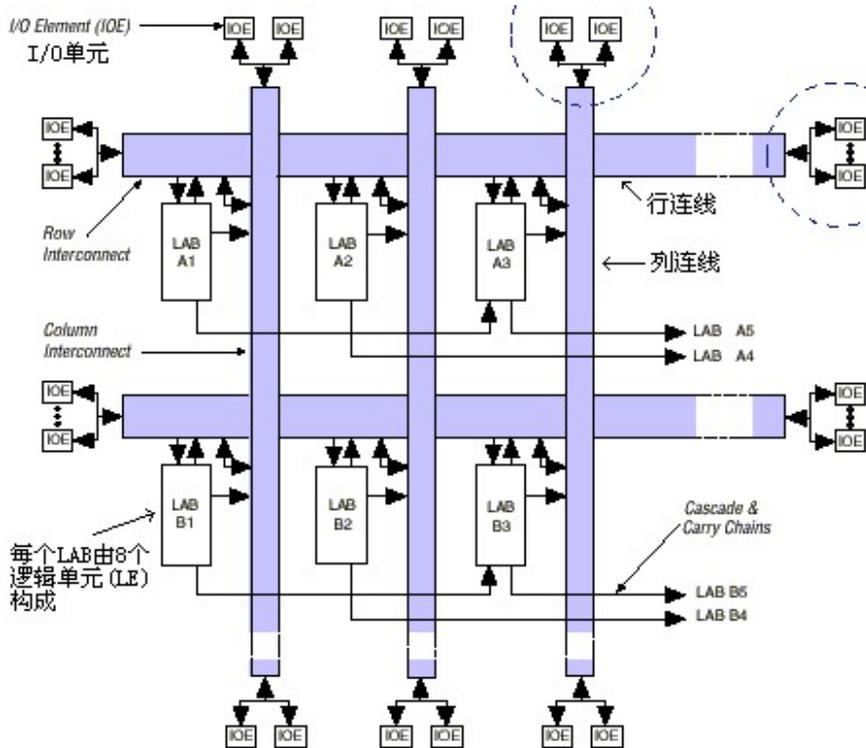


图 1.3.3 - 3 al tera FLEX/ACEX 芯片的内部结构

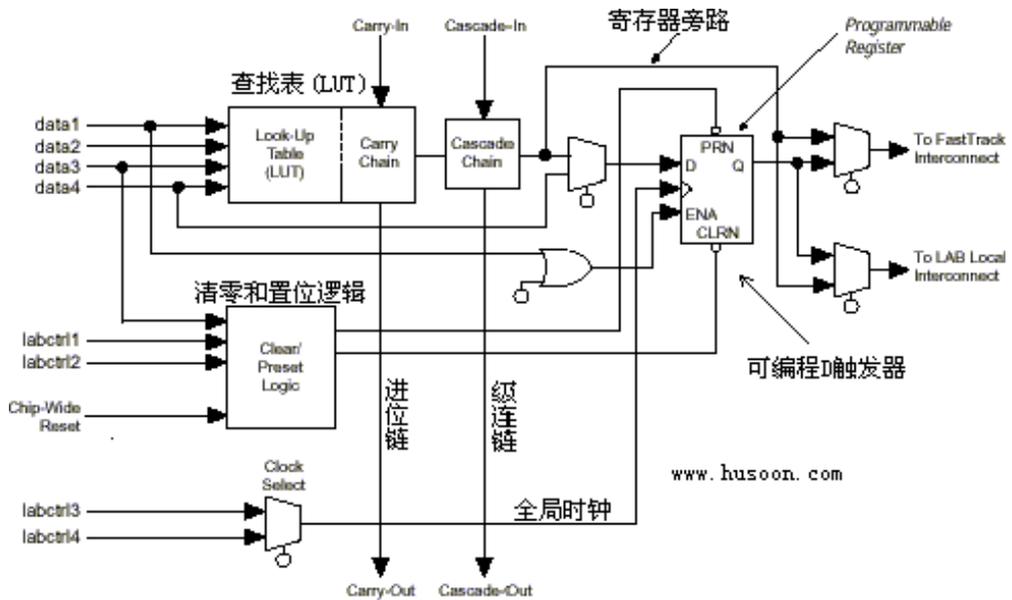


图 1.3.3 - 4 逻辑单元 (LE) 内部结构

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 , 一个触发器和相关的相关逻辑。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 , 具体请参阅数据手册)

1.3.4 查找表结构的 FPGA 逻辑实现原理

我们还是以这个电路的为例 :

