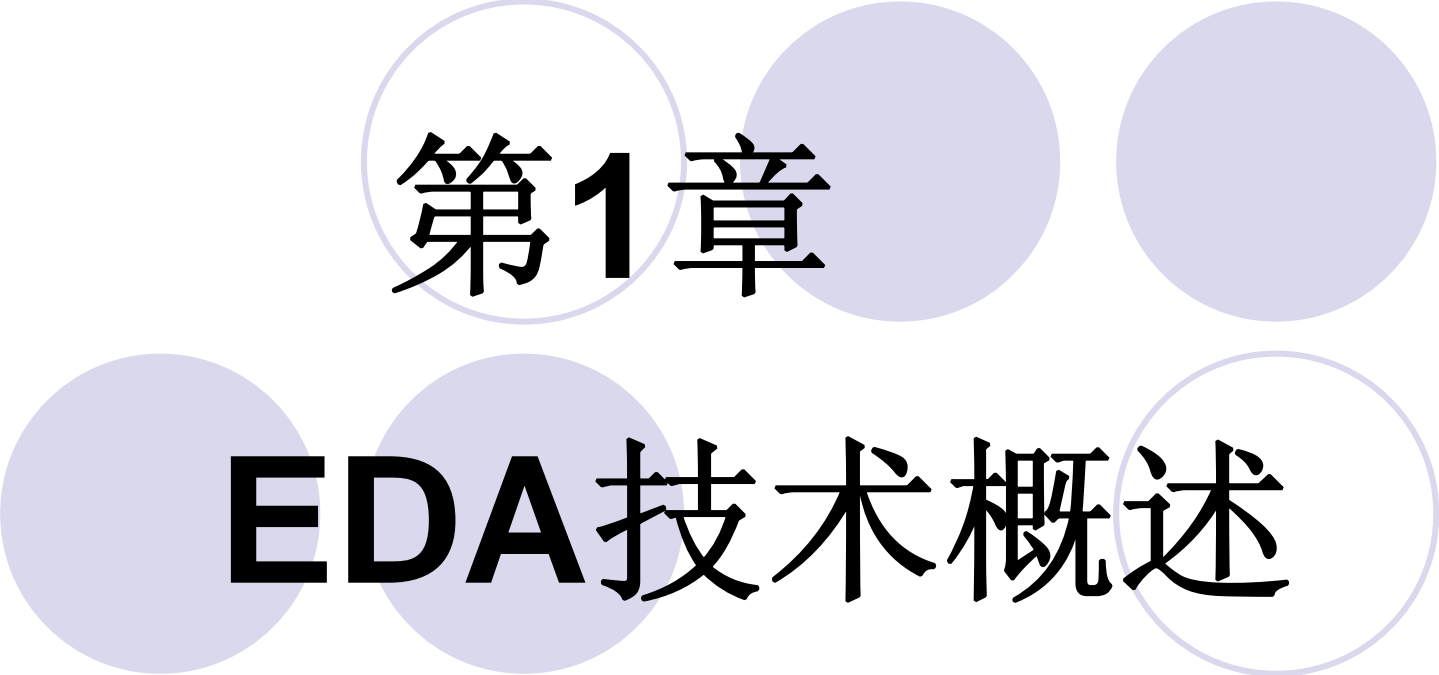




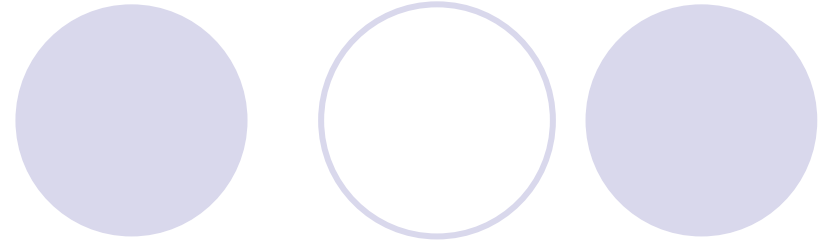
EDA技术与Verilog HDL



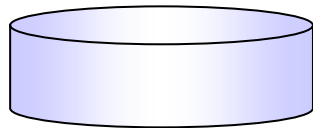
第1章

EDA技术概述

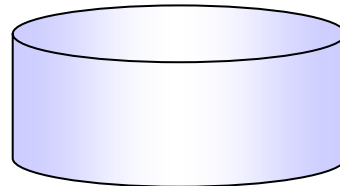
1.1 EDA技术



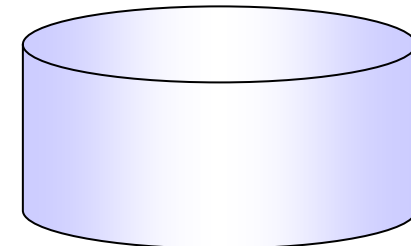
现代电子设计技术的核心已日趋转向基于计算机的电子设计自动化技术 → **EDA (Electronic Design Automation)** 技术。



20世纪70年代
EDA技术雏形




20世纪80年代
EDA技术基础形成



20世纪90年代
EDA技术成熟和实用

1.1 EDA技术



21世纪后

- 在**FPGA**上实现**DSP**应用成为可能。
- 在一单片**FPGA**中实现一个完备的可随意重构的嵌入式系统成为可能。
- 在仿真和设计两方面支持标准硬件描述语言的功能强大的**EDA**软件不断推出。
- 电子领域各学科的界限更加模糊，更互为包容。
- 用于**ASIC**设计的标准单元已涵盖大规模电子系统及复杂**IP**核模块。
- 软硬**IP**核在电子行业的产业领域广泛应用。
- **SoC**高效低成本设计技术的成熟。
- 复杂电子系统的设计和验证趋于简单。

