



大连海事大学
DALIAN MARITIME UNIVERSITY



可编程逻辑器件 设计基础

课程编号：1713013170

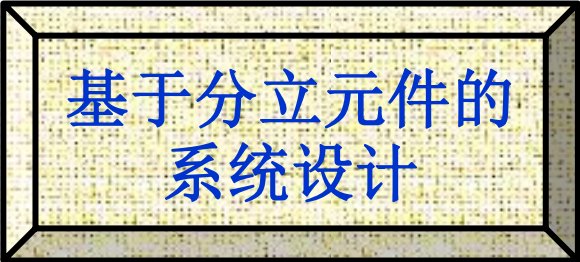
学时/学分：24/1.5

考核方式：考查

2021~2022学年第一学期

课程简介

- 主要目的：培养学生应用可编程逻辑器件知识和硬件描述语言知识设计数字系统的能力。
- 主要内容：可编程逻辑器件的基本结构和工作原理，常用设计工具软件的使用方法，硬件描述语言的程序结构、语言要素等知识，基于**PLD**的数字系统设计过程和方法。
- 数字电路系统的设计方法



基于分立元件的
系统设计



基于
PLD
的系统
设计



基于
计算机的
系统设计



课程目标



1

能解释可编程逻辑器件相关的常用术语与基本概念

2

能应用硬件描述语言，实施程序编写

3

能基于开发环境和硬件描述语言完成可编程逻辑器件设计流程

4

能设计基于可编程逻辑器件的数字系统

5

能查找阅读可编程逻辑器件有关的技术资料



主要内容 CONTENTS

1

概述

2

硬件描述语言基础

3

Verilog语言语法

4

数字逻辑单元设计

5

数字系统综合设计

课程说明

□ 课程教材：

《可编程逻辑器件设计基础》 自编讲义2021年9月

□ 参考书：

《EDA技术与Verilog设计》 王金明编著, 电子工业出版社, 2019年2月

《EDA原理及Verilog HDL实现》 何宾主编, 清华大学出版社, 2016年12月

《EDA技术与VerilogHDL》 黄继业编著, 清华大学出版社, 2010年4月

□ 考核方式：考查

□ 成绩构成：平时成绩50%+期末考试50%

第1章EDA技术概述

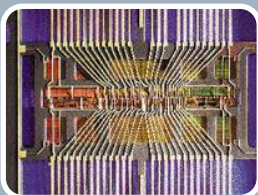
课程目标

课程目标1：能解释可编程逻辑器件相关的常用术语与基本概念

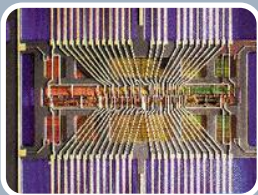
课程目标5：能查找阅读可编程逻辑器件有关的技术资料

第1章基础知识

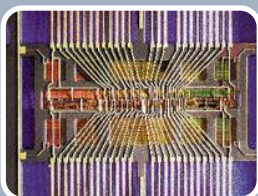
第1章基础知识



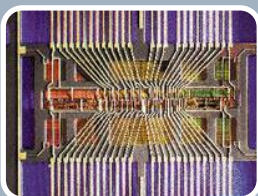
1.1 EDA技术综述



1.2 PLD设计方法



1.3 可编程逻辑器件



1.4 EDA相关技术

1.1 EDA技术综述

1.1.1 EDA技术发展:

随着计算机、集成电路、电子系统设计的发展，经历了三个发展阶段：

- 计算机辅助设计**CAD (Computer Assist Design)**: 20世纪70年代，技术发展初期。
- 计算机辅助工程设计**CAE (Computer Assist Engineering)**: 20世纪80年代。
- 电子设计自动化**EDA (Electronic Design Automation)**: 20世纪90年代。



1.1 EDA技术综述

1.1.2 EDA技术涵义

□ 广义EDA技术：通过计算机及电子系统的辅助分析和设计软件，完成电子系统的设计过程。

□ 狭义EDA技术：以大规模PLD为设计载体，以HDL为系统逻辑描述的主要表达方式，以大规模PLD开发软件及实验开发系统为设计工具，完成用软件方式设计的电子系统到硬件系统的转换工作，最终形成集成电子系统或专用集成芯片的技术。

编译、化简、分割、综合及优化、布局布线、仿真；
对特定目标芯片的适配编译、逻辑映射、编程下载工作

EDA

第1章基础知识

3. 完成目标芯片的适配编译、逻辑映射和编程下载等工作

2. 自动地完成逻辑编译、综合、优化、布局布线和仿真

1. 采用硬件描述语言(或原理图)完成设计输入

基于**EDA**
软件平台

以**PLD**
为设计载体

实现
既定
电子
线路
系统
功能

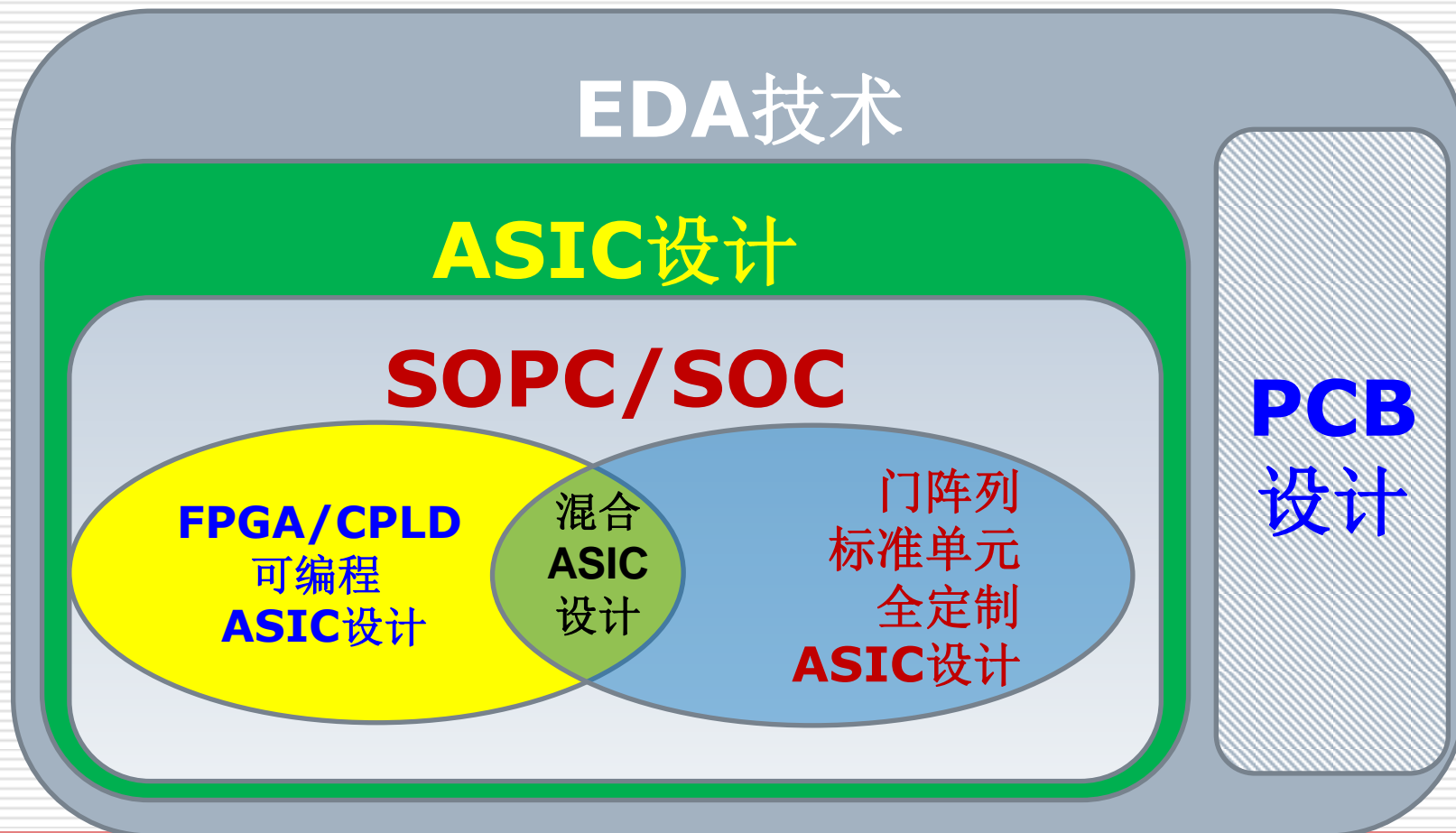
特点:

- 利用软件的方式完成对系统硬件功能的实现。
- 所设计电路直接实现在专用芯片上, 满足千差万别的用户设计要求。



EDA技术实现目标

完成专用集成电路ASIC或印制电路板PCB的设计与实现



1.1 EDA技术综述

1.1.3 EDA技术主要内容：

- ① 大规模可编程逻辑器件PLD：利用EDA技术进行电子系统设计的载体；
- ② 硬件描述语言HDL：是利用EDA技术进行电子系统设计的主要表达手段；
- ③ EDA设计软件EDAS：是利用EDA技术进行电子系统设计的自动化设计工具；
- ④ 相关的硬件平台：是利用EDA技术进行电子系统设计的下载工具及硬件验证工具；

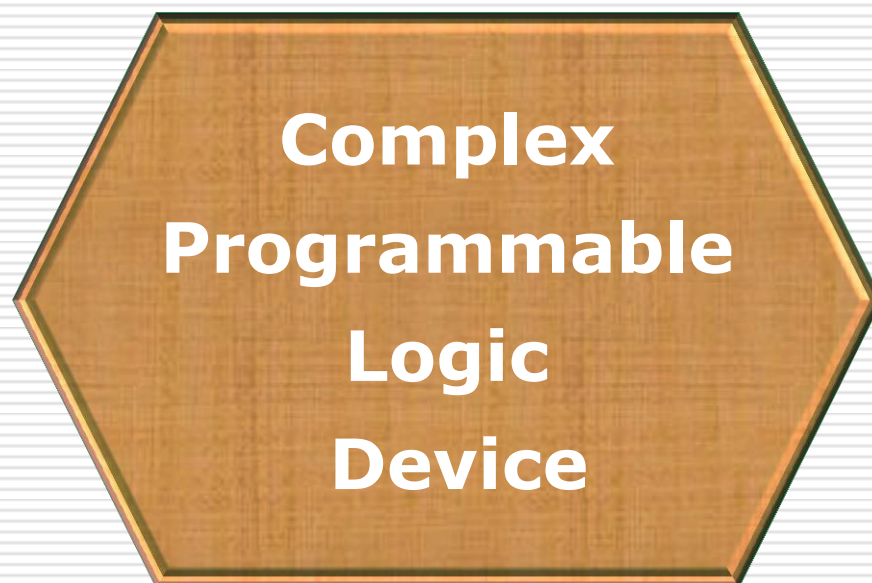


(1) 大规模可编程逻辑器件

◆ 类型:



