

第一章

数字系统设计概述



目录 CONTENTS

1

集成电路发展现状

2

数字系统概念

3

数字系统的实现方法

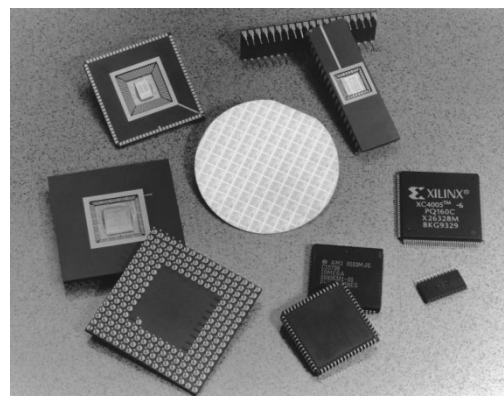
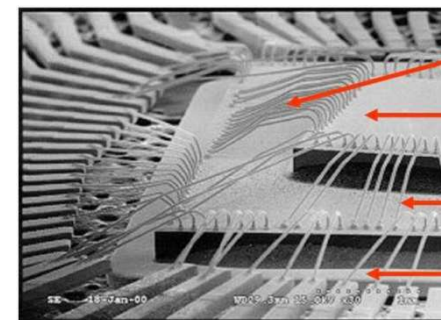
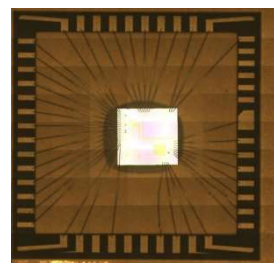
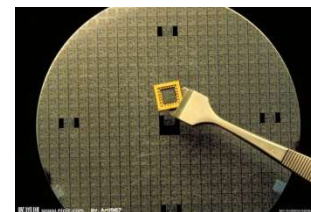
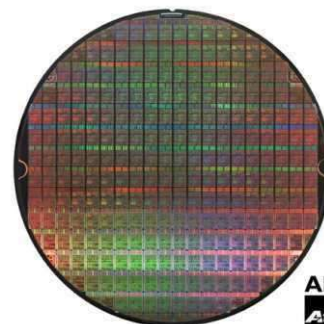
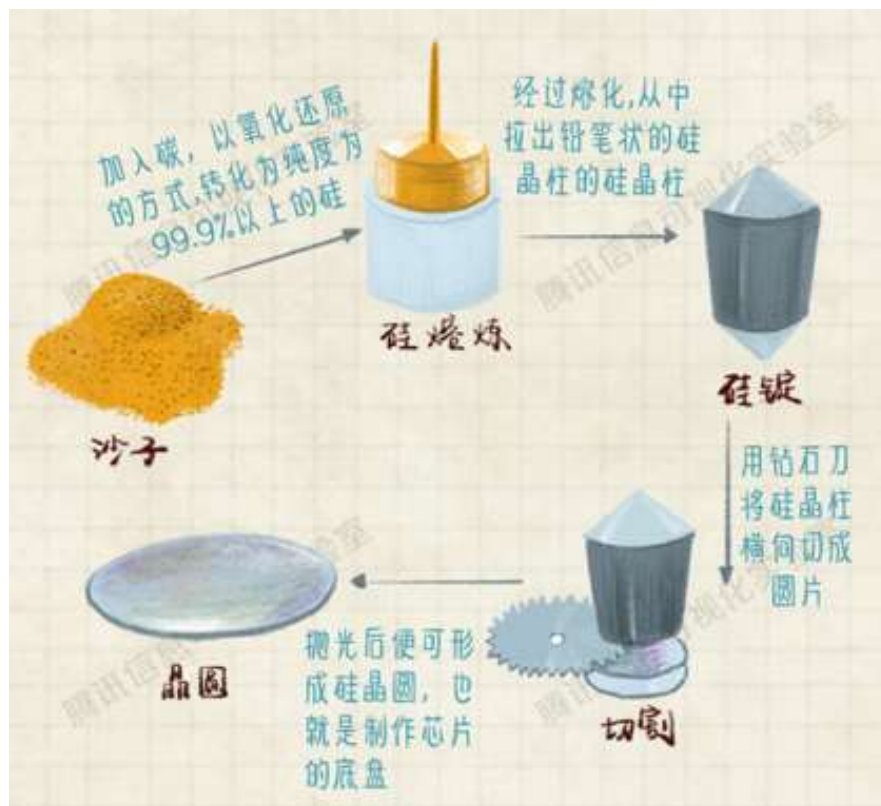
4

基于HDL和EDA工具的设计流程



集成电路制造过程

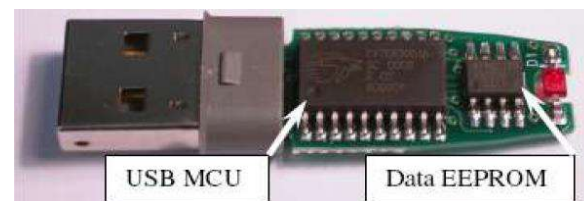
点石成金



集成电路发展现状-产业特点

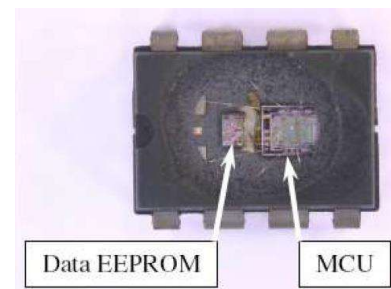
□ 集成电路产品周期比较短

- 电子产品更新换代非常快
- 先进的设计方法学



□ 设计与制造分开

- Fabless公司
- 加工、封装、测试
- 联发科



1997年05月：聯發科成立於新竹科學工業園區
1998年：推出全球最快速的48X CD-ROM晶片組
1999年：推出全球最快速12X DVD-ROM晶片組
2000年：推出CD-ROM單晶片解決方案
2003年：全球第一套Combo單晶片解決方案
2006年：GSM手機Turn-Key解決方案