1.2 EDA技术应用对象

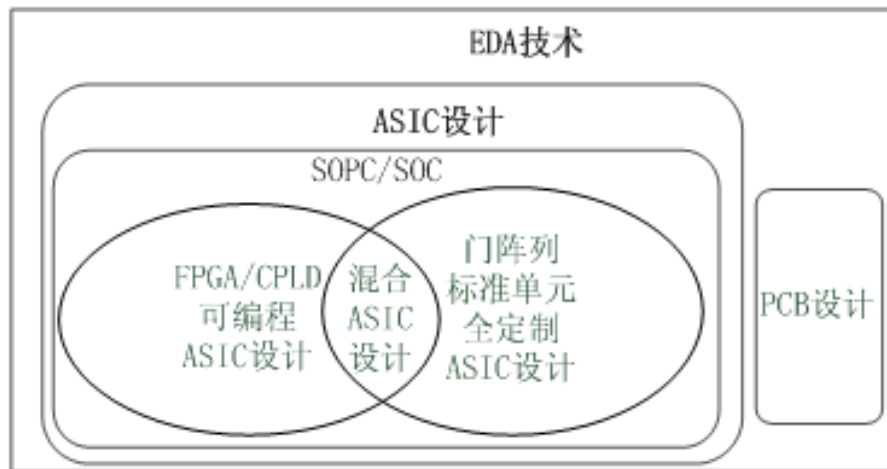


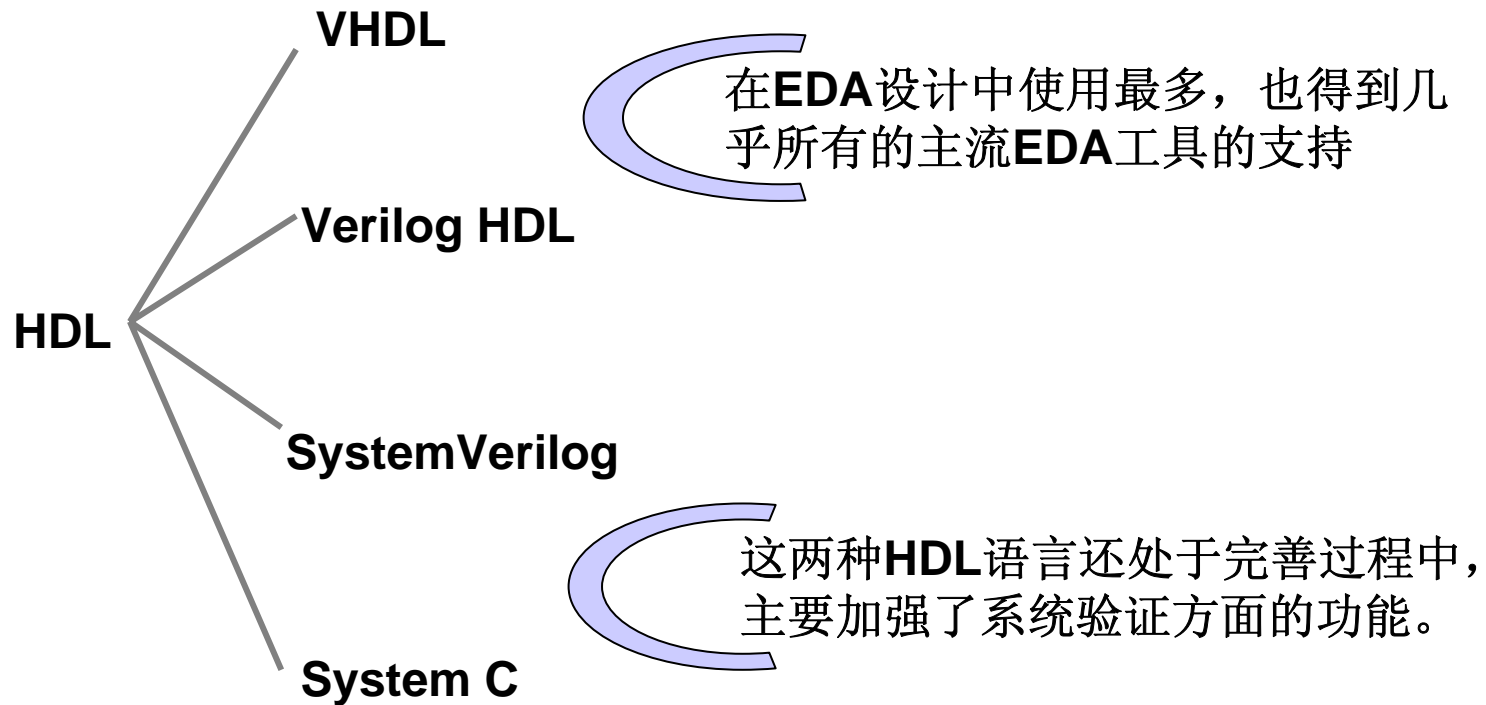
图 1-1 EDA 技术实现目标

1. 可编程逻辑器件

2. 半定制或全定制ASIC

3. 混合ASIC

1.3 硬件描述语言VHDL



1.4 EDA技术的优势

1. 保证设计过程的正确性，大大降低设计成本，缩短设计周期。
2. 有各类库的支持。
3. 极大地简化设计文档的管理。
4. 日益强大的逻辑设计仿真测试技术。
5. 设计者拥有完全的自主权，再无受制于人之虞。
6. 良好的可移植与可测试性，为系统开发提供了可靠的保证。
7. 能将所有设计环节纳入统一的自顶向下的设计方案中。
8. **EDA**不但在整个设计流程上充分利用计算机的自动设计能力，而且在各个设计层次上利用计算机完成不同内容的仿真模拟，在系统板设计结束后仍可利用计算机对硬件系统进行完整的测试。