现场可编程门阵列
FPGA



复杂可编程逻辑器件
CPLD

(1) 大规模可编程逻辑器件

◆特点:

- 集成规模大，便于EDA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的独立性使得各类逻辑功能块软件有很好的兼容性和可移植性。
- 开发周期短、投资风险小、产品上市速度快、市场适应能力强和硬件升级回旋余地大；充分检验后的VHDL设计可迅速实现ASIC投产。

(2) 硬件描述语言HDL

类型	VHDL	Verilog	ABEL
起源	美国国防部的VHSIC	集成电路的设计	可编程逻辑器件的设计
描述层次	高级描述语言，适用于行为级和RTL级的描述，最适于描述电路的行为	低级描述语言，适用于RTL级和门电路级的描述，最适于描述门级电路	低级描述语言，适用于RTL级和门电路级的描述，最适于描述门级电路
设计要求	设计者可不了解电路的结构细节	设计者需了解电路的结构细节	设计者需了解电路的结构细节
综合过程	经过行为级→RTL级→门电路级的转化，几乎不能直接控制门电路的生成	经过RTL级→门电路级的转化，易于控制电路资源，简单	经过RTL级→门电路级的转化，易于控制电路资源，简单
综合器的要求	层次较高，不易控制底层电路，对综合器的性能要求较高	对综合器的性能要求较低	对综合器的性能要求较低
EDA工具	支持EDA工具很多	支持EDA工具很多	支持EDA工具少 Dataio

(3) 开发工具软件EDAS

- 意义：以计算机软件为主，将典型的单元电路封装起来形成固定模块并形成标准的硬件开发语言（如HDL语言）供设计人员使用。
- 功能：自动地完成逻辑编译、化简、分割、综合及优化、布局布线、仿真以及对于特定目标芯片的适配编译和编程下载等工作。
- 构成：综合器和适配器两个软件包。
- 综合器功能：将设计者在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。

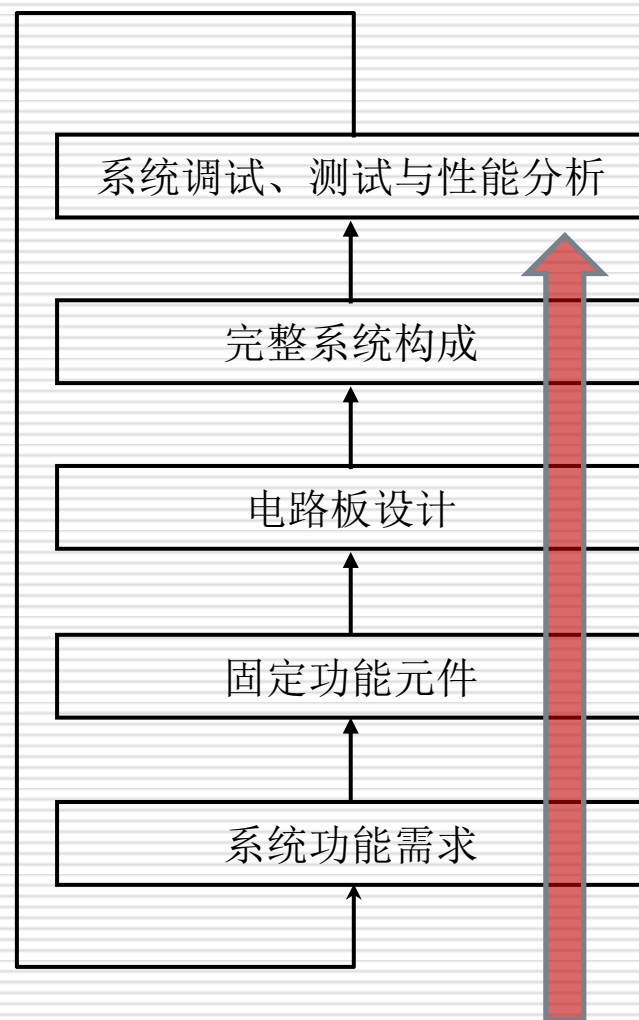
(4) 硬件开发平台

- 意义：提供芯片下载电路及EDA实验/开发的外围资源，以供硬件验证用。
- 构成：
 - ① 实验或开发所需的各类基本信号发生模块，包括时钟、脉冲、高低电平等；
 - ② PLD输出信息显示模块，包括数码显示、发光管显示、声响指示等；
 - ③ 监控程序模块，提供“电路重构软配置”；
 - ④ 目标芯片适配座以及上面的FPGA/CPLD目标芯片和编程下载电路。

1.2 PLD设计方法

1.2.1 电子设计传统方法

- ❑ 特点：以固定功能元件为基础，基于电路板的自下而上设计方法。
- ❑ 缺点：
 - 依赖于设计人员的经验；
 - 依赖于现有通用元器件；
 - 设计后期的仿真不易实现，并且调试复杂；
 - 设计实现周期长，灵活性差，耗时耗力，效率低下。



□ Bottom-up设计

- 选择标准集成电路，或将各种基本单元（如各种门电路及加法器、计数器模块等）做成基本单元库，调用这些基本单元，逐级向上组合，直到设计出满足需要的系统。
- 使设计者关注细节，对系统的整体性能，当设计出现问题时，常需要从头再来。
- 效率低、成本高，容易出错。



建金字塔

1.2 PLD设计方法

1.2.2 PLD设计

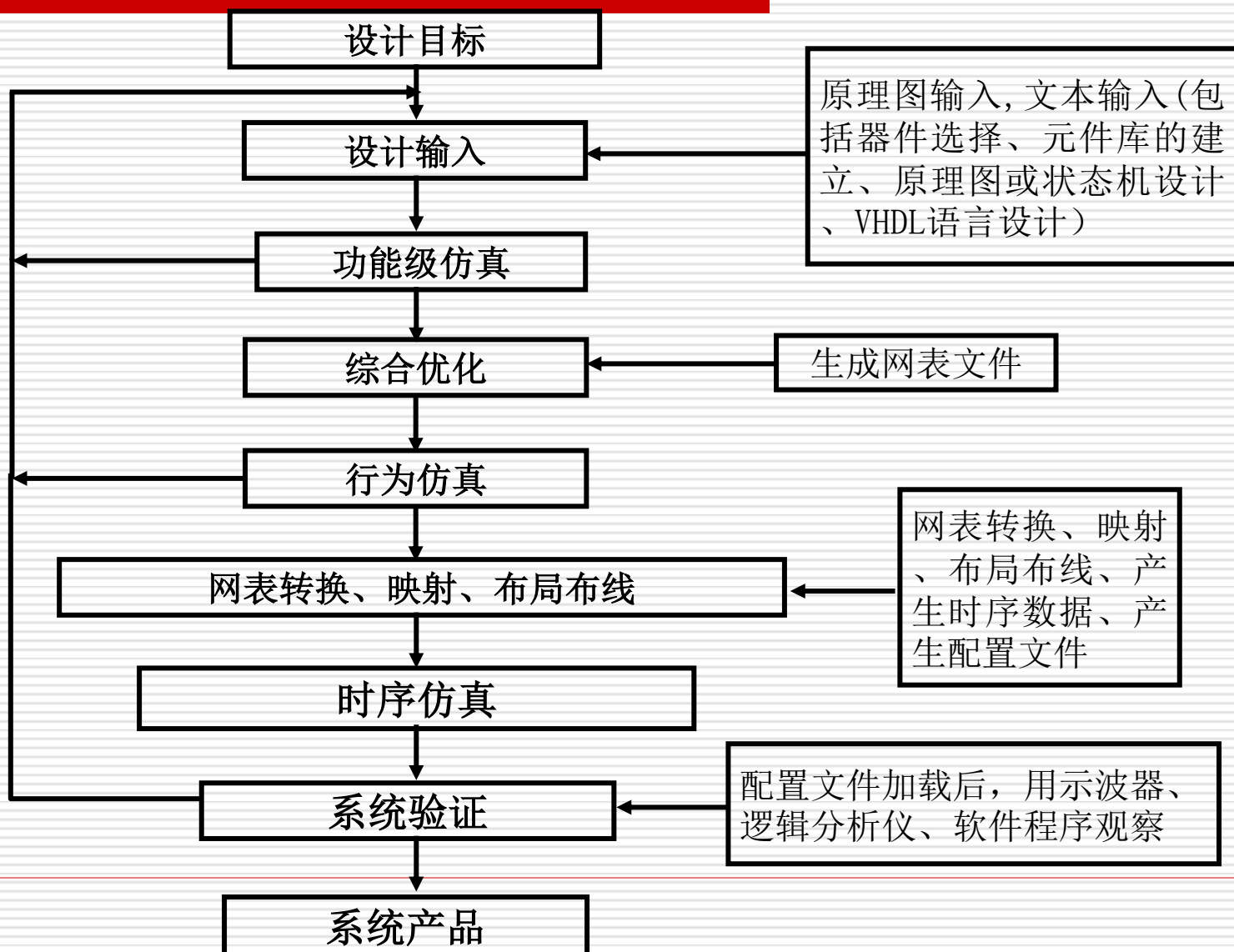
□ PLD设计特点

- 通过芯片的设计实现各种逻辑数字功能，将原来电路板设计完成的工作大部分放到芯片设计中进行。
- 由设计者自己定义器件的内部逻辑和引脚，引脚定义具有灵活性，降低了原理图和印刷板设计的工作量及难度，增加了设计的自由度，提高了效率。
- 减少了所需芯片的种类和数量，缩小了体积，降低了功耗，提高了系统的可靠性。

□ Top-down设计

- 从系统设计入手，在顶层进行系统功能方框图的划分和结构设计
- 在功能级进行仿真、纠错，并用**HDL**对高层次的系统行为进行描述，然后采用综合工具将设计转化为具体门电路网表，对应的物理实现是**PLD**器件或专用集成电路**ASIC**。
- 设计的主要仿真和调试过程是在高层次上完成的，有利于早期发现结构设计上的错误，避免设计工作的浪费，也减少了逻辑功能仿真的工作量，提高了设计的一次成功率。
- 设计过程：设计→验证→修改设计→再验证

FPGA开发的一般流程



1.2.2 PLD设计流程

① 设计目标

- 在系统设计之前，首先进行方案论证、系统设计和**FPGA**芯片选择等准备工作。
- 根据任务要求(如系统的指标和复杂度)，对工作速度和芯片本身的各种资源、成本等方面进行权衡，选择合理的设计方案和合适的器件类型。
- 采用自顶向下设计方法，把系统分成若干个基本单元，然后再把每个基本单元划分为下一层次的基本单元，直到可以直接使用**EDA**元件库为止。