目录 CONTENTS

1

集成电路发展现状

2

数字系统概念

3


数字系统的实现方法

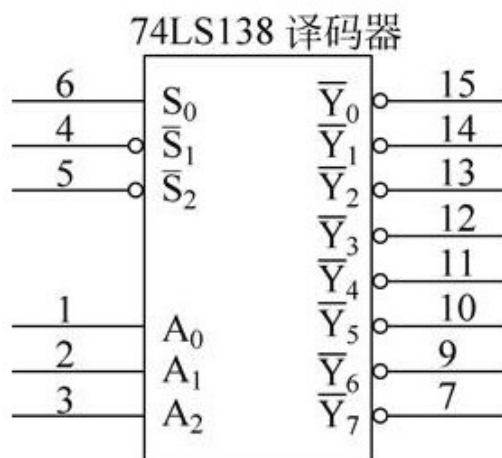
4

基于HDL和EDA工具的设计流程



□ 功能级电路

- 

[illegible]



数字系统的概念

□ 系统

- 钱学森：由相互作用相互依赖的若干组成部分结合而成的，具有特定功能的有机整体
- 而且这个有机整体又是它从属的更大系统的组成部分。
- 计算机
 - 软件系统
 - 硬件系统
 - 运算子系统
 - 存储子系统



数字系统的概念

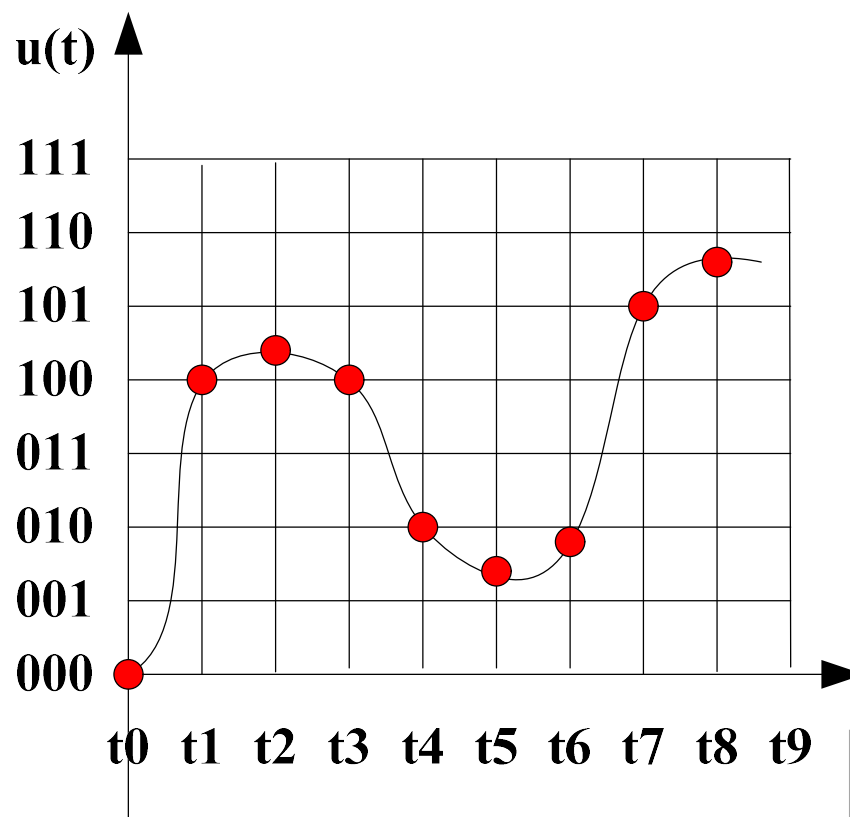
□ 数字系统 (Digital System)

- 由若干数字电路和逻辑部件构成的、能够实现**数据存储、传送和处理等复杂功能**的数字电子系统
 - 通用处理器：CPU
 - 专用数字系统：GPU、存储器、调制解调器等