1.5 面向FPGA的EDA开发流程

1.5.1 设计输入

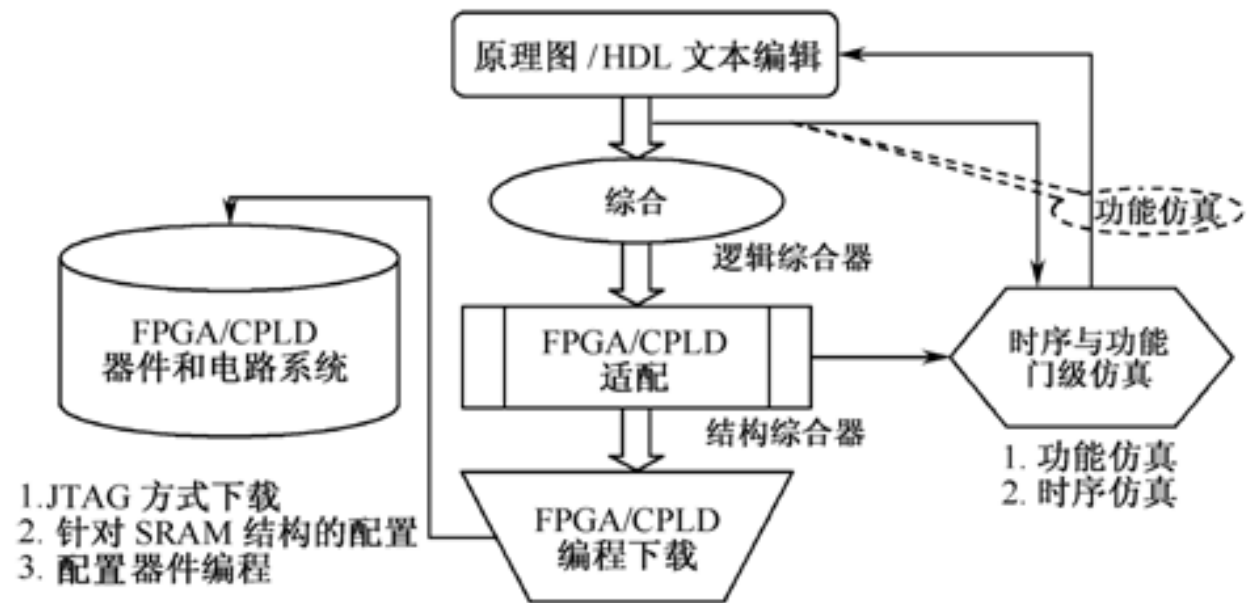


图 1-2 FPGA 的 EDA 开发流程

1. 图形输入

2. 硬件描述语言代码文本输入

1.5 面向FPGA的EDA开发流程

1.5.2 综合

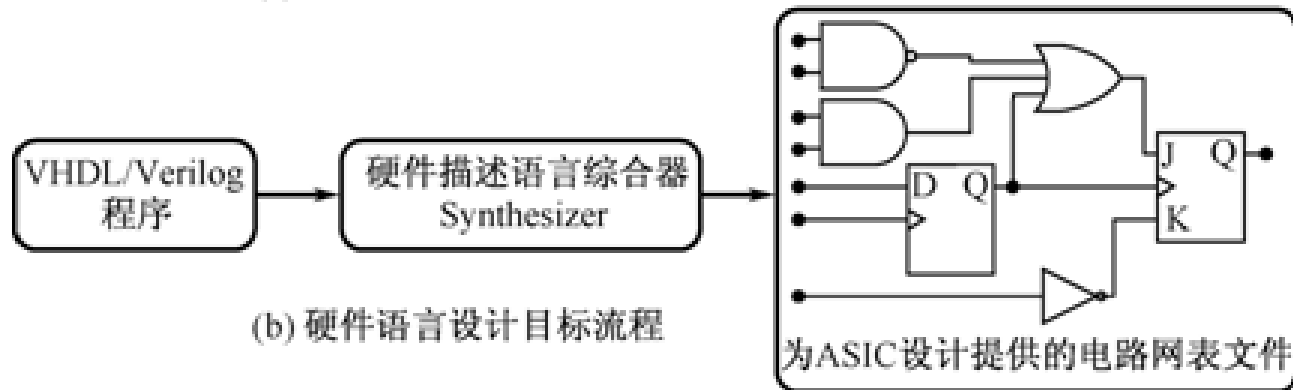
(1)自然语言综合



(a) 软件语言设计目标流程

(2)行为综合

(3)逻辑综合



(b) 硬件语言设计目标流程

(4)版图综合或结构综合

图 1-3 编译器和综合的功能比较

1.5 面向FPGA的EDA开发流程

1.5.2 综合

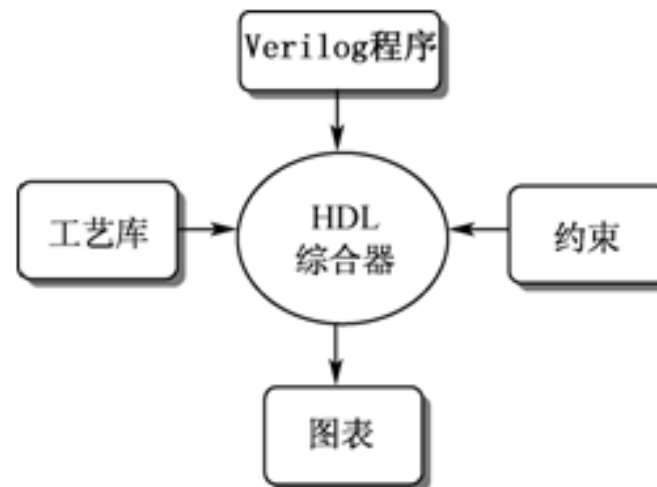


图 1-4 HDL 综合器运行流程

1.5.3 适配（布线布局）

1.5 面向FPGA的EDA开发流程

1.5.4 仿真

(1) 时序仿真

(2) 功能仿真

1.5.5 RTL描述

1.6 可编程逻辑器件

1.6.1 PLD的分类



1.6 可编程逻辑器件

1.6.2 PROM可编程原理

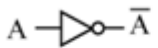



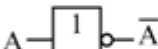
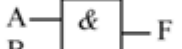
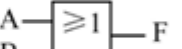
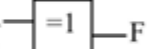
	非门	与门	或门	异或门
IEEE 1991 版 标准逻辑符号				
IEEE 1984 版 标准逻辑符号				
逻辑表达式	$\bar{A} = \text{NOT } A$	$F = A \cdot B$	$F = A + B$	$F = A \oplus B$