② 设计输入

- 将所设计的系统或电路以开发软件要求的某种形式表示出来，并输入**EDA**工具。
- 常用方法：
 - 原理图输入：将所需的器件从元件库中调出来，画出原理图。描述方式直观并易于仿真，在**PLD**发展的早期应用比较广泛，但效率很低，且不易维护，更主要的缺点是可移植性差。
 - **HDL**语言输入法：目前在实际开发中应用广泛，主流语言是**Verilog HDL**和**VHDL**。

③ 功能仿真（前仿真）

- 在编译前对所设计的电路进行逻辑功能验证。
- 功能仿真没有延迟信息，仅对初步的功能进行检测。
- 仿真前，要先利用波形编辑器和**HDL**等建立波形文件和测试向量（即将所关心的输入信号组合成序列）；
- 仿真结果将会生成报告文件和输出信号波形，从中便可以观察各个节点信号的变化。若发现错误，则返回设计修改逻辑设计。

④ 综合**Synthesis**优化

- 将较高级抽象层次的描述转化成较低层次的描述。
- 具体是指将设计输入编译成由与门、或门、非门、**RAM**、触发器等基本逻辑单元组成的逻辑连接网表，根据目标与要求优化所生成逻辑连接；
- 综合优化使层次设计平面化，供**PLD**布局布线软件进行实现。
- 利用**PLD**制造商的布局布线功能，由综合后生成的标准门级结构网表产生真实具体的门级电路。
- 所有综合器都支持门级结构、**RTL**级的**HDL**程序的综合。

⑤ 综合后仿真

- 检查综合结果是否和原设计一致。
- 在仿真时，把综合生成的标准延时文件反标注到综合仿真模型中去，可估计门延时带来的影响。
- 由于该步骤不能估计线延时，因此和布线后的实际情况还有一定的差距，并不十分准确。
- 目前的综合工具较为成熟，对于一般的设计可以省略这一步。
- 如果在布局布线后发现电路结构和设计意图不符，则需要回到综合后仿真来确认问题之所在。

⑥ 布局布线

- 将综合生成的逻辑网表配置到具体的**PLD**芯片；
- 布局将逻辑网表中的硬件原语和底层单元合理地配置到芯片内部的固有硬件结构上，并且往往需要在速度最优和面积最优之间作出选择；
- 布线根据布局的拓扑结构，利用芯片内部的各种连线资源，合理正确地连接各个元件。布线结束后，软件工具会自动生成报告，提供有关设计中各部分资源的使用情况。
- 由于只有**PLD**芯片生产商对芯片结构最为了解，所以布局布线必须选择芯片开发商提供的工具。

⑦ 时序仿真（后仿真）

- 将布局布线的延时信息反标注到设计网表中来检测有无时序违规（即不满足时序约束条件或器件固有的时序规则，如建立时间、保持时间等）现象。
- 时序仿真包含的延迟信息最全，也最精确，能较好地反映芯片的实际工作情况。
- 不同芯片的内部延时不同，不同的布局布线方案也给延时带来不同的影响，通过对系统和各个模块进行时序仿真，分析其时序关系，估计系统性能，以及检查和消除竞争冒险很有必要。

⑧ 板级仿真与验证

- 板级仿真主要应用于高速电路设计中，对高速系统的信号完整性、电磁干扰等特征进行分析，一般都以第三方工具进行仿真和验证。

⑨ 芯片编程与调试

- 指产生使用的数据文件，然后将编程数据下载到**PLD**芯片中。

1.3 可编程逻辑器件

IC设计方法分类：根据产量、设计周期等因素等。

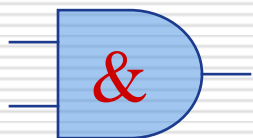
集成度

PLD器件	简单型PLD SPLD	PROM
		PLA
		PAL
		GAL
	高密度PLD HDPLD	CPLD
		FPGA

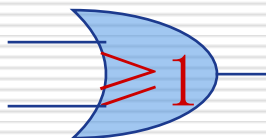


1.3.1 基本逻辑器件概述

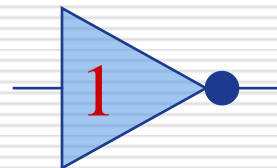
(1) 基本逻辑门



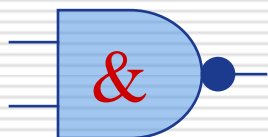
与门



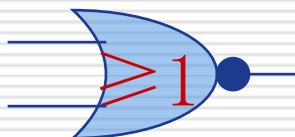
或门



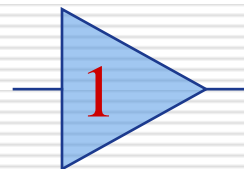
非门



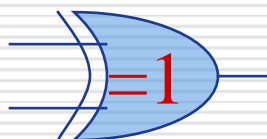
与非门



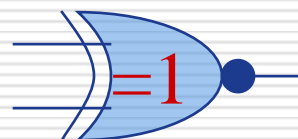
或非门



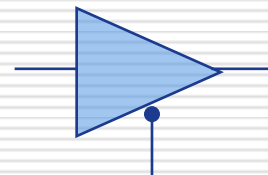
缓冲门



异或门



同或门



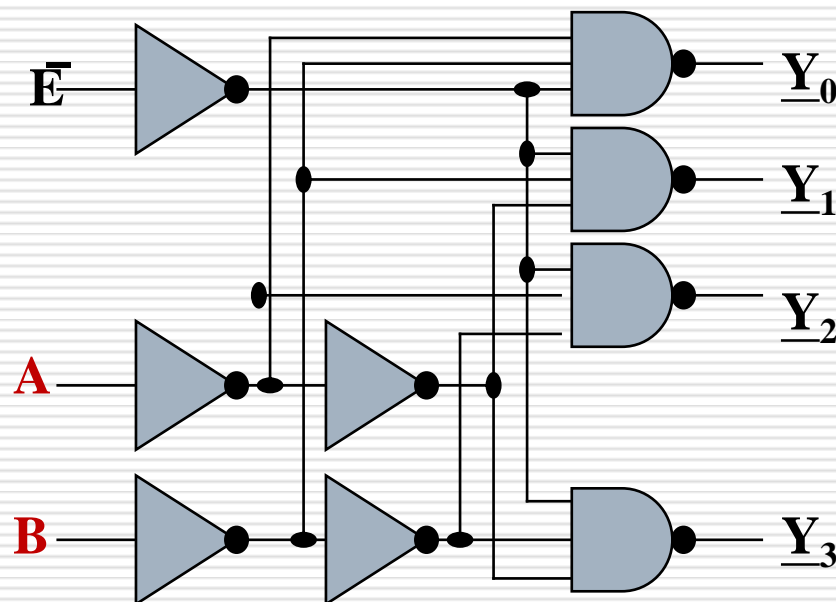
三态门

(2) 译码器

□ 有 n 个输入变量， 2^n 个输出。当输入为某一状态时，仅有一个对应的输出有效，而其余输出无效。

功能表

\bar{E}	A	B	\underline{Y}_0	\underline{Y}_1	\underline{Y}_2	\underline{Y}_3
0	0	0	0	1	1	1
0	1	0	1	0	1	1
0	0	1	1	1	0	1
0	1	1	1	1	1	0
1	X	X	1	1	1	1

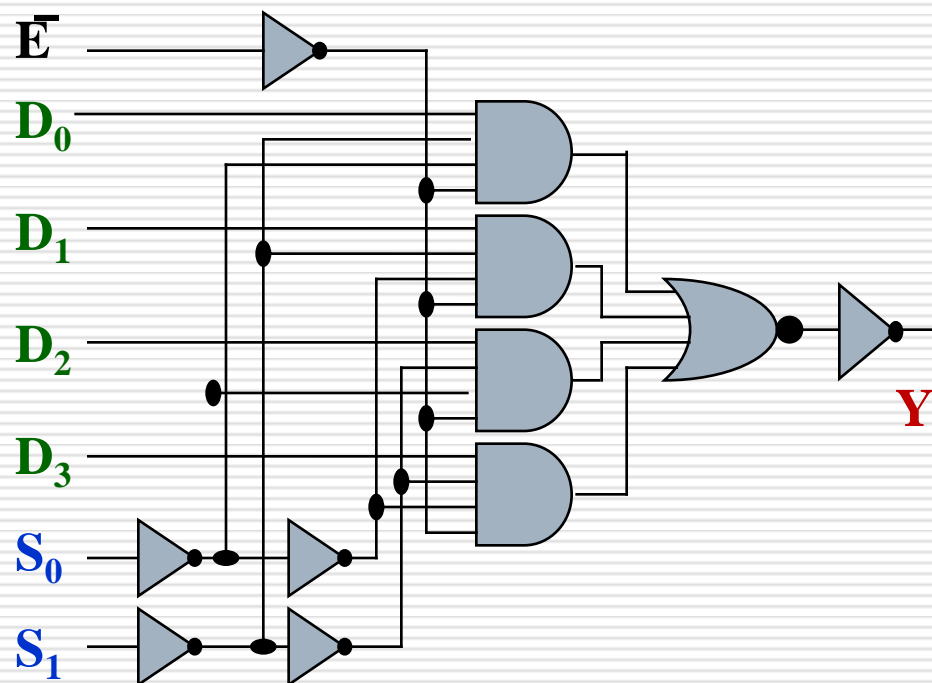


(3) 多路开关（数据选择器）

- 从多个输入通道中选择某一个通道的数据作为输出。
- 例四通道选一

功能表

\bar{E}	S_1	S_0	Y
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3
1	X	X	0

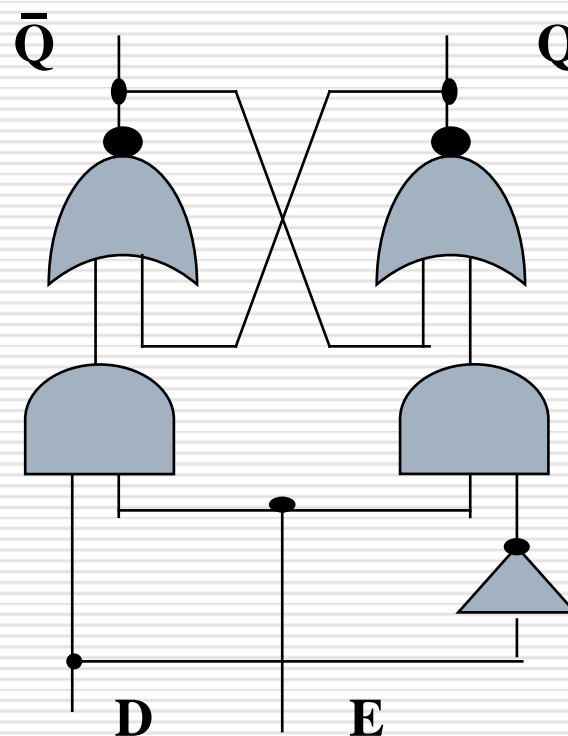
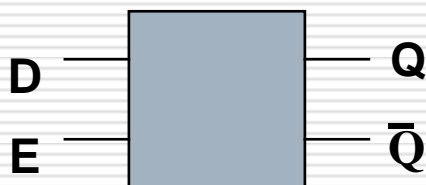