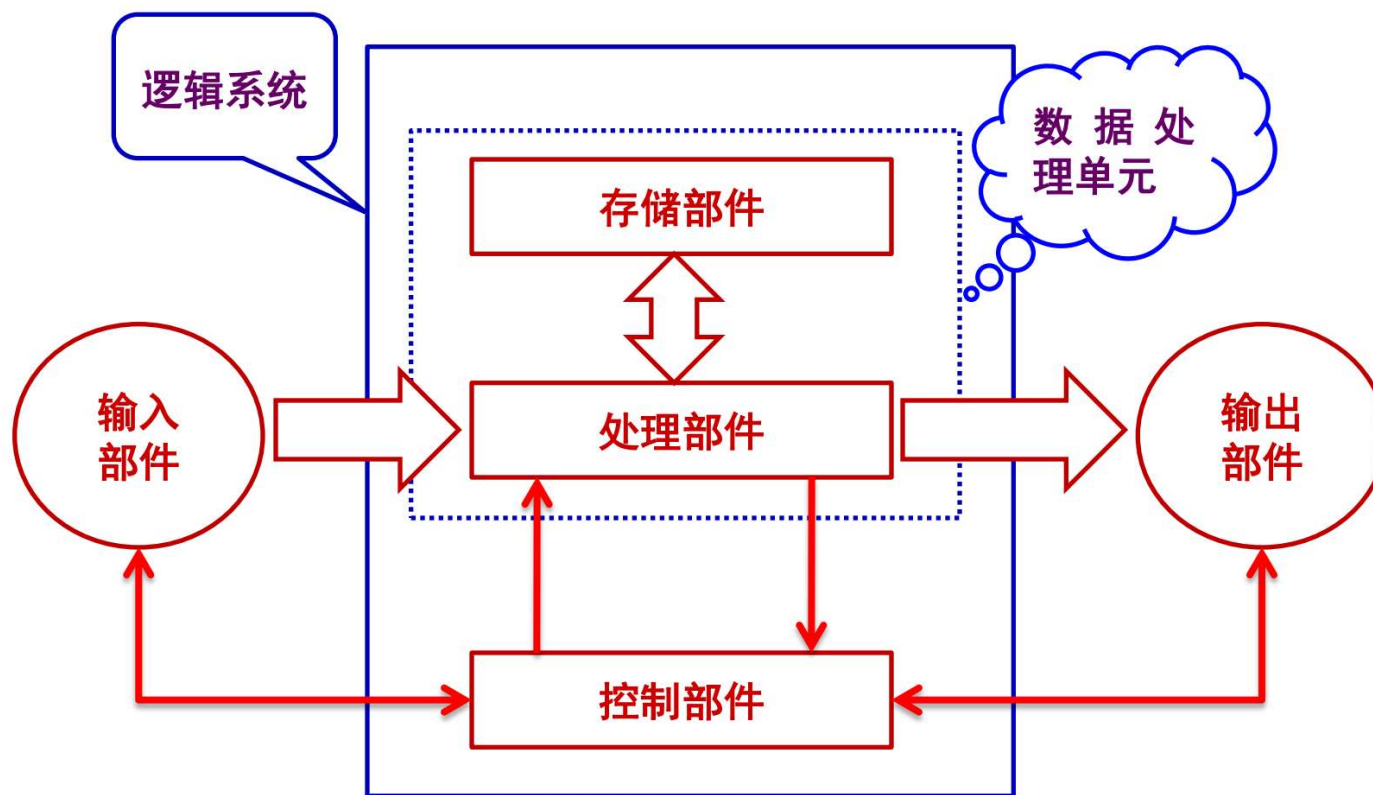
数字系统的概念

- 数字系统主要处理离散信号：数字、文字、图像、视频、声音等
- 数字系统在处理模拟信号之前，要对模拟信号进行量化（数字化）处理。



数字系统的概念

- 数字系统在结构上分为数据处理单元和控制单元



数字系统内部所传输处理的对象均为基本数字逻辑变量



目录 CONTENTS

1

集成电路发展现状

2

数字系统概念

3

数字系统的实现方法

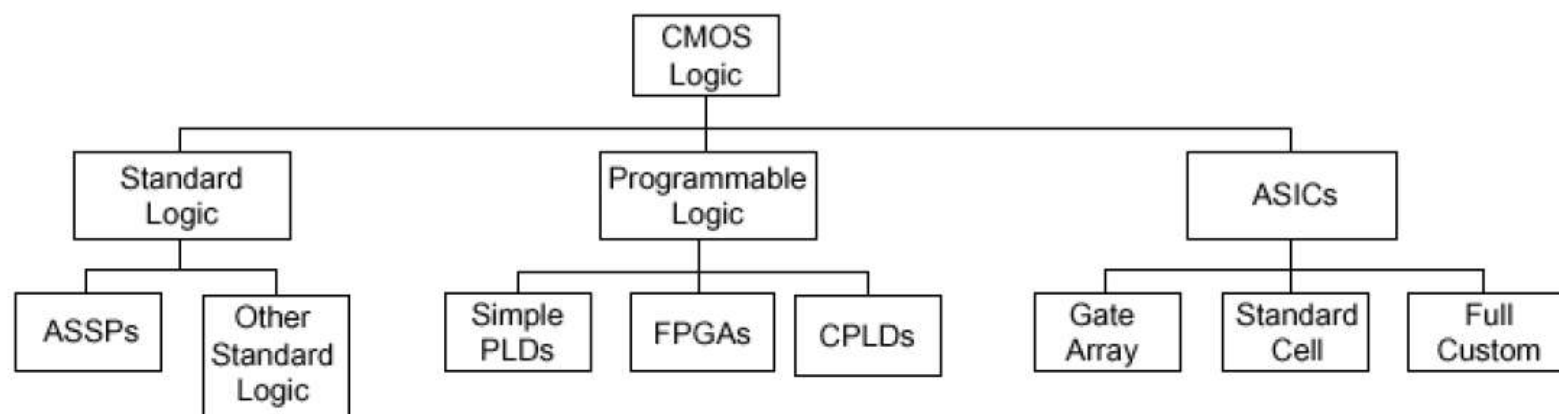
4

基于HDL和EDA工具的设计流程



数字系统的实现方法

- ASSP (Application Specific Standard Product) 专用功能标准产品
- PLD (Programmable Logic Device) 可编程逻辑器件
- ASIC (Application Specific Integrated Circuit) 专用集成电路