图 1-5 两种不同版本的国际标准逻辑门符号对照表

1.6 可编程逻辑器件

1.6.2 PROM可编程原理



图 1-6 PLD 的互补缓冲器

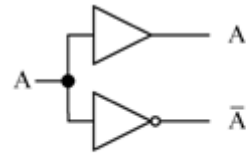


图 1-7 PLD 的互补输入



图 1-8 PLD 中与阵列的表示



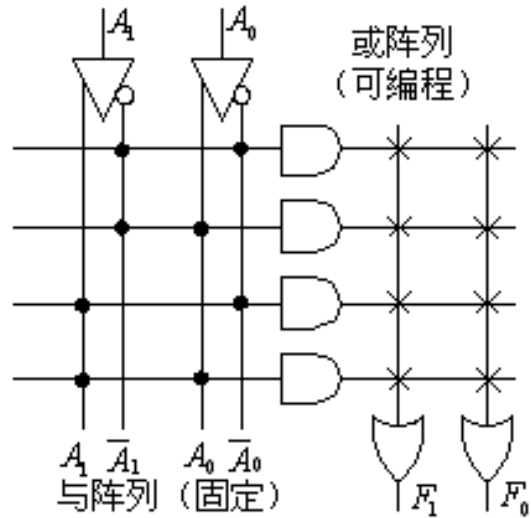
图 1-9 PLD 中或阵列的表示



图 1-10 阵列线连接表示

1.6 可编程逻辑器件

1.6.2 PROM可编程原理

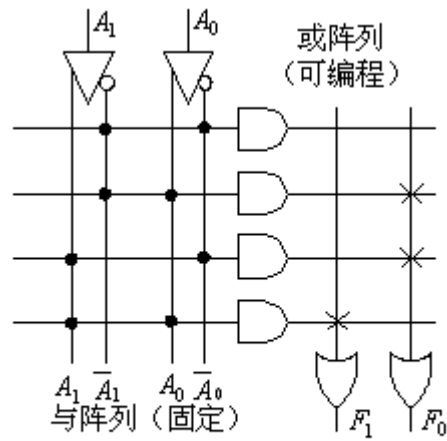


$$S = A_0 \oplus A_1 \quad , \quad C = A_0 \cdot A_1$$

图 1-11 PROM 表达的 PLD 阵列图

1.6 可编程逻辑器件

1.6.2 PROM可编程原理



$$F_0 = A_0 \bar{A}_1 + \bar{A}_0 A_1 \quad , \quad F_1 = A_1 A_0$$