(4) 电位触发 触发器

- 接收满足一定条件的输入数据，具有记忆功能，时序逻辑电路。

功能表

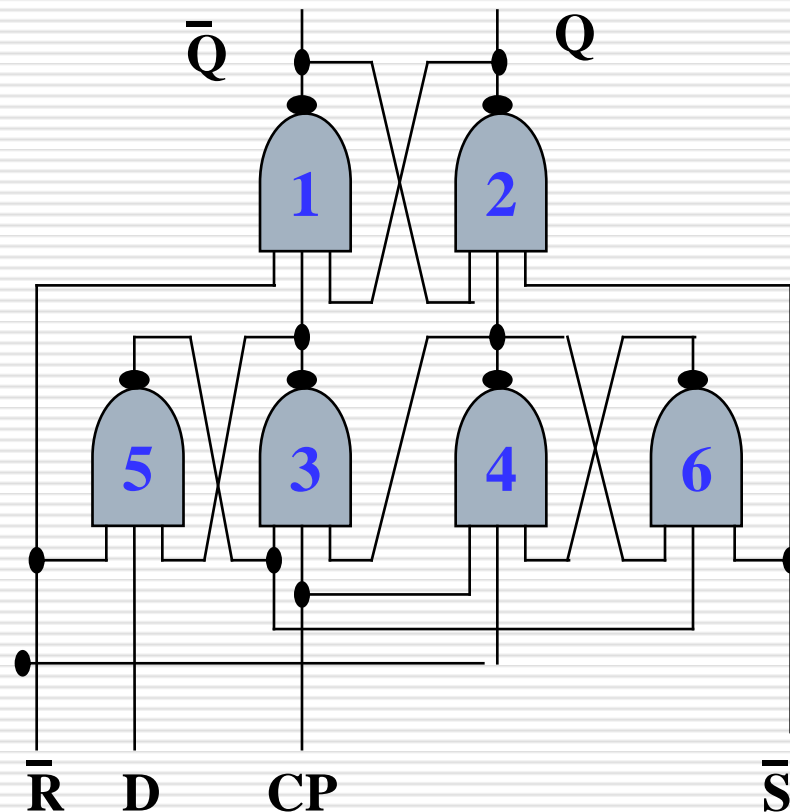
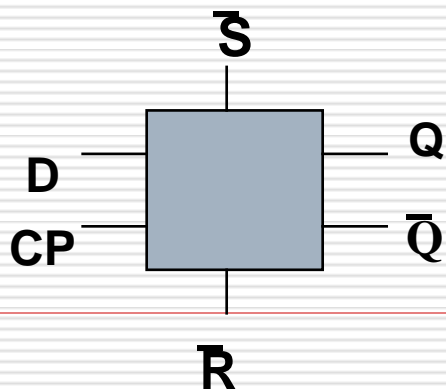
E	D	Q	\bar{Q}
0	X	Q_0	\bar{Q}_0
1	0	0	1
1	1	1	0



(5) D触发器 (边沿触发)

功能表

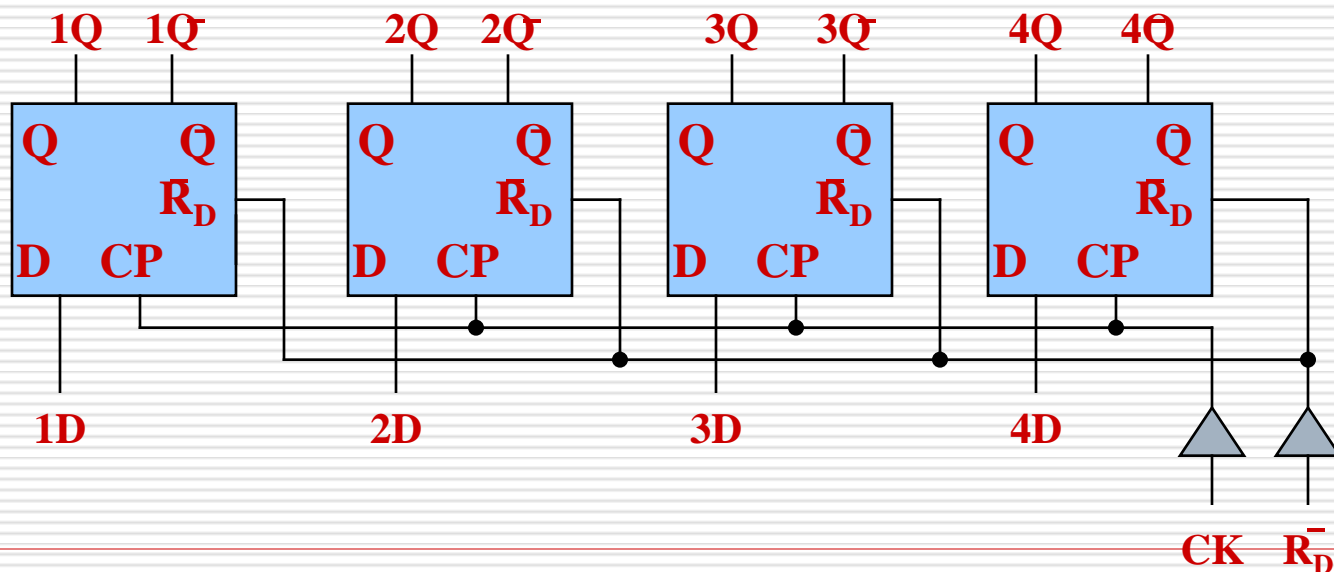
\bar{R}	\bar{S}	CP	D	Q	\bar{Q}
0	1	X	X	0	1
1	0	X	X	1	0
1	1	\uparrow	0	0	1
1	1	\uparrow	1	1	0



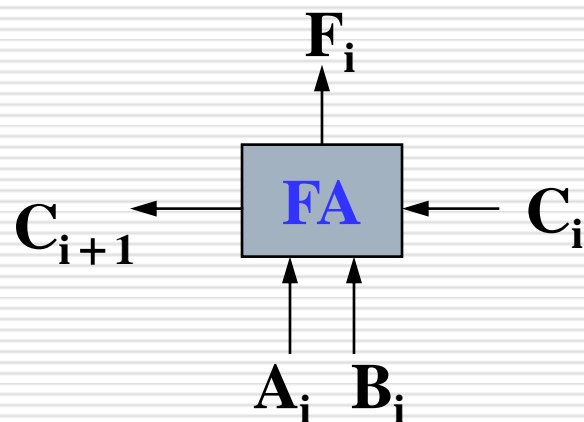
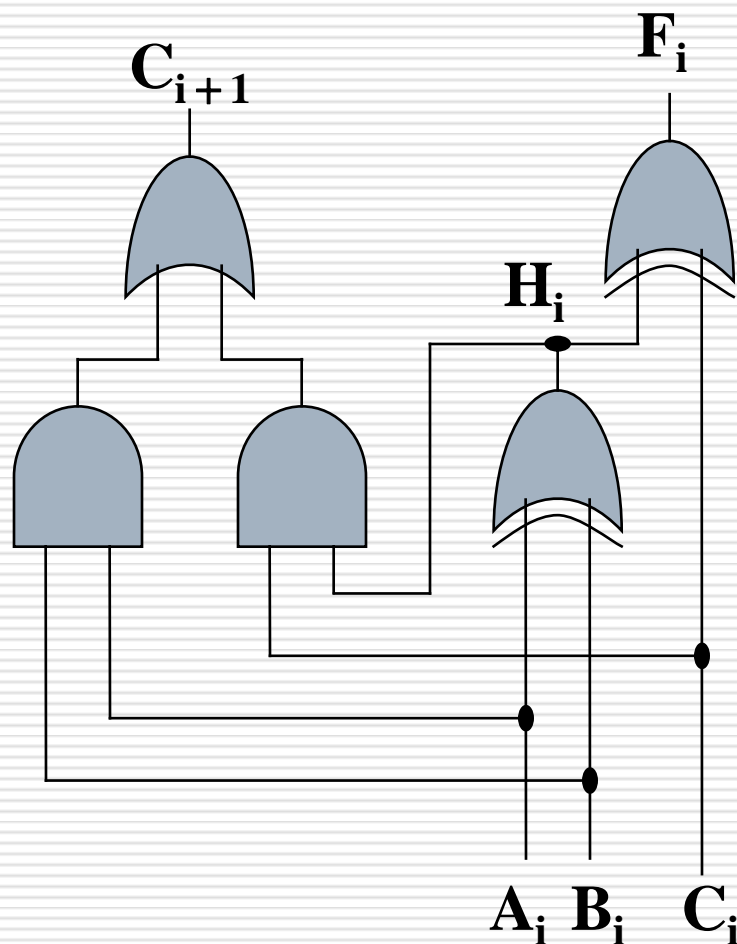
(6) 寄存器

用于暂存数据和指令。采用正边沿触发D触发器和锁存器构成。

\bar{R}_D	CK	1Q	2Q	3Q	4Q
1	↑	1D	2D	3D	4D
0	X	0	0	0	0



(7) 全加器

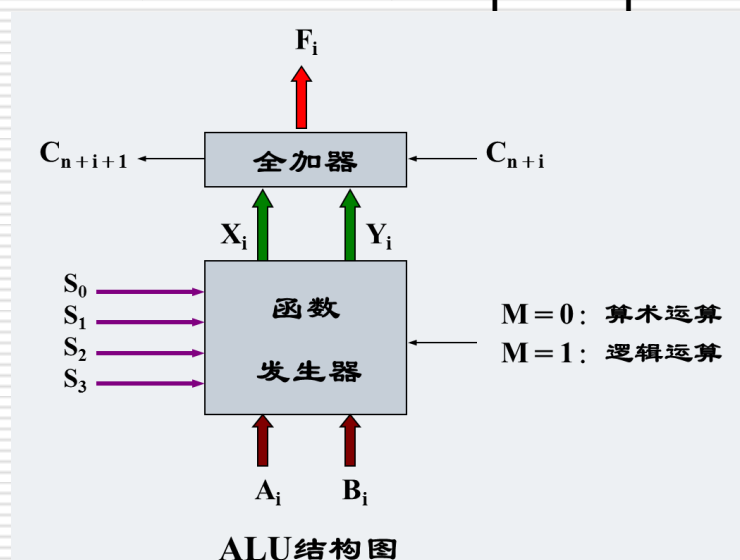


$$F_i = A_i + B_i + C_i$$

进位 C_{i+1}

(8) 多功能逻辑部件

- 目的：既实现算术操作，又能进行逻辑运算。
- 方法：将参与运算的二进制数 A_i 、 B_i 先由控制参数进行控制，产生函数 X_i 、 Y_i ，再送至全加器运算。