数字系统的实现方法-ASSP

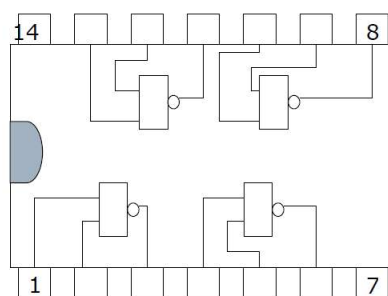
□ 专用功能标准产品

- 具备特定功能的芯片：74系列、4000系列
- 反相器、或门、或非门、与或非门、比较器、译码器

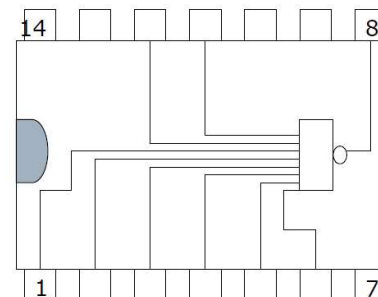


□ 特点

- 非常通用
- 功能相对简单
- 适合设计小规模系统



74LS00
2输入4与非门

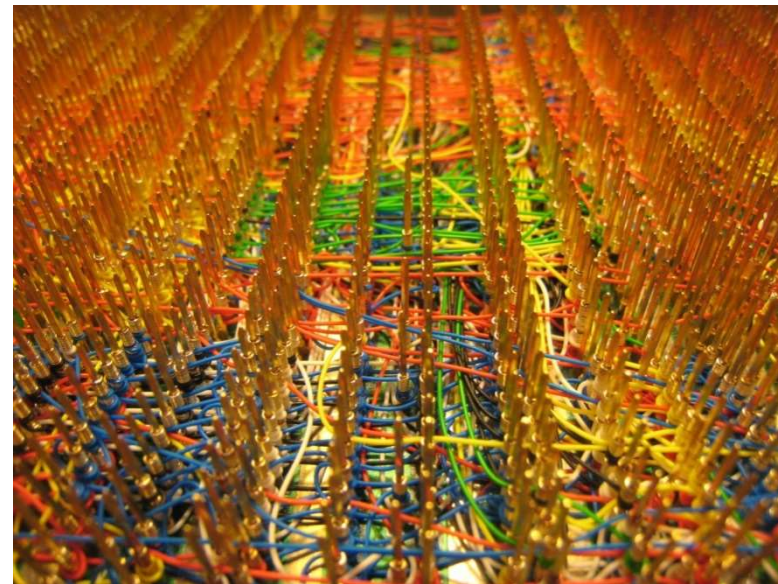
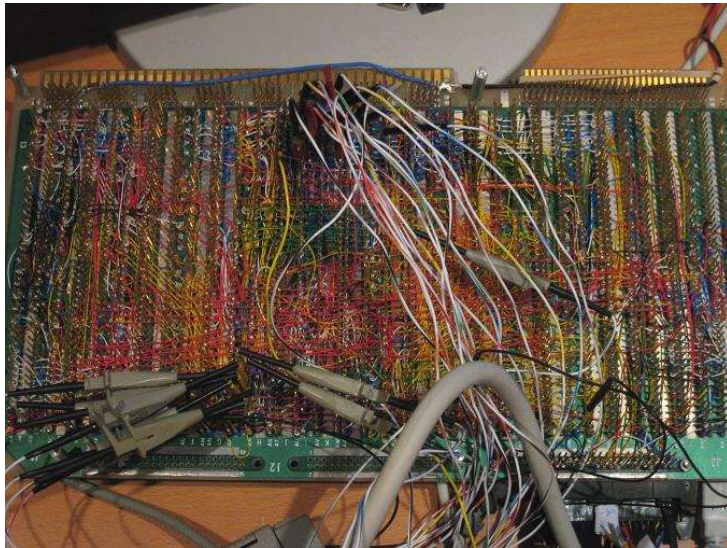


74LS30
8输入与非门

数字系统的实现方法-ASSP

□ 纯手工搭建CPU

- 2009年世界创意大会, Steve
- Big Mass Of Wires



用ASSP搭建系统

□ 纯手工搭建CPU



数字系统的实现方法-PLD

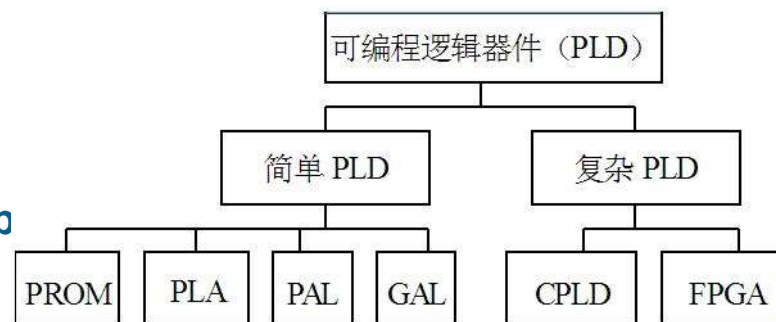
□ 可编程逻辑器件

□ 分类

- 复杂可编程逻辑器件(Complex Programmable Logic Device, CPLD) – 小规模
- 现场可编程门阵列 (Field - Programmable Gate Array, FPGA) – 大型系统

□ 特点

- 灵活
- 规模：几百-上千万门
- 价格：几十-几万人民币
- FPGA接口丰富
- 设计开发周期短、设计制造成本低
- 适用于非批量生产应用



可编程逻辑器件 (PLD)

基于乘积项技术的PLD
(CPLD)

与阵列固定，或阵列可编程

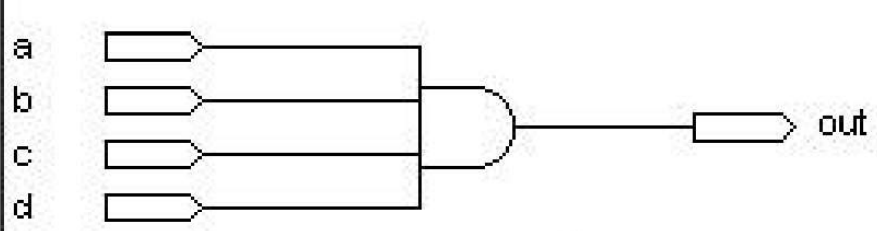
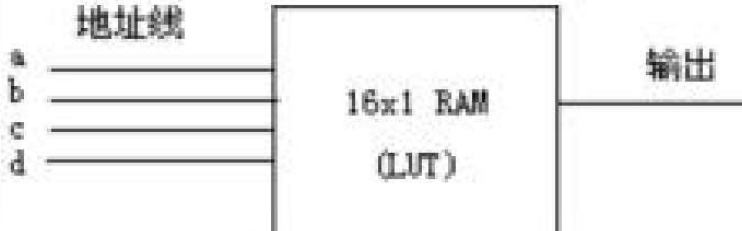
与阵列、或阵列均可编程

与阵列可编程，或阵列固定

基于查找表技术的PLD
(FPGA)

基于查找表技术的PLD—FPGA

下图是一个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

CPLD与FPGA的对比

	CPLD	FPGA
内部结构	Product—term	Look—up Table
程序存储	内部EEPROM	SRAM，外挂EEPROM
资源类型	组合电路资源丰富	触发器资源丰富
集成度	低	高
使用场合	完成控制逻辑	能完成比较复杂的算法
速度	慢	快
其他资源	—	EAB，锁相环
保密性	可加密	一般不能保密

数字系统的实现方法-PLD

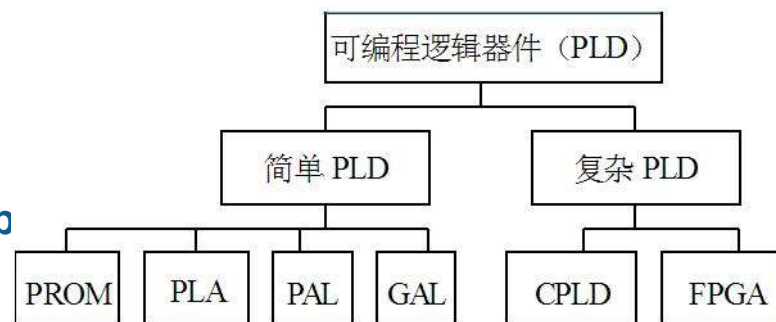
□ 可编程逻辑器件

□ 分类

- 复杂可编程逻辑器件(Complex Programmable Logic Device, CPLD) – 小规模
- 现场可编程门阵列 (Field - Programmable Gate Array, FPGA) – 大型系统