图 1-12 用 PROM 完成半加器逻辑阵列

1.6 可编程逻辑器件

1.6.3 GAL

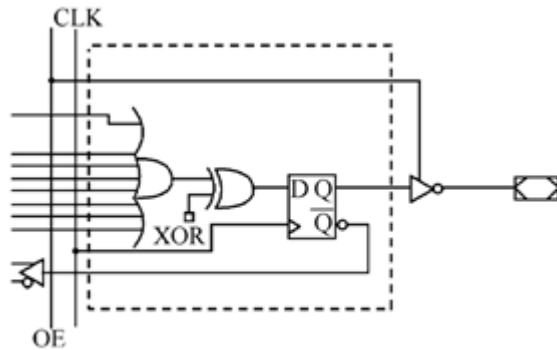


图 1-14 寄存器输出结构

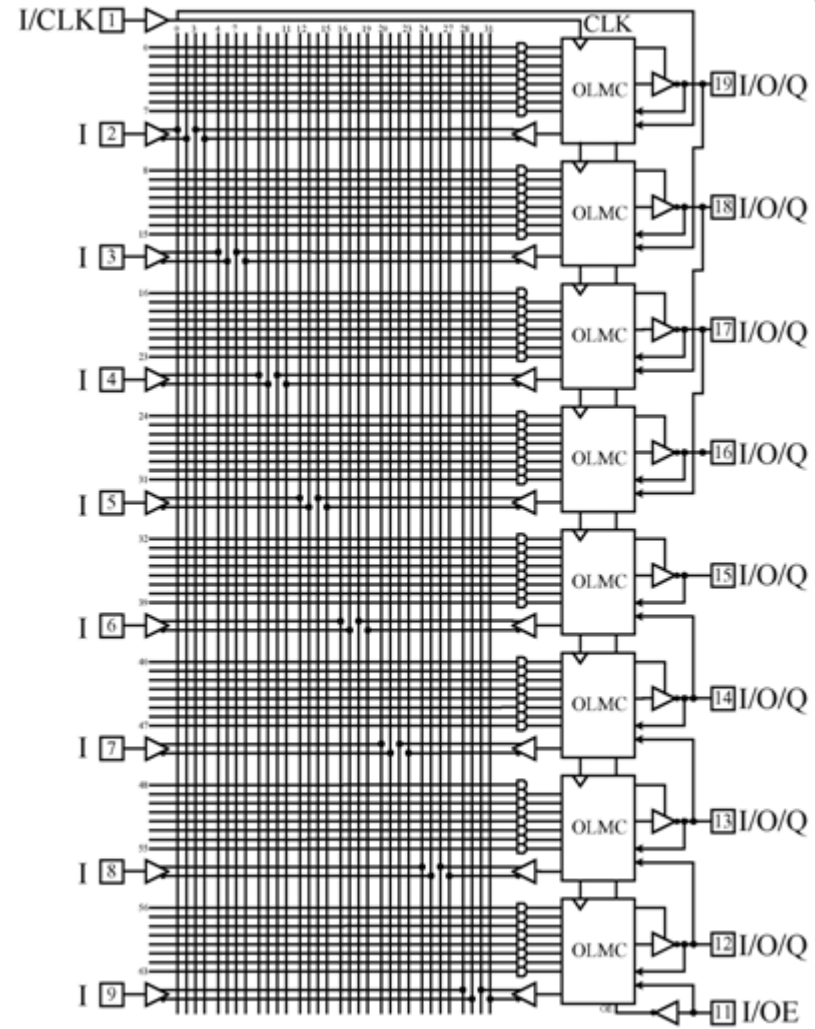


图 1-13 GAL16V8 的结构图

1.7 CPLD的结构与可编程原理

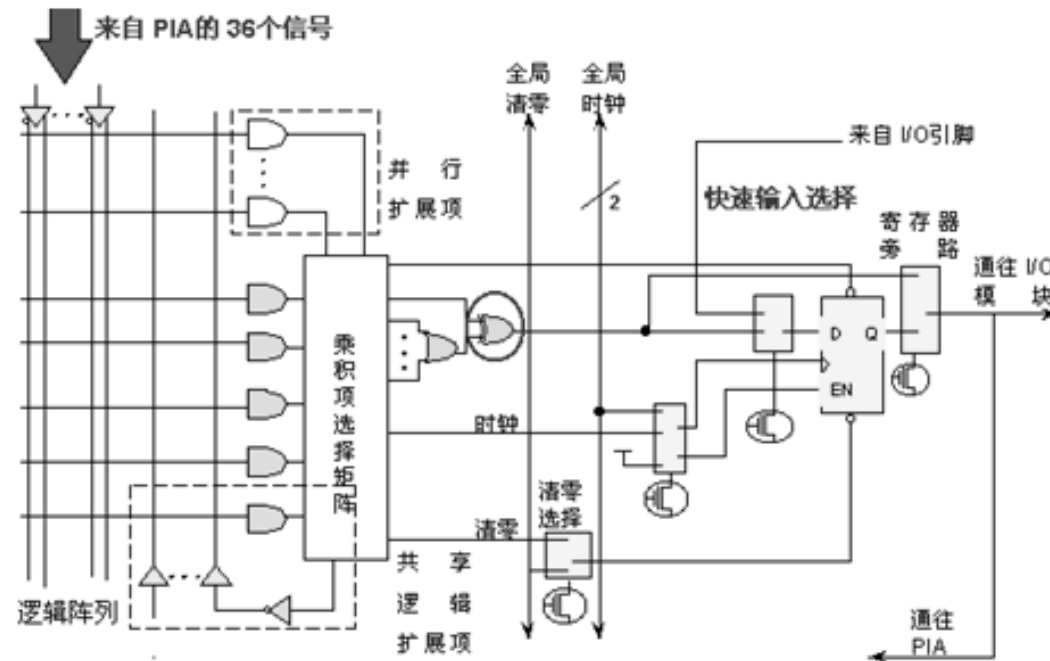


图 1-15 MAX7000 系列的单个逻辑宏单元结构

1.7 CPLD的结构与可编程原理

1. 逻辑阵列块

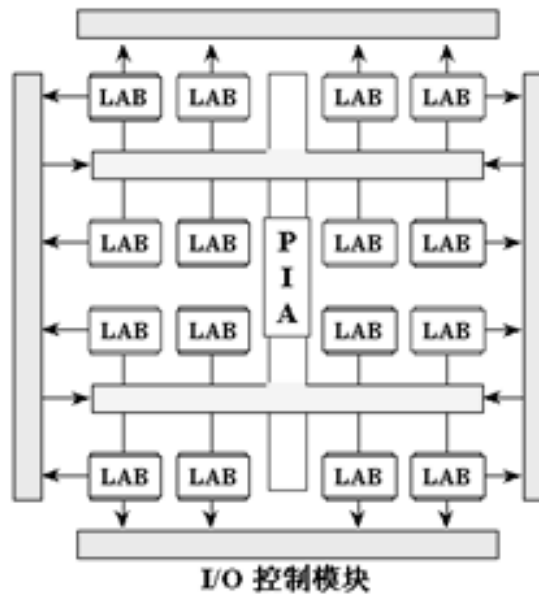


图 1-16 MAX7128S 的结构

1.7 CPLD的结构与可编程原理

2. 逻辑宏单元

3. 可编程连线阵列

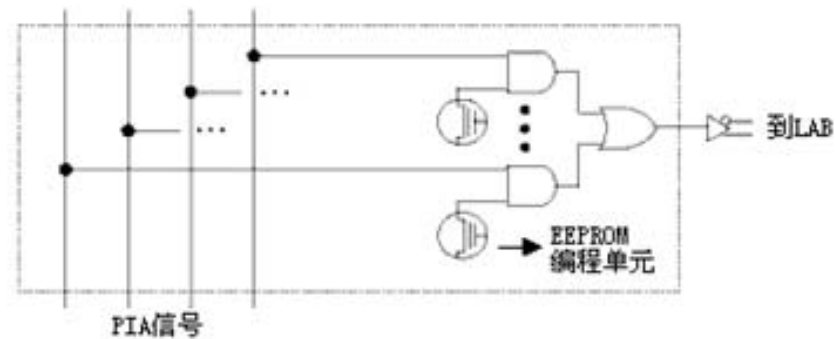