- 特点：同样的 A_i 、 B_i 经由不同的控制组合，形成不同的 X_i 、 Y_i ，实现不同的运算。

1.3.2 PLD器件原理概述

PLD器件分类表（按集成度）

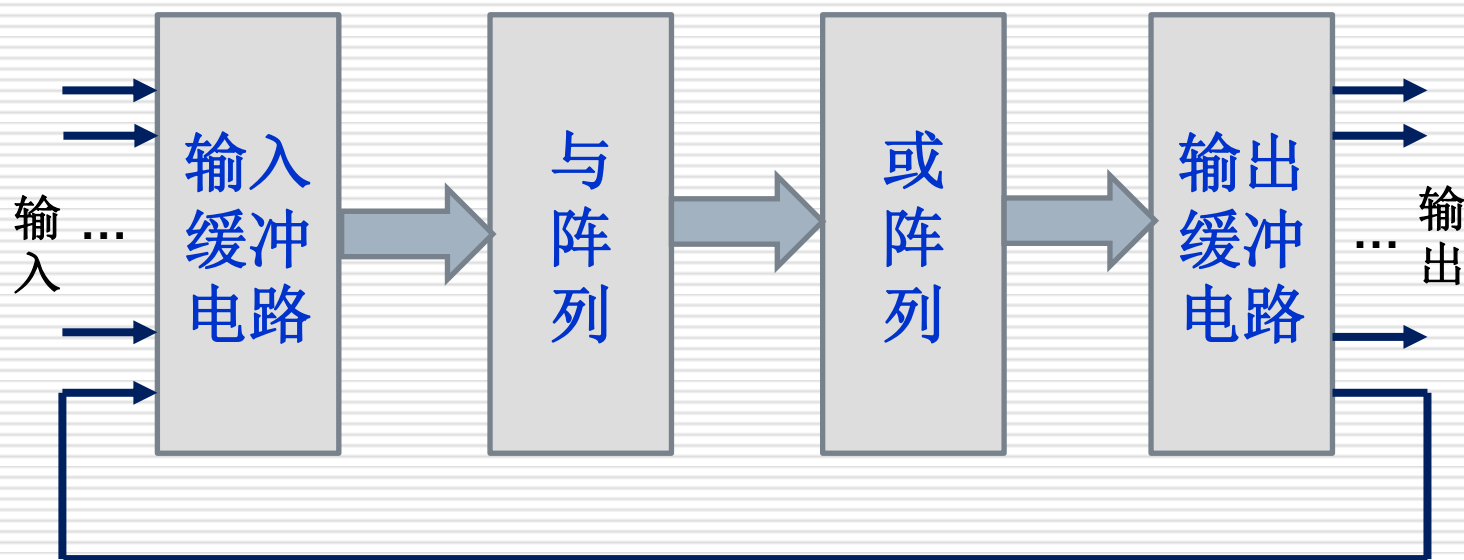
PLD 器件	简单PLD SPLD	可编程只读存储器 PROM (Programmable Read Only Memory)	
		可编程逻辑阵列 PLA (Programmable Logic Array)	基本淘汰
		可编程阵列逻辑 PAL (Programmable Array Logic)	可由 GAL 完全替代
	高密度PLD HDPLD	通用阵列逻辑 GAL (Generic Array Logic)	结构简单，价格便宜、使用方便
		复杂可编程逻辑器件 CPLD (Complex Programmable Logic Device) 现场可编程逻辑阵列 FPGA (Field Programmable Gate Array)	当前 PLD 器件的主流

1.3.2 PLD基本原理

① 基本原理

- 任何组合逻辑函数均可化为“与或”表达式，用“与门-或门”二级电路；
- 任何时序电路都可由组合电路加上存储元件（触发器）构成；
- 与或阵列加上触发器即可以实现任意的数字电路。
- PLD采用上述结构，再加上灵活配置的互连线，从而实现任意的逻辑功能。

② 基本结构



PLD基本结构

1.3.3 PLD器件结构

① 实现机理

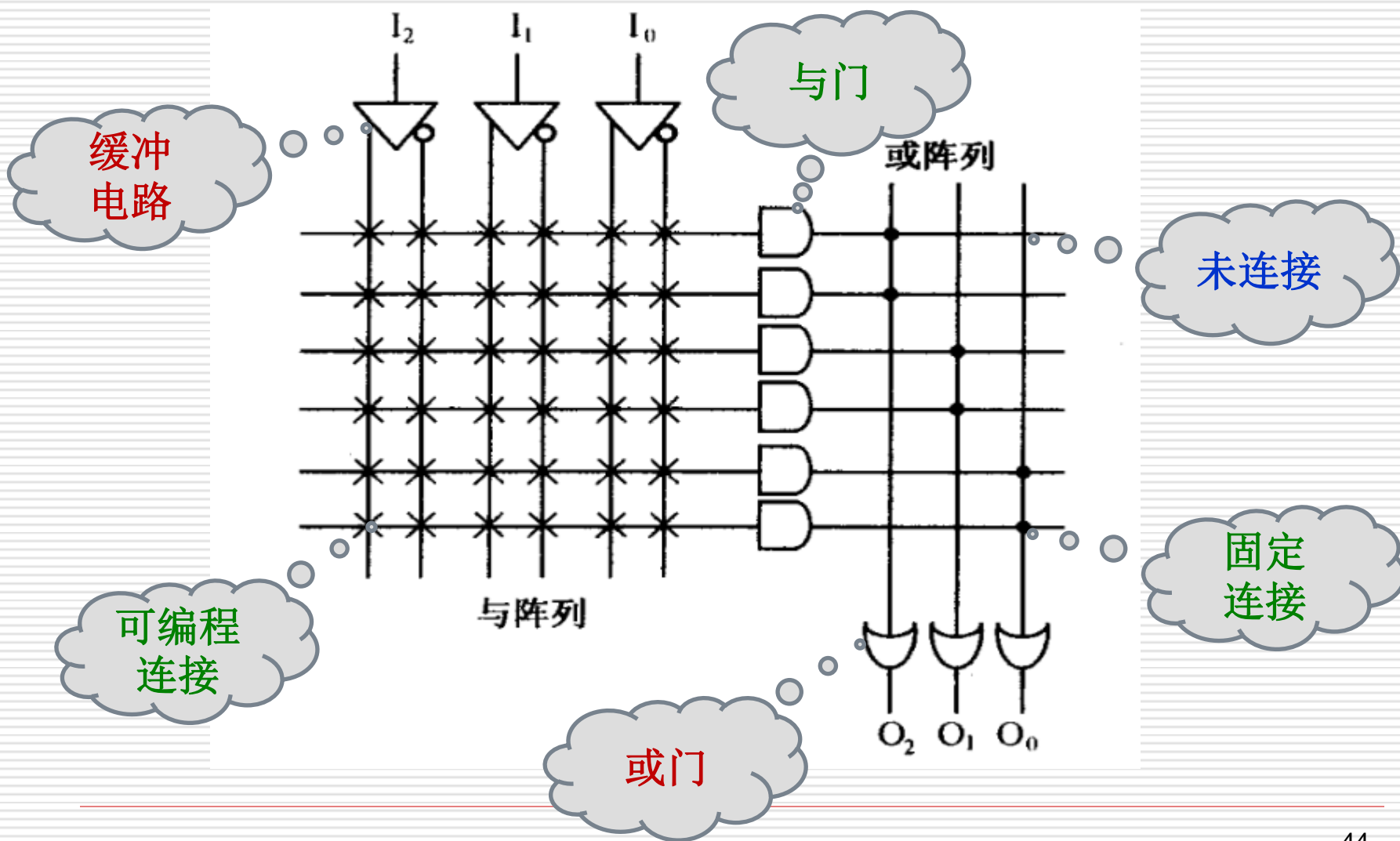
- ❖ 查表：早期**PROM**、**EPROM**和**EEPROM**只能完成简单的数字逻辑功能。

$$Y = F(x)$$

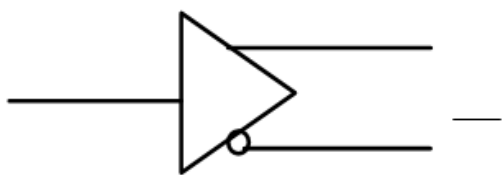
地址作为输入**x**，单元存储内容作为输出**Y**

- ❖ 乘积项：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“与一或”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。

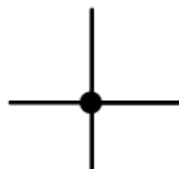
简单逻辑阵列图



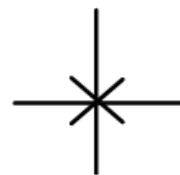
PLD 阵列中的符号表示



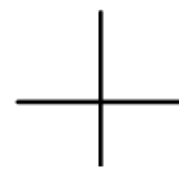
PLD 缓冲电路



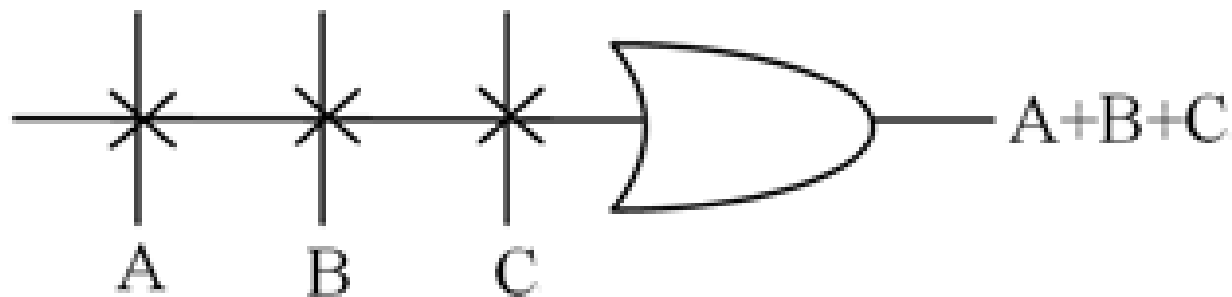
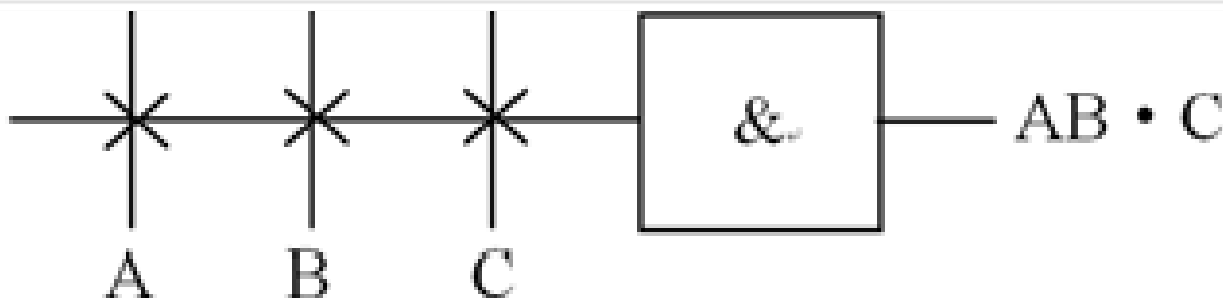
固定连接