□ 特点

- 灵活
- 规模：几百-上千万门
- 价格：几十-几万人民币
- FPGA接口丰富
- 设计开发周期短、设计制造成本低
- 适用于非批量生产应用



数字系统的实现方法-ASIC

□ 专用集成电路

- 针对某一电路系统的要求专门设计
- 具有特定电路功能，通常市场上买不到
- 性价比好
 - 体积小、低功耗
 - 可靠性
- 应用广泛





全定制

□ 主要设计思想

- 利用集成电路的最基本设计方法（不使用现有库单元），对集成电路中所有的元器件进行精工细作的设计方法

□ 设计特点

- 精工细作，设计要求高、周期长，设计成本昂贵。

□ 适用范围

- 该方法尤其适宜于模拟电路，数模混合电路。
- 以及对速度、功耗、管芯面积、其它器件特性（如线性度、对称性、电流容量、耐压等）有特殊要求的场合；
- 在没有现成元件库的场合。





基于标准单元的设计

□ 基于标准单元的设计方法

- 将预先设计好的称为标准单元的逻辑单元，如与门，或门，多路开关，触发器等，按照某种特定的规则排列，通常按照等高不等宽的原则排列，留出宽度可调的布线通道

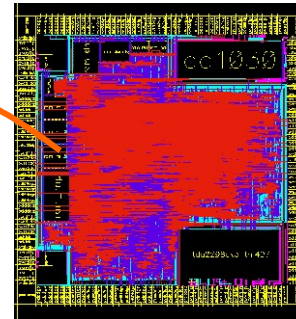
□ 优缺点

- 用标准单元库，省时、省钱、少风险地完成ASIC设计任务
- 标准单元可以置放于芯片的任何位置
- 可内嵌定制的功能单元
- 开发周期较短，开发成本低于全定制
- 需要花钱购买或自己设计标准单元库

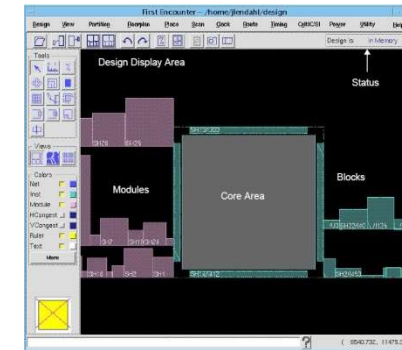
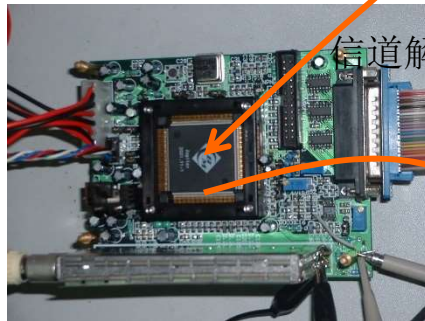