图 1-17 PIA 信号布线到 LAB 的方式

1.7 CPLD的结构与可编程原理

4. I/O控制块

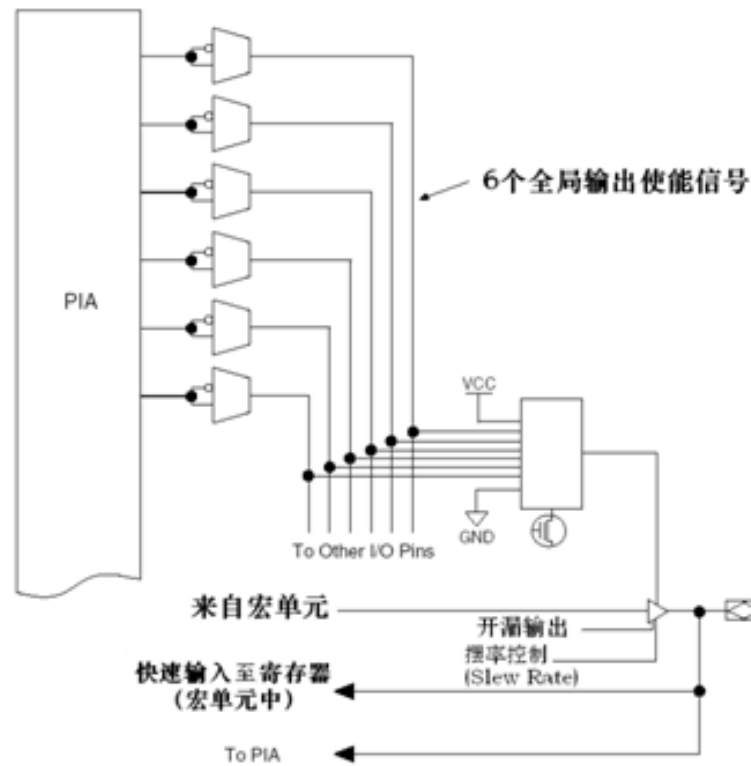


图 1-18 EPM7128S 器件的 I/O 控制块

1.8 FPGA的结构与工作原理

1.8.1 查找表逻辑结构

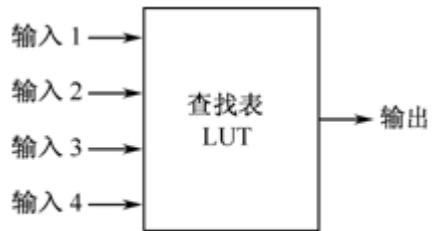


图 1-19 FPGA 查找表单元

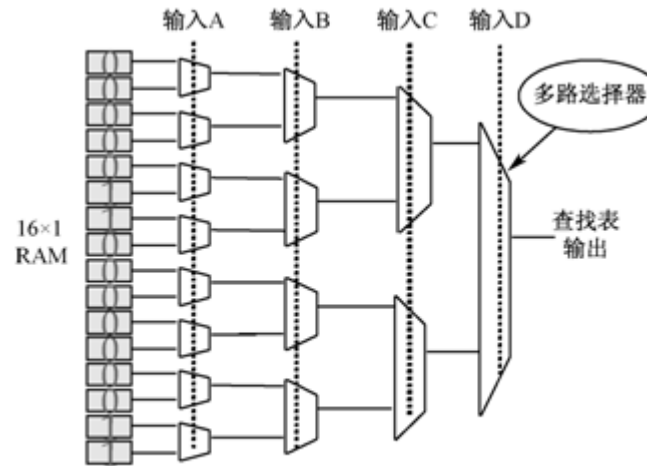


图 1-20 FPGA 查找表单元内部结构

1.8.2 Cyclone III系列器件的结构原理

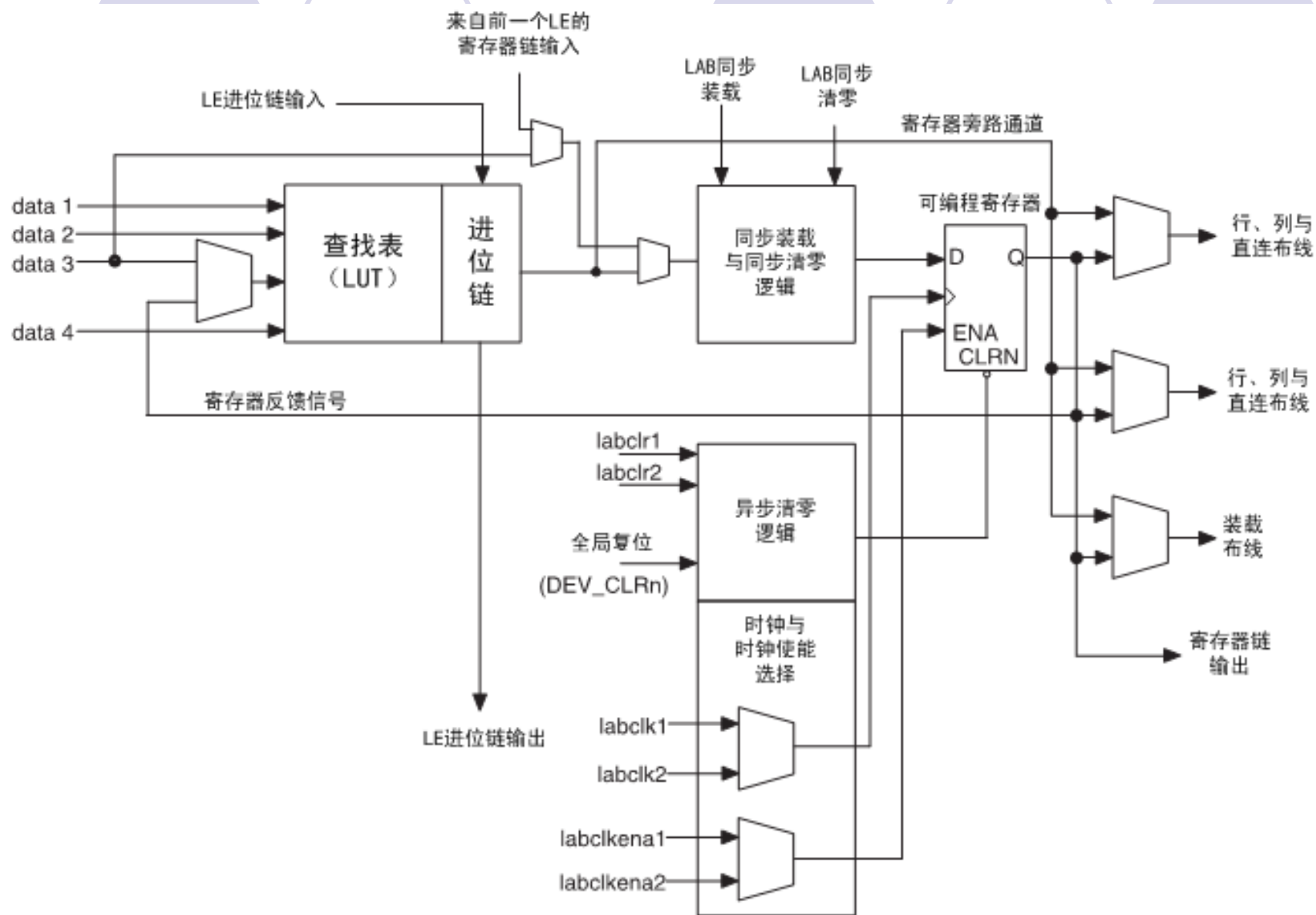


图 1-21 Cyclone III 的 LE 结构图

1.8 FPGA的结构与工作原理

1.8.2 Cyclone III系列器件的结构原理