可编程连接



未连接



② 编程特点

- 熔丝（Fuse）和反熔丝（Antifuse）编程器件，
 - 可擦除的可编程只读存储器（EPROM）编程器件
 - 电信号可擦除的可编程只读存储器（EEPROM）编程器件（如：CPLD）
 - SRAM编程器件（如：FPGA）。
- 前3类为非易失性器件，编程后配置数据保留在器件上；第4类为易失性器件，掉电后配置数据会丢失，因此在每次上电后需要重新进行数据配置。
- 熔丝结构

以“与/或”阵列为基础PLD类型

- 可编程只读存储器PROM
(Programmable Read Only Memory) ;
- 可编程逻辑阵列FPLA
(Programmable Logic Array) ;
- 可编程阵列逻辑PAL
(Programmable Array Logic) ;
- 通用阵列逻辑GAL
(Generic Array Logic)

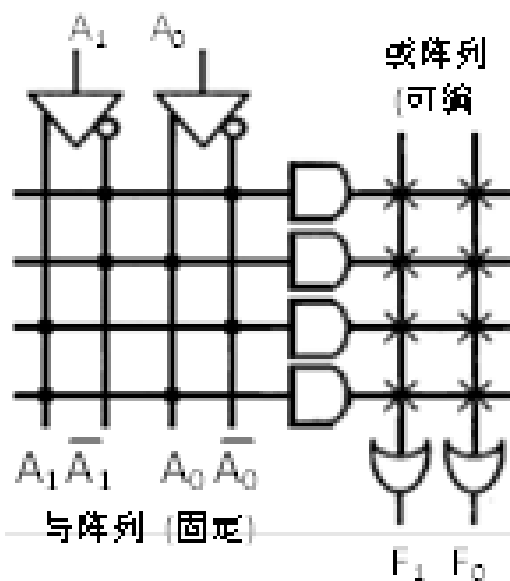
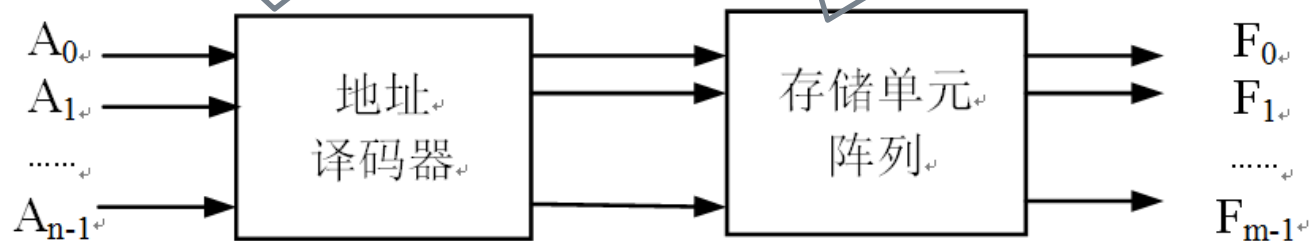
特点:

速度特性较好；结构过于简单；实现电路规模较小。

PROM结构

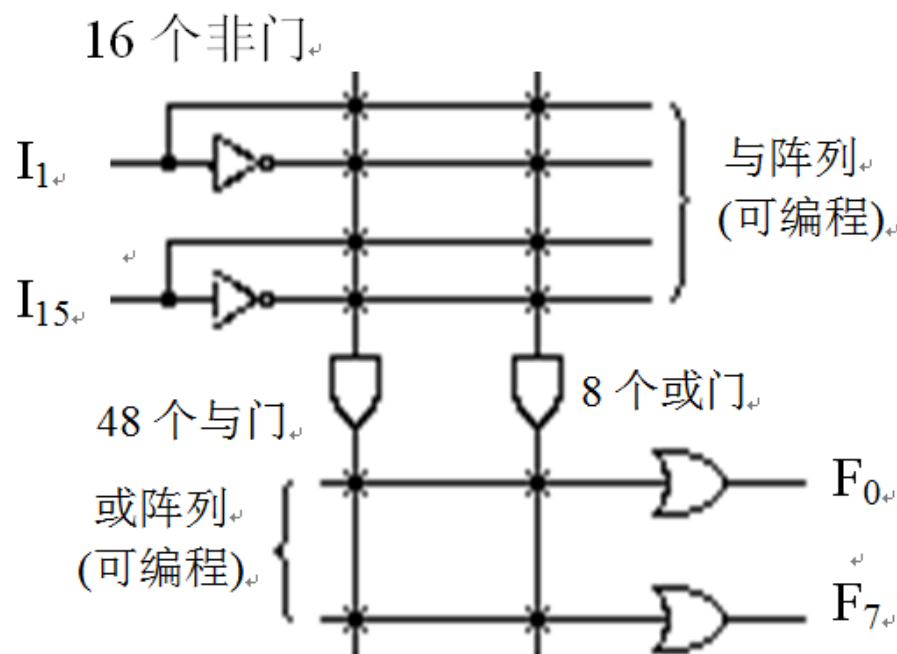
可视为一个连接
固定的与阵列

可视为一个可编
程的或阵列



可编程逻辑阵列PLA结构

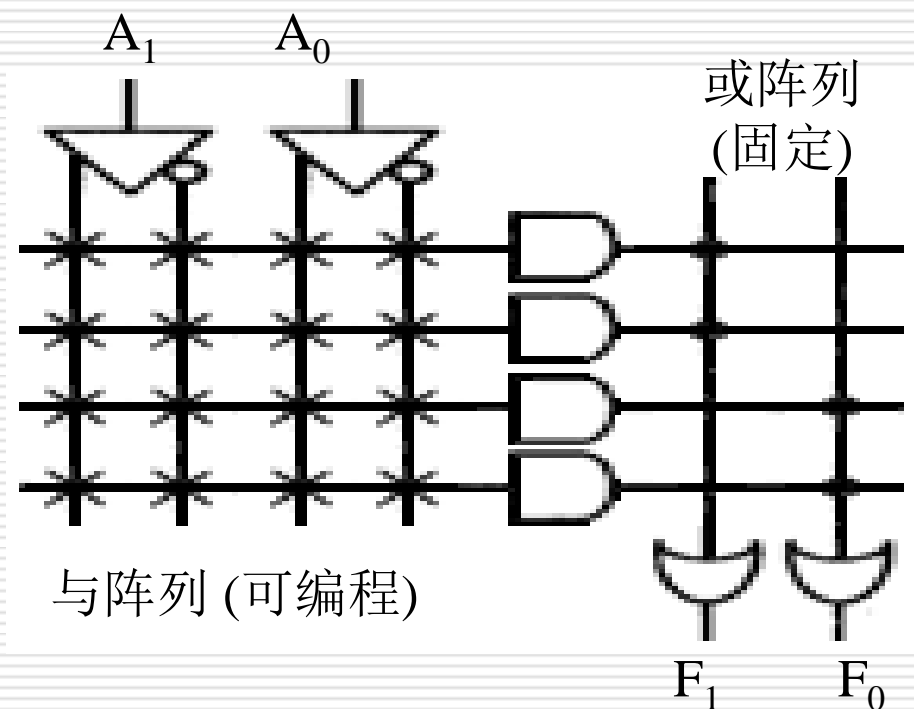
□ 由可编程的与阵列和可编程的或阵列构成



特点：提高了芯片的利用率，节省芯片面积；但对开发软件的要求高，优化算法复杂；构造出的器件运行速度低。

可编程阵列逻辑PAL结构

□ 与阵列是可编程的，或阵列是固定的。



特点：使得送到或门的乘积项的数目是固定的，大大地简化了设计算法。

③高密度PLD

□特点：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。

□类型：

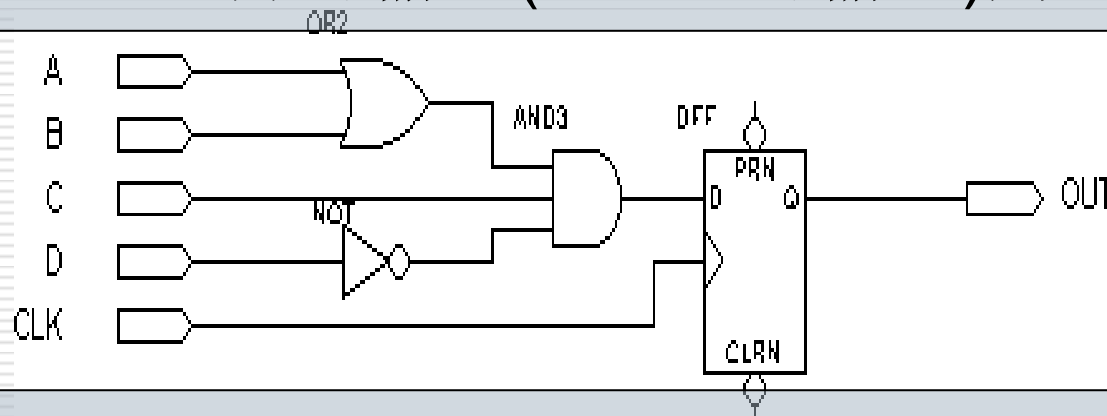
➤基于乘积项（**Product-Term**）结构：主要结构是与或阵列，包含一个或多个与或阵列采用E²PROM或Flash工艺，具有非易失性，器件容量小于5000门。

➤基于查找表（**look Up Table**）结构：物理结构基于SRAM和数据选择器MUX，通过查表方式实现函数功能。集成度高（可实现百万门以上的设计规模），速度快，逻辑功能强，需外挂EEPROM。

➤Xilinx公司把基于查找表技术的SRAM工艺，要外挂配置用的EEPROM的PLD叫FPGA；把基于乘积项技术的Flash（类似EEPROM工艺）工艺的PLD叫CPLD。

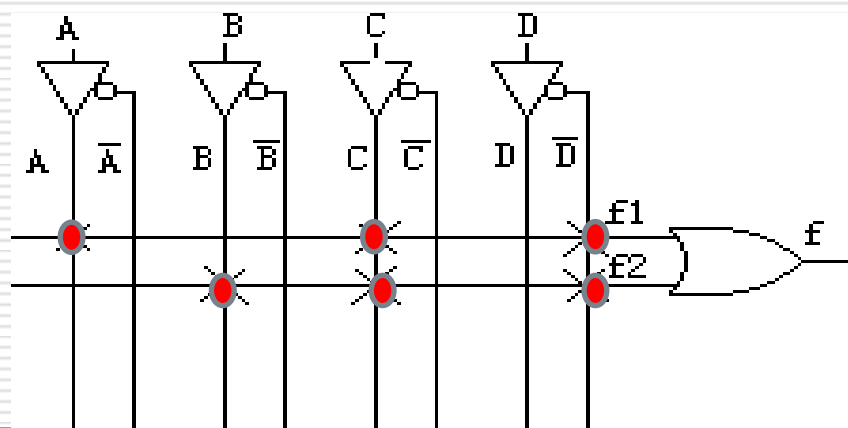
PLD应用实例:

假设组合逻辑的输出(AND3的输出)为f,



$$\text{则 } f = (A+B) * C * (!D) = \underline{A * C * !D} + \underline{B * C * !D}$$

PLD实现方式



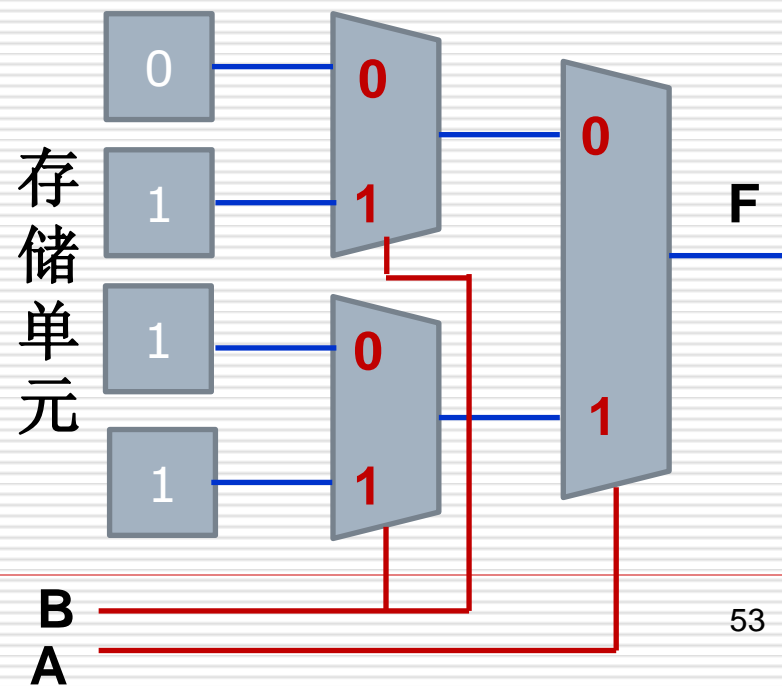
1.3.4 FPGA基本原理与结构

□ 查找表LUT(Look-Up-Table)原理:

物理结构基于SRAM。N个输入项的逻辑函数可由一个 2^N 位容量的SRAM实现，函数值存放在SRAM中，地址线起输入线作用，地址即输入变量值；SRAM的输出为逻辑函数值，由连线开关实现与其他功能块的连接。

2输入
或门

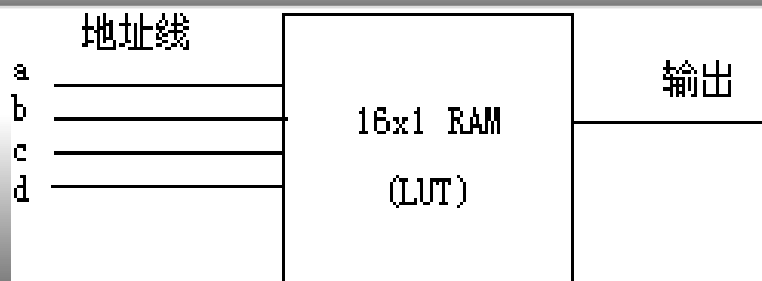
真值表		
输入		输出
A	B	F
0	0	0
0	1	1
1	0	1
1	1	1



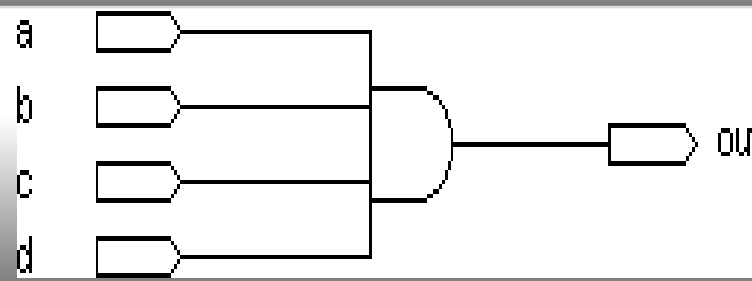
查找表LUT原理与实现

4输入与门的查找表

实际逻辑电路

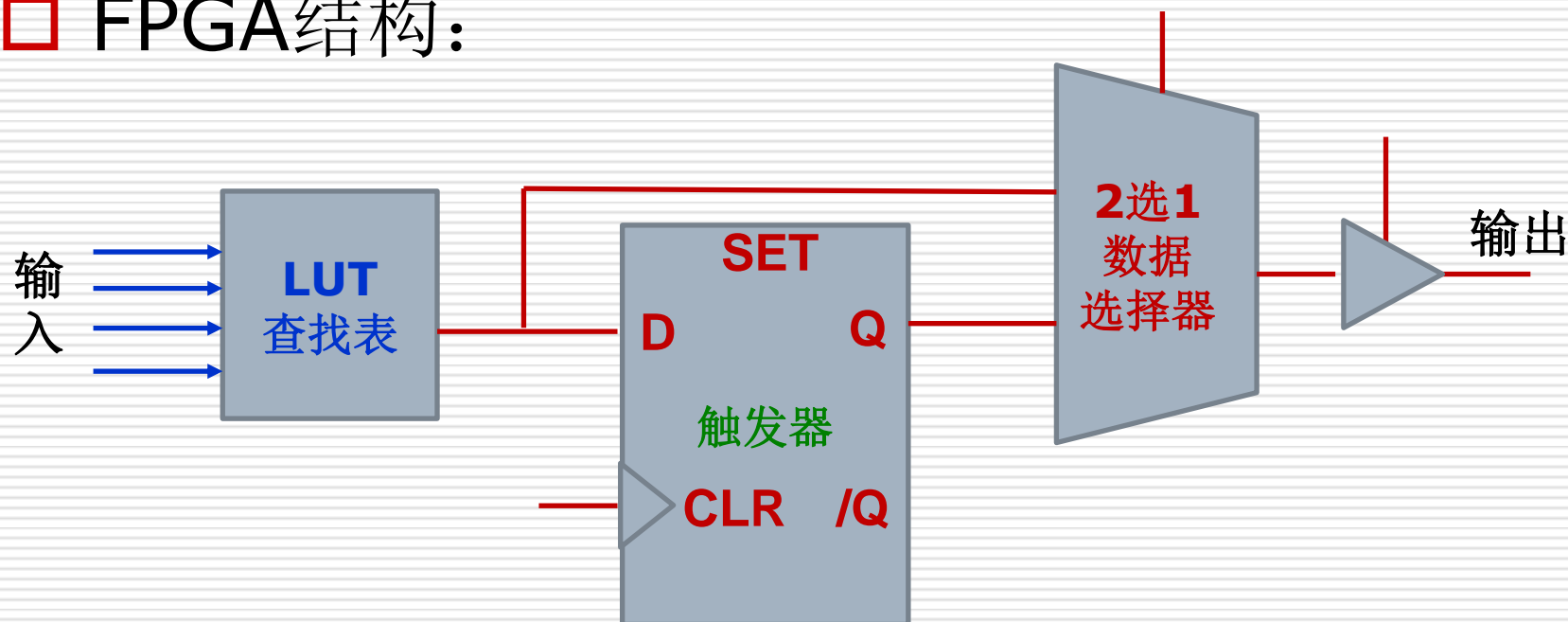


LUT的实现功能



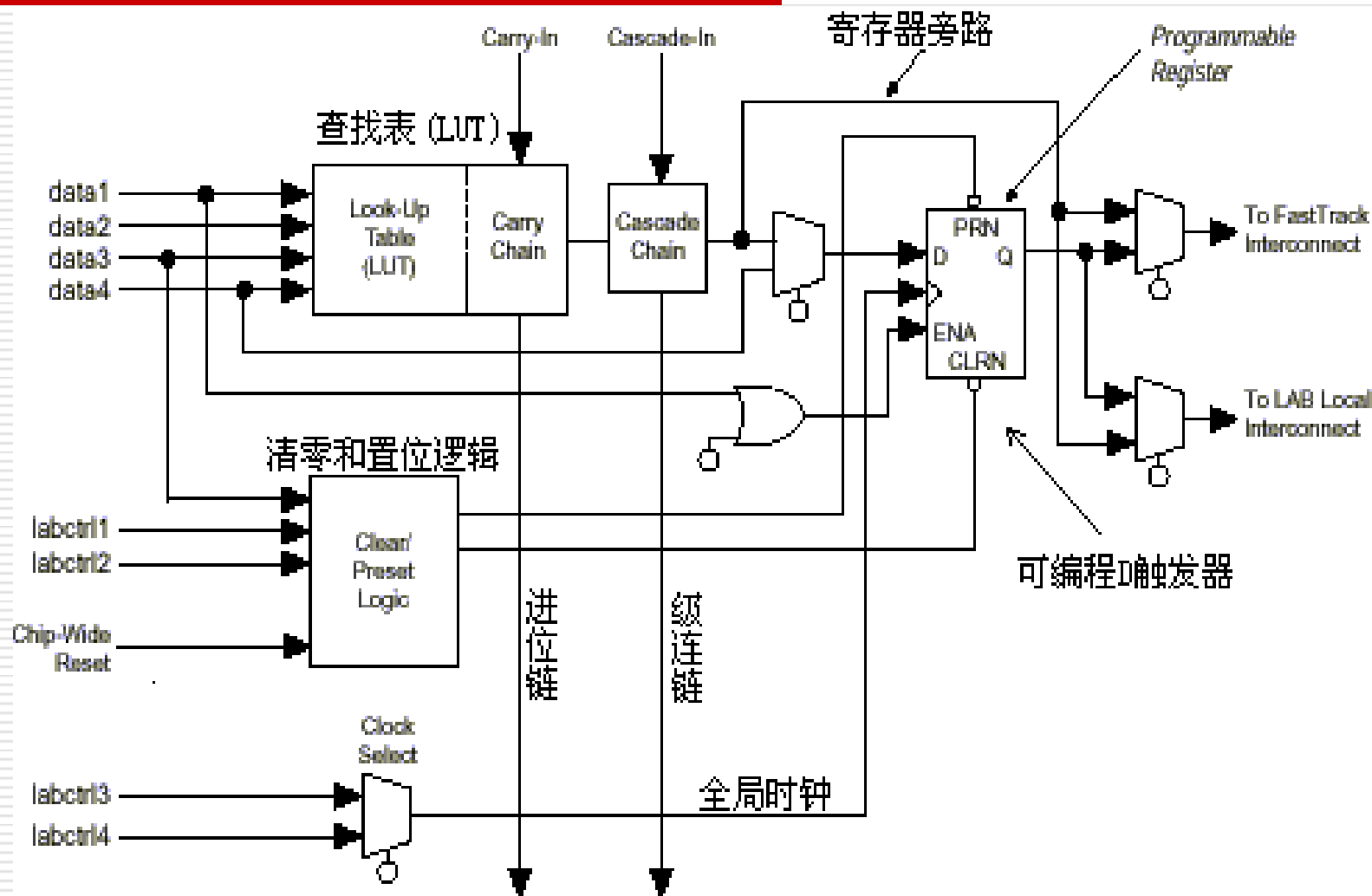
a,b,c,d 输入	逻辑输出	地址	RAM中存储的内容
0000	0	0000	0
0001	0	0001	0
....	0	...	0
1111	1	1111	1