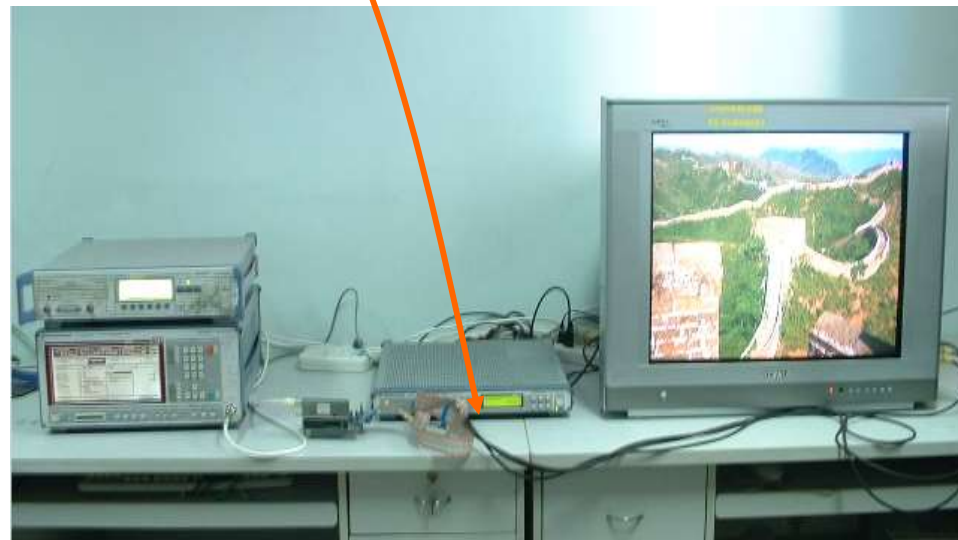
信道芯片版图



信道解调芯片系统板



有线数字电视接收系统测试平台





目录 CONTENTS

1

集成电路发展现状

2

数字系统概念

3

数字系统的实现方法

4

基于HDL和EDA工具的设计流程



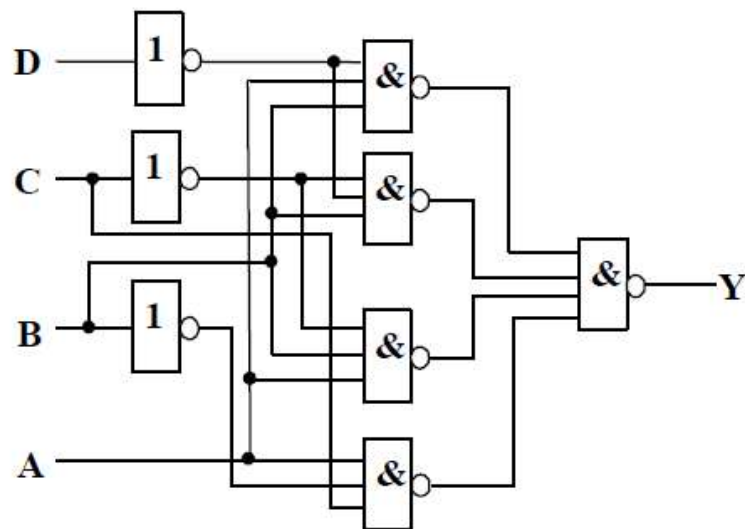
传统的设计方法

□ 以组合逻辑电路设计为例

- 任意时刻的输出仅取决于该时刻电路的输入，与过去的输入无关

□ 电路特点

- 由逻辑门电路组成，无记忆器件
- 信号是单向传输的，无反馈回路



传统的设计方法

1. 确定输入信号和输出信号
2. 确定输入和输出的逻辑状态关系
 - 逻辑函数
 - 真值表
3. 优化逻辑状态表达式
 - 卡诺图化简等
4. 实现电路

$$y1 = a \text{ and } b \text{ or } c$$

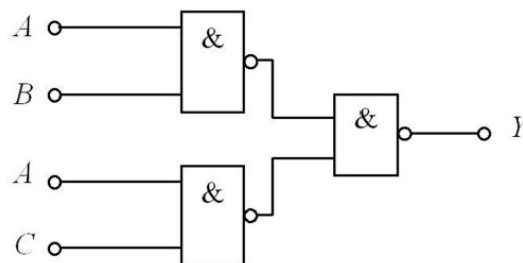
$$y2 = d + e$$

输入			输出
A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Y A	BC			
	00	01	11	10
0				
1		1	1	1

$$Y = AB + AC = \overline{\overline{AB}} + \overline{\overline{AC}} = \overline{\overline{AB} \cdot \overline{AC}}$$

第3、4部可以
交给计算机自
动完成



新的组合逻辑电路的设计步骤

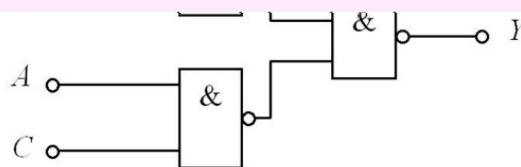
1. 确定输入信号和输出信号
2. 确定输入和输出的逻辑状态关系
3. 用HDL（硬件描述语言）正确描述电路功能

输入			输出
A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

```
module aa(y,a,b,c)
```

