计算机组成原理

第一章

（1）冯诺依曼体系结构的主要内容：

（指令数据采用二进制，实现了“程序存储”，硬件由5个基本部分构成）

论述类两个重要思想：

二进制（不是开关连线），存储程序（不是十进制）

明确了计算机的五部分：

运算器、控制器、存储器、输入设备、输出设备

指令在计算机中是按地址顺序执行的

（2）计算机系统的组成

计算机硬件：运算器、控制器、存储器、输入设备、输出设备

计算机软件：系统软件，应用软件

第二章

**（1）运算器的作用**

算数运算

逻辑运算

**（2）计算器中的机器码**

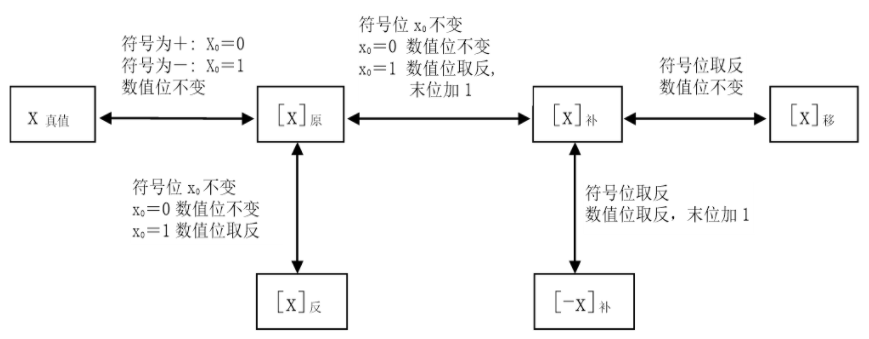
原码：二进制数真值加上符号位，正：0，负：1

反码：正数与原码一致；负数，除了符号位全部求反

补码：正数与原码一致；负数，反码的基础上在末尾加1

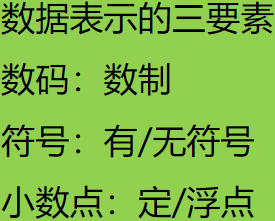
移码：正数与原码一致；负数，在补码的基础上给符号位求反

不同机器码间的转换：

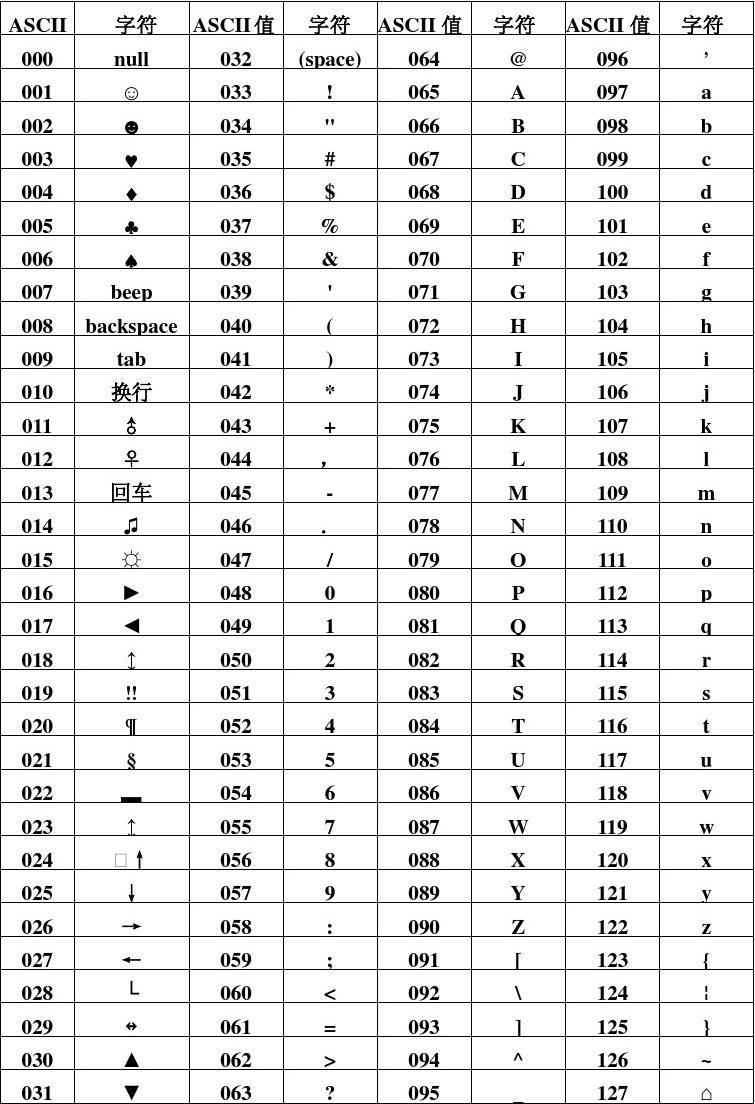


注意：补码是目前微型机使用

**（3）0表达唯一的机器码**



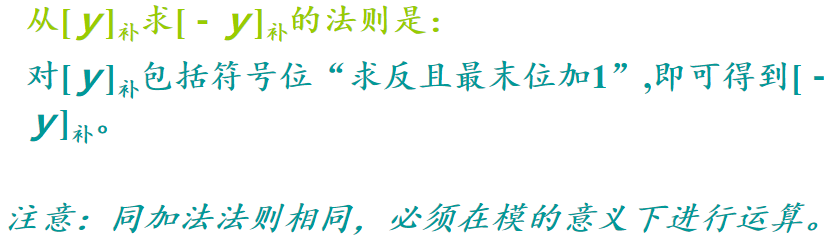
**（4）计算器中字母与字符的编码ASCII码**



**（5）定点数的加减运算：**

补码加法：[x]补+[y]补=[x+y]补

补码减法：[x-y]补=[x-(y)]补=[x]补+[-y]补



**（6）定点数的加减运算溢出产生的原因：**

运算结果超出了机器字长表达的范围

**（7）定点数加减法溢出判断的方法**

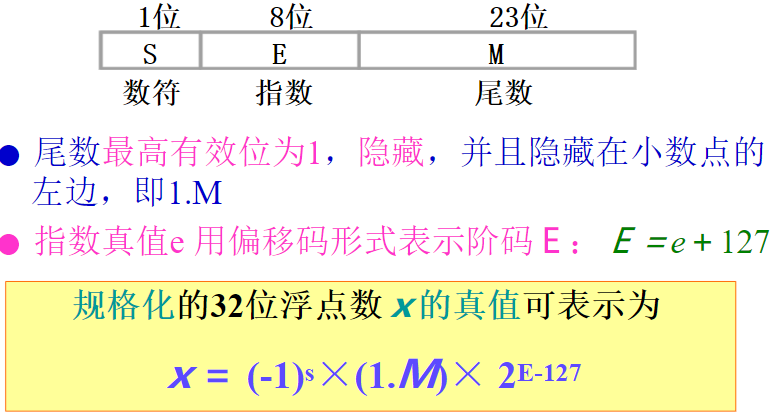