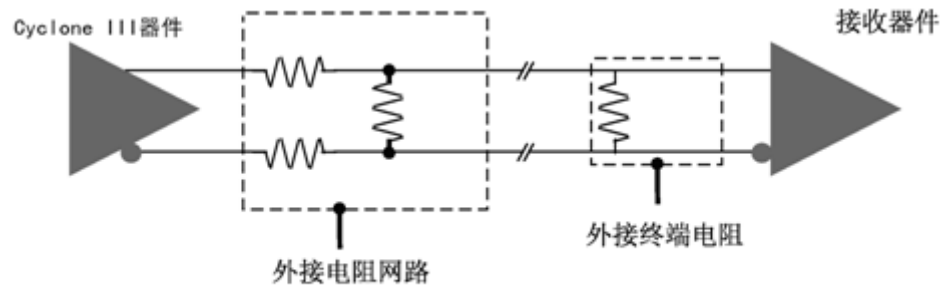


图 1-22 LVDS 连接

1.9 硬件测试技术

1.9.1 内部逻辑测试

1.9.2 JTAG边界扫描测试

表 1-1 边界扫描 IO 引脚功能

引脚	描述	功能
TDI	测试数据输入(Test Data Input)	测试指令和编程数据的串行输入引脚。数据在 TCK 的上升沿移入。
TDO	测试数据输出(Test Data Output)	测试指令和编程数据的串行输出引脚，数据在 TCK 的下降沿移出。如果数据没有被移出时，该引脚处于高阻态。
TMS	测试模式选择(Test Mode Select)	控制信号输入引脚，负责 TAP 控制器的转换。TMS 必须在 TCK 的上升沿到来之前稳定。
TCK	测试时钟输入(Test Clock Input)	时钟输入到 BST 电路，一些操作发生在上升沿，而另一些发生在下降沿。
TRST	测试复位输入(Test Reset Input)	低电平有效，异步复位边界扫描电路(在 IEEE 规范中，该引脚可选)。