□ FPGA结构:



在**FPGA**的逻辑块中，除**LUT**外一般还包含触发器用于存储**LUT**的输出值，实现时序逻辑电路。

基于查找表(LUT)的FPGA的结构



1.3.5 Xilinx公司芯片简介

(1) Xilinx公司器件系列

□ CPLD产品:

- ① CoolRunner系列; 含CoolRunner-II和CoolRunner XPLA3两个系列
- ② XC9500系列; 含XC9500XL和XC9500两个系列。

□ FPGA产品:

- ① Spartan系列
- ② Virtex系列

(2) Xilinx公司器件-FPGA

➤ Xilinx公司目前有两大类FPGA产品：

① Spartan系列

主要面向低成本的中低端应用，是目前业界成本最低的一类FPGA；

② Virtex系列

主要面向高端应用，属于业界的顶级产品。

➤ 这两个系列的差异仅限于芯片的规模和专用模块上，都采用了先进的0.13um、90nm 甚至65nm 制造工艺，具有相同的卓越品质。

(3) Xilinx公司FPGA器件-Spartan-6系列

- 于2009年推出的新一代的FPGA芯片；
- 该系列的芯片功耗低，容量大。逻辑单元的容量从3,400-148,000，但功耗只有以前Spartan芯片的一半，并且有更快的，更复杂的连接性能。
- 该系列芯片基于45nm的铜处理技术工艺。该系列的芯片提供了6输入的查找表逻辑。

spartan-6系列芯片的主要技术特征

芯片 XC6	逻辑单元	可配置的逻辑块CLB			DSP 48A 1	BRAM		CM T	存储器 控制器 块	PCI-E 端点 模块	最大 DTP 发送 接收 器	I/O 块	最大 I/O
		片 Slice	触发器	最大 分布 RAM		18 KB	最大 (Kb)						
SLX4	3,366	526	4,208	32	4	8	144	1	0	0	0	4	120
SLX9	9,152	1,430	11,440	90	16	32	576	2	2	0	0	4	200
SLX16	14,579	2,278	18,224	136	32	32	576	2	2	0	0	4	232
SLX25	24,051	3,758	30,064	229	38	52	936	2	2	0	0	4	266
SLX45	43,661	6,822	54,576	401	58	116	2,088	4	2	0	0	4	358
SLX100	101,261	15,822	126,576	930	182	268	4,824	6	4	0	0	6	482
5LX150	147,571	23,058	184,464	1,355	182	268	4,824	6	4	0	0	6	498
5LX25T	24,051	3,758	30,064	229	38	52	936	2	2	1	2	4	250
5LX45T	43,661	6,822	54,576	401	58	116	2,088	4	2	1	4	4	296
5LX100T	101,261	15,822	126,576	930	182	268	4,824	6	4	1	8	6	376
5LX150T	147,571	23,058	184,464	1,355	182	268	4,824	6	4	1	8	6	396

1.4 EDA相关技术

1.4.1 IP和复用技术

- **IP (Intellectual Property)**：原意是指知识产权、著作权等，在**IC**设计领域可理解为实现某种功能的设计。
- **IP核(IP模块)**：指完成某种功能的设计模块。
 - 软核：指在寄存器级或门级对电路功能用**HDL**进行描述，表现为**HDL**代码。
 - 硬核：指以版图形式实现的设计模块，它基于一定的设计工艺，通常用**GDSII**表示。
 - 固核：完成了综合的功能块，通常以网表的形式提交给用户使用。

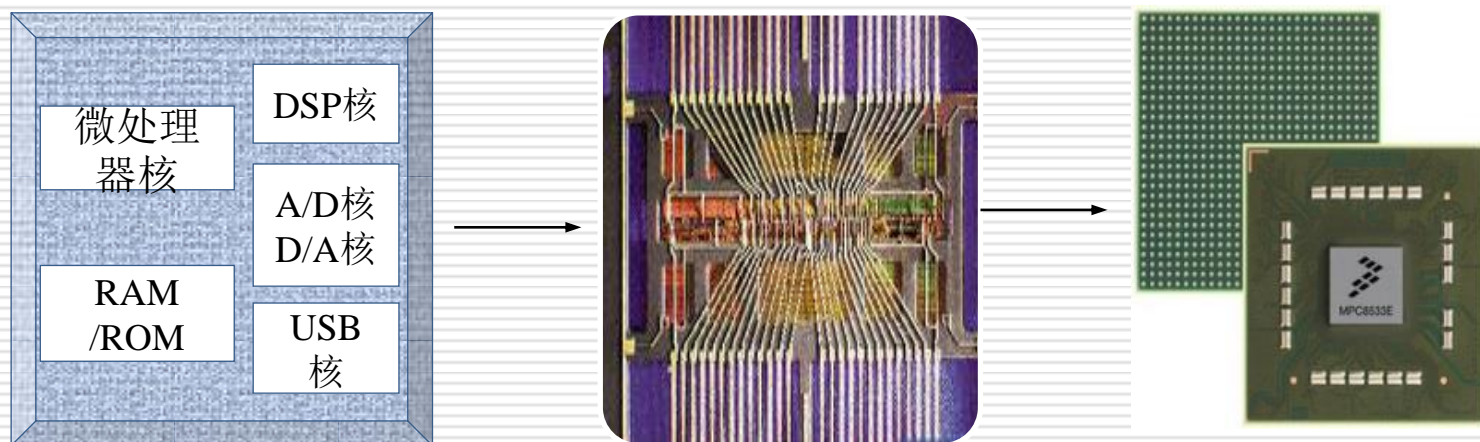


- ❑ **IP复用**：在设计一个系统时，可以自行设计各个功能模块，也可以采用**IP**模块来构建。如此进一步缩短设计周期，提高设计效率。
- ❑ **特点**：基于**IP**复用的开发能够节省成本、缩短开发周期、避免重复劳动、提高设计效率。
- ❑ **注意问题**：如**IP**版权的保护、**IP**的保密级**IP**间的集成等。
- ❑ 基于**IP**复用的设计技术无疑会成为电子系统设计的重要手段之一。

1.4 EDA相关技术

1.4.2 片上系统SoC (System on Chip)

❑ **SOC**: 指在一个芯片上集成微处理器核 (**MCU Core**)、数字信号处理器核 (**DSP Core**)、存储器核 (**RAM/ROM**)、**A/D**核、**D/A**核及**USB**接口核等, 构成一个系统, 即用一个芯片实现一个完整的系统。



1.4 EDA相关技术

1.4.3 边界扫描测试技术

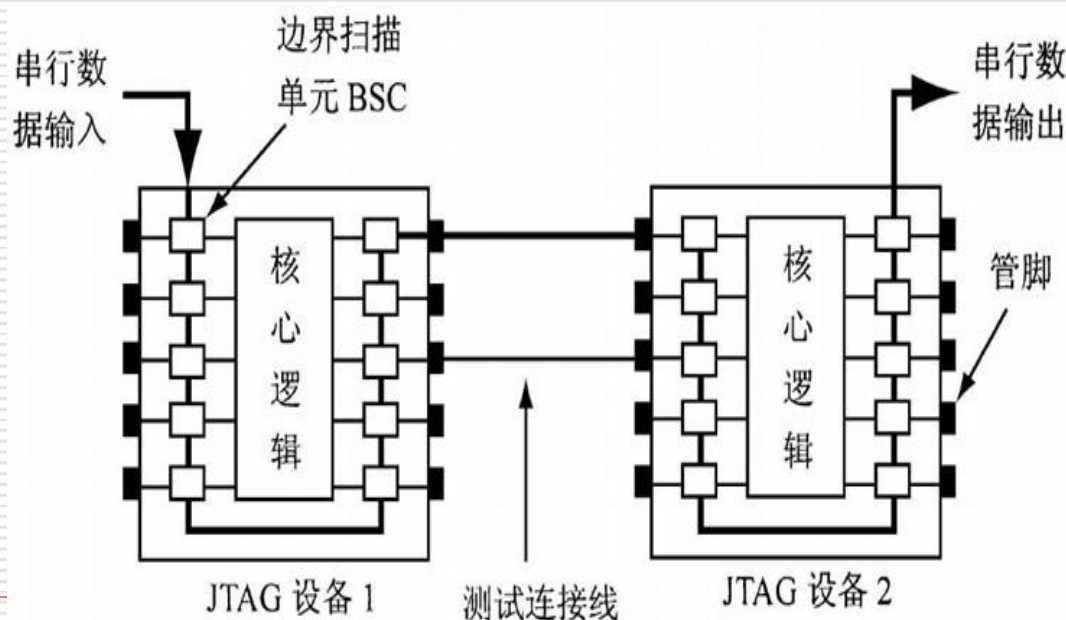
- ❑ JTAG(Joint Test Action Group 联合测试行动小组): 为了解决超大规模集成电路VLSI的测试问题, 1986年, 由IC领域专家所成立
- ❑ BST(Boundary Scan Test)边界扫描测试技术规范: 由JTAG指定的一种国际标准测试协议, 主要用于芯片内部测试及对系统进行仿真、调试。
- ❑ JTAG技术: 是一种嵌入式调试技术, 它在芯片内部封装了专门的测试电路 TAP (Test Access Port , 测试访问口), 通过专用的 JTAG 测试工具对内部节点进行测试, 提供了有效测试高密度引线器件的能力。



1.4 EDA相关技术

JTAG边界扫描结构

提供了一个串行扫描路径，能捕获器件核心逻辑的内容，也能测试遵守JTAG规范的器件之间的引脚连接情况，而且可以在器件正常工作时捕获功能数据。测试数据从左边的一个扫描单元串行移入，捕获的数据从右边的一个边界扫描单元串行移出，然后同标准数据进行比较，从判断芯片性能的好坏。



1.4 EDA相关技术

□ JTAG特点:

- **JTAG**边界扫描测试技术提供了一种合理又有效的方法，用以对高密度、引脚密集的器件和系统进行测试；
- **JTAG**接口还具有编程下载、在线调试等功能；
- 目前高密度数字器件（**CPU**、**DSP**、**ARM**、**FPGA**）几乎都具备标准的**JTAG**接口。