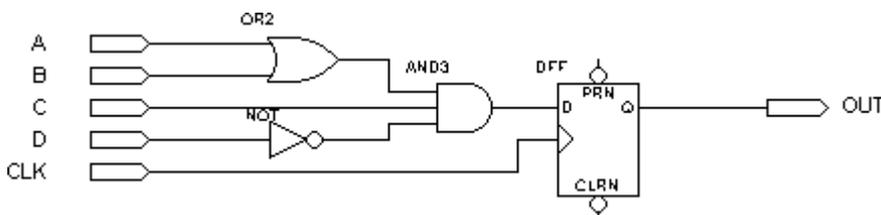


图 1.3.4 - 1

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 , 这样组合逻辑就实现了。该电路中 D 触发器是直接利用 LUT 后面 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 就完成了图 1.3.4 - 1 所示电路的功能。(以上这些步骤都是由软件自动完成的 , 不需要人为干预)

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 就可以实现复杂的逻辑。

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,对这种 FPGA,就不需要外加专用的配置芯片。

1.3.5 选择 CPLD 还是 FPGA ?

根据上一篇 PLD 的结构和原理可以知道,CPLD 组合逻辑的功能很强,一个宏单元就可以分解十几个甚至 20 - 30 多个组合逻辑输入。而 FPGA 的一个 LUT 只能处理 4 输入的组合逻辑,因此,CPLD 适合用于设计译码等复杂组合逻辑。但 FPGA 的制造工艺确定了 FPGA 芯片中包含的 LUT 和触发器的数量非常多,往往都是几千上万,CPLD 一般只能做到 512 个逻辑单元,而且如果用芯片价格除以逻辑单元数量,FPGA 的平均逻辑单元成本大大低于 CPLD。所以如果设计中使用大量触发器,例如设计一个复杂的时序逻辑,那么使用 FPGA 就是一个很好选择。

对于初学者，一般使用 CPLD，因为 CPLD 芯片价格低（象 EPM7064SLC44-10 只要 22 元就够了）；许多 CPLD 为 5V，可以直接和 CMOS 以及 TTL 电路电压兼容，不必考虑电源转换问题；CPLD 很多芯片的封装的是 PLCC，插拔很方便，而 FPGA 一般是 QFP 封装，一旦损坏，很难从系统电路上取下。但对于产品开发适合选用 FPGA。

1.4 PLD 的开发和设计

PLD 的开发和设计需要三方面：（1）PLD 软件；（2）PLD 硬件；（3）电路设计的基本知识

PLD 软件：

由于 PLD 软件已经发展的相当完善，用户甚至可以不用详细了解 PLD 的内部结构，也可以用自己熟悉的方法：如原理图输入或 HDL 语言来完成相当优秀的 PLD 设计。（但了解 PLD 的内部结构，将有助于提高我们设计的效率和可靠性）

如何获得 PLD 开发软件软件呢？ ALTERA 公司提供免费试用版或演示版（当然商业版大都是收费的），例如：可以免费从 <http://www.altera.com/> 上下载 Altera 公司的 Maxplus2（Baseline 版或 E+MAX 版，但 ALTERA 公司目前需要免费注册后才可以下载），或向其代理商索取这套软件。以上免费软件都需要在网上注册申请 License 文件。通常这些免费软件已经能够满足一般设计的需要，当然，要想软件功能更强大一些，只能购买商业版软件。

如果您打算使用 VHDL 或 Verilog HDL 硬件描述语言来开发 PLD/FPGA, 通常还需要使用一些专业的 HDL 开发软件, 这是因为 FPGA 厂商提供的软件的 HDL 综合能力一般都不是很强, 需要其他软件来配合使用, 但作为初学者, ALTERA 公司的 MAXPLUS 和 QUARTUS 可以完成基本的设计。

PLD 硬件:

开发 ALTERA 公司的 PLD 需要以下硬件: 电脑 (带并口)、PLD 芯片 (如果是 FPGA 还需要 EEPROM, 没有 EEPROM 那么调电程序消失)、电源、下载电缆和下载电路。

(对于上面软硬件, 如果你购买 www.husoon.com 的 HS102 型 FPGA/CPLD 开发板, 你将获得除电脑外的所有资源, 其中软件不作为我们的产品, 仅供学习使用)

电路设计的基本知识

主要指一些数字模拟电路、HDL 语言等知识和实际电路设计经验。时下很多初学者认为有了 MAXPLUS 和 QUARTUS 就会 FPGA 了, 其实这是很不对的, 实际上数字模拟电路、HDL 语言等知识和实际电路设计经验比了解软件重要; 学会软件的使用就象你学会拿铲子了, 但作出怎么样的才完全在你做菜的水平上。