让计算机软件优化并生成电路

```
assign y=a&b+a&c;  
endmodule
```





基于HDL的数字系统设计方法

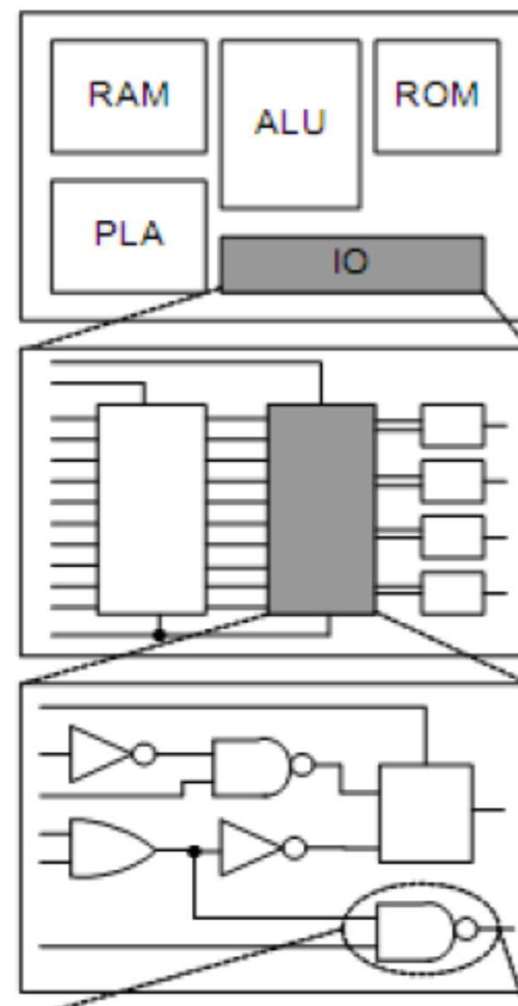
□ 关键要素

- 数字系统设计EDA工具
- Top-down设计思想
- 层次化设计方法
- 硬件描述语言（HDL）



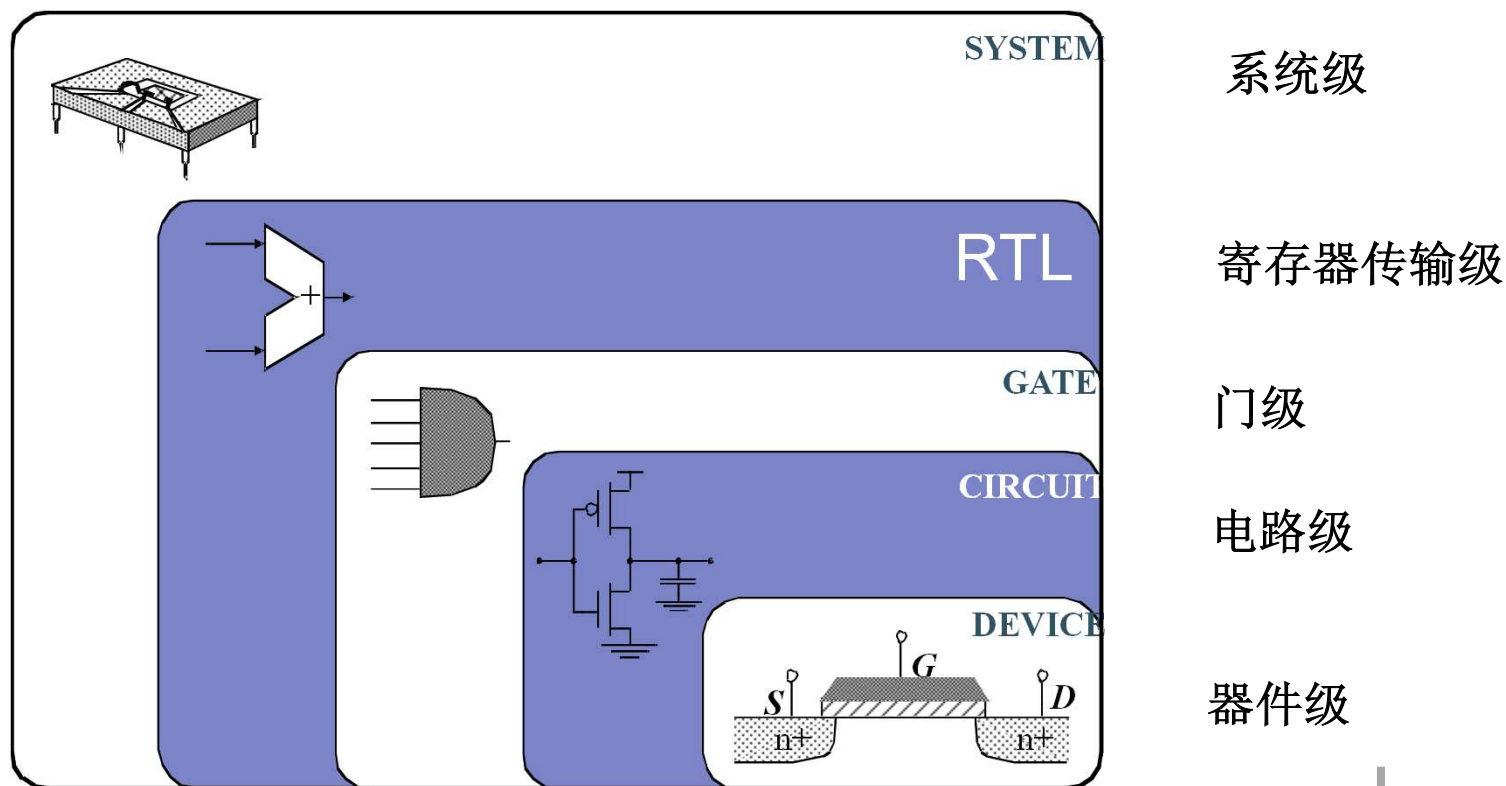
层次化设计方法 - 设计抽象

- 层次式设计方法的核心思想是通过设计抽象。
- 各层次的设计由若干模块或单元组成。在进行某一层次的设计时，我们只需考虑模块的整体外部行为特性，而不必考虑内部组成。
- 这样模块被抽象成为一个隐藏了内部细节的黑箱，其外部特性的描述构成该模块的模型。

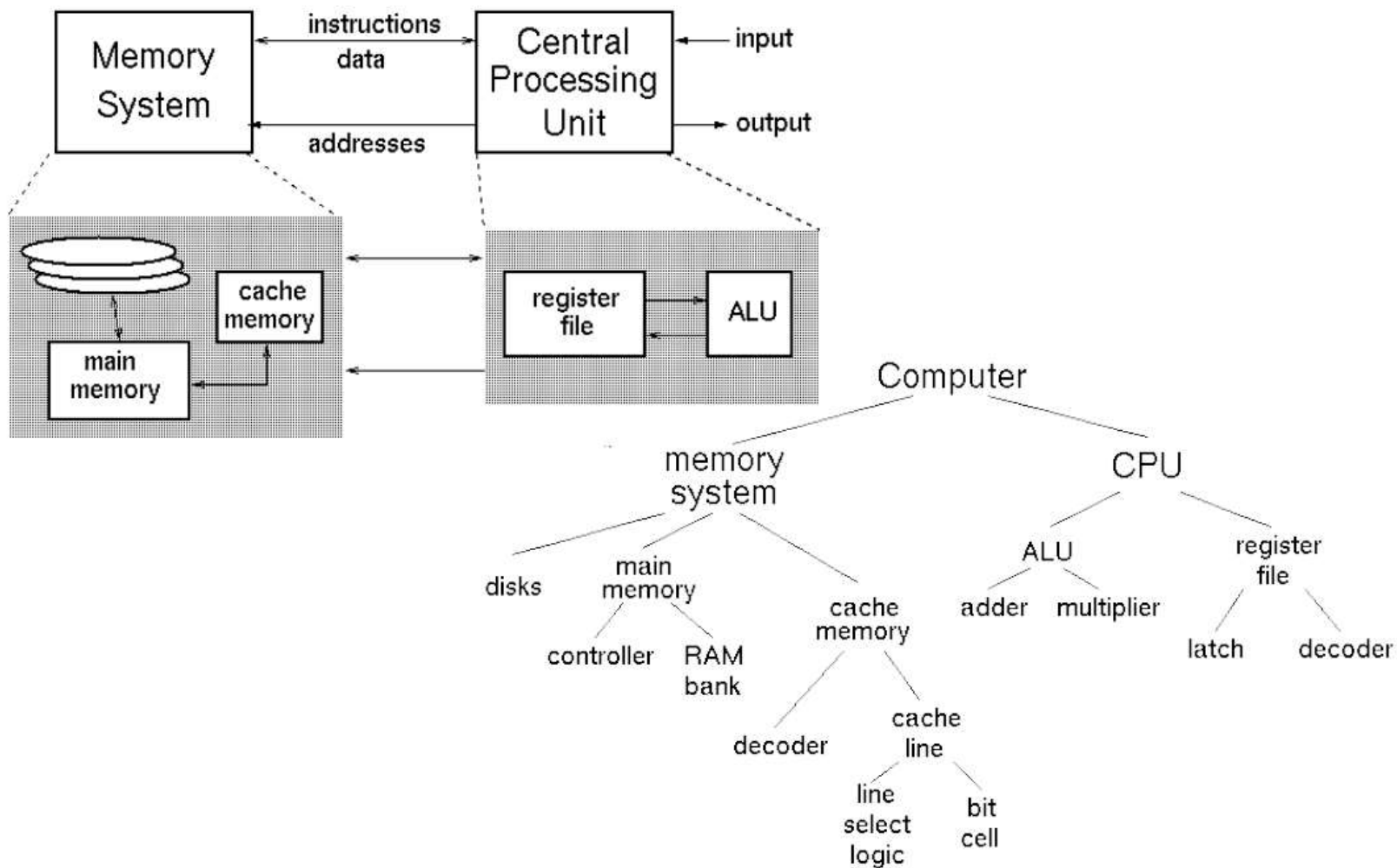


层次化设计方法 - 设计层次

- 每一层次处理的模块或单元数量是有限的。因此高度复杂的大规模数字设计就通过这样多层次、模块化的分割解决了
- 数字电路与系统通常分为如下层次

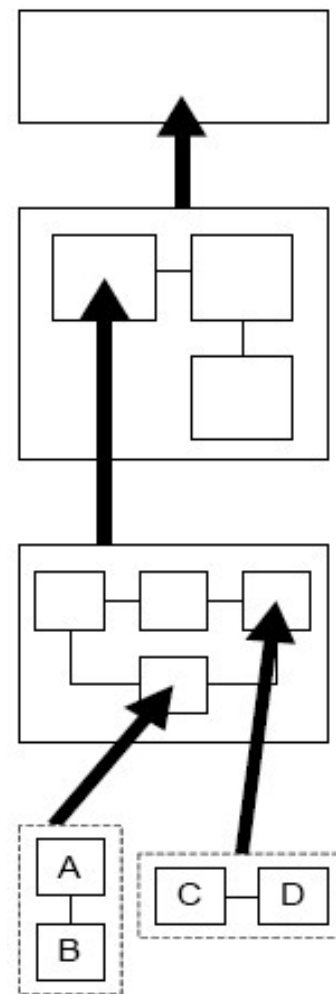


层次式设计



Bottom-up设计

- 传统的设计是自底向上的设计方法。如设计一个数字系统，必须首先决定使用的器件（如74系列的器件，不同类别的RAM、ROM、CPU等），然后构成各个功能模块（如数据采集、处理、通信接口等模块），最后合成整个系统。
- 主要问题是在进行底层设计时缺乏对系统总体性能的把握，在合成整个系统后，可能发现无法达到要求的功能或性能，**难以保证一次设计的成功率。**
- 因此自底向上的设计方法效率低，不能满足大规模系统的设计需要。





Top-down设计

- 自顶向下设计方法得以实施的前提是：
- 各层次、各类模块功能与性能的精确定义与描述方法（建模）
- 各层次的仿真及综合工具
- 底层精确的工艺库支持
- 这就是EDA技术要解决的主要问题。因此只有在EDA技术发展逐步成熟以后，自顶向下的设计才成为可能



Top-down基本设计流程

□ 系统设计

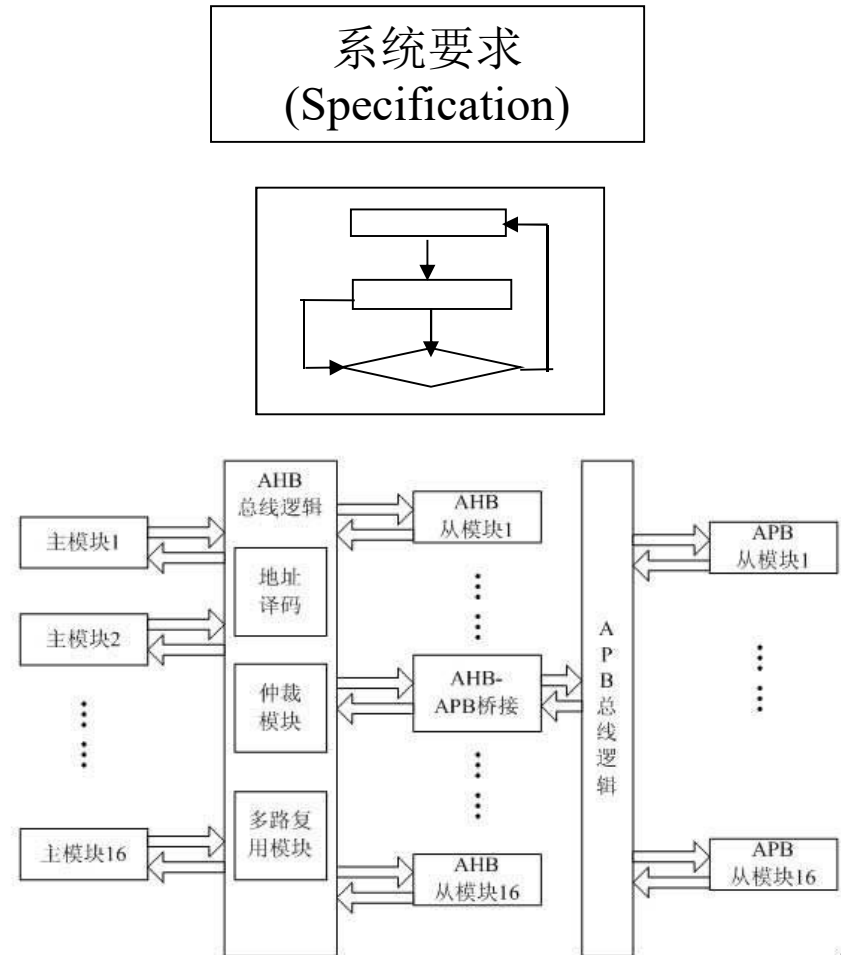
- 系统需求定义
- 软硬件设计划分
- 算法设计
- 体系结构设计

□ 逻辑设计

- 逻辑综合
- 逻辑优化

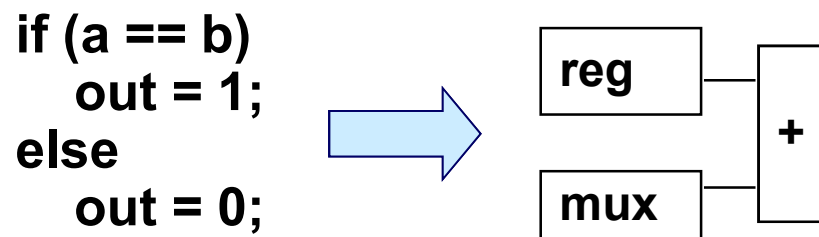
□ 物理设计（不同）

- 布局规划（Floorplanning）
- 布局（Placement）
- 布线（Routing）



逻辑设计

- 主要通过硬件描述语言的行为描述实现
- 寄存器传输级
 - 设计者编写的代码描述的是每一个clock下，触发器集合A的旧值进行逻辑运算，产生新值，存入触发器集合B的过程。
 - 可综合得到电路实现
- 依靠逻辑综合，之后的设计基本可由EDA工具自动完成





Top-down设计的优点

- Top-down设计方法可在系统级完成整个系统的模拟和性能分析，一开始就掌握实现目标系统的性能状况
- 随着设计层次向下进行，系统逐步细化，每一层次的设计都经过仿真验证，性能得到保证，从而保证了最后设计结果的正确性，设计成功率高
- 当规模越大时，这种方法的优越性越明显



EDA支持下基于标准单元的ASIC设计流程