1.10 编程与配置

基于电可擦除存储单元的**EEPROM**或**Flash**技术

基于**SRAM**查找表的编程单元。

基于反熔丝编程单元。

主动配置方式

被动配置方式

1.11 Quartus II

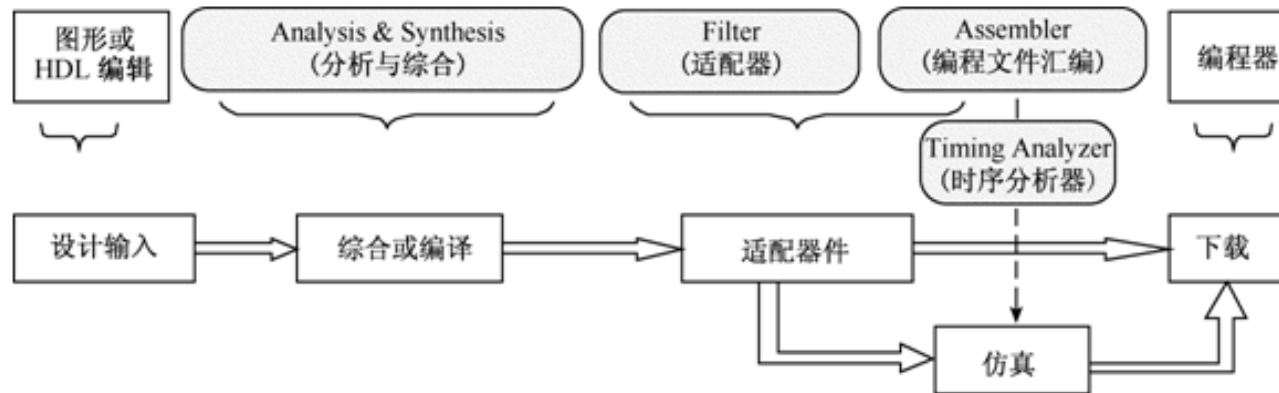
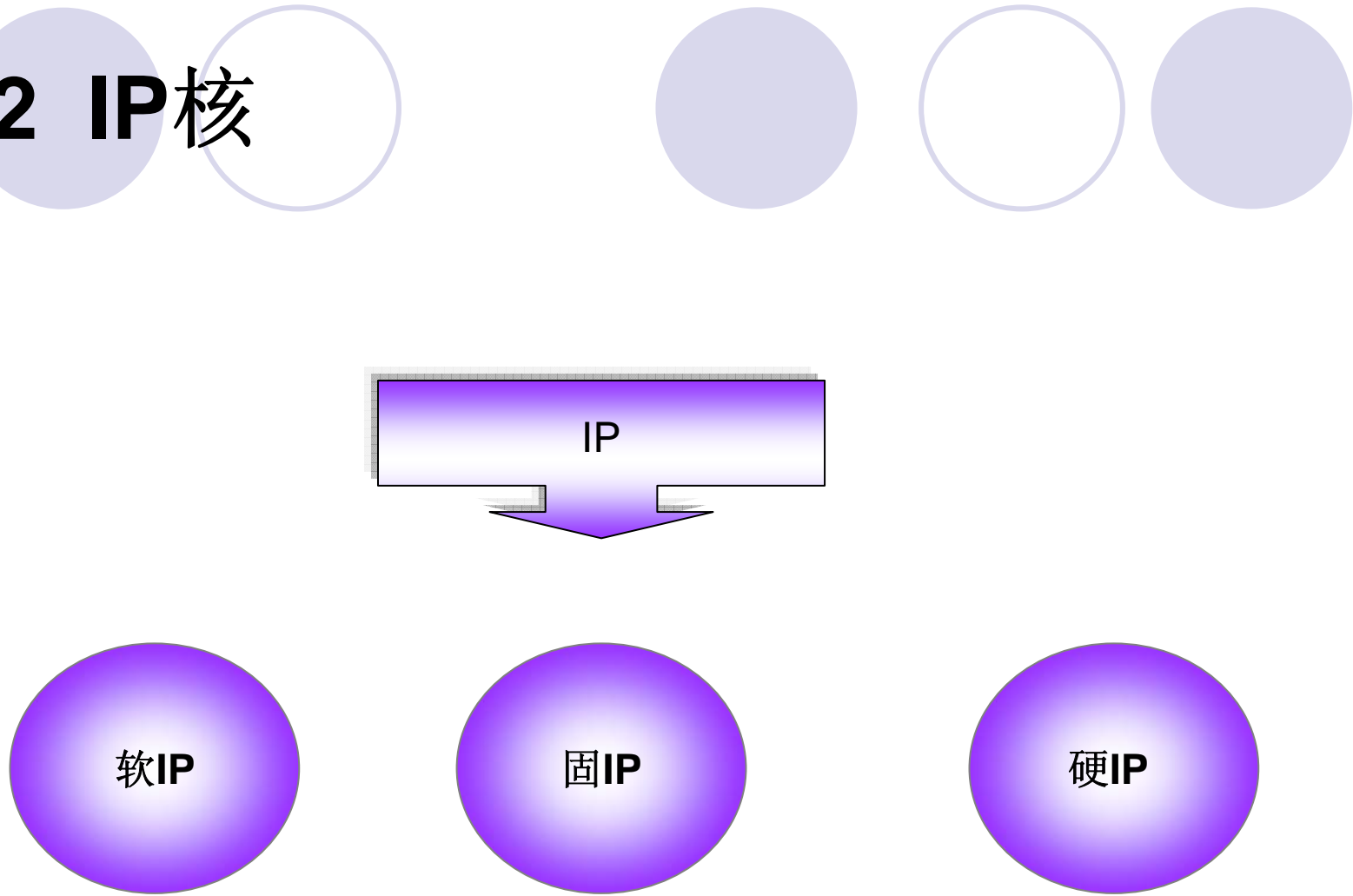


图 1-23 Quartus II 设计流程

1.12 IP核



1.13 EDA的发展趋势

- ☆ 在一个芯片上完成的系统级的集成已成为可能。
- ☆ 可编程逻辑器件开始进入传统的**ASIC**市场。
- ☆ **EDA**工具和**IP**核应用更为广泛。
- ☆ 高性能的**EDA**工具得到长足的发展，其自动化和智能化程度不断提高。
- ☆ 为嵌入式系统设计提供了功能强大的开发环境。
- ☆ 计算机硬件平台性能大幅度提高，为复杂的**SOC**设计提供了物理基础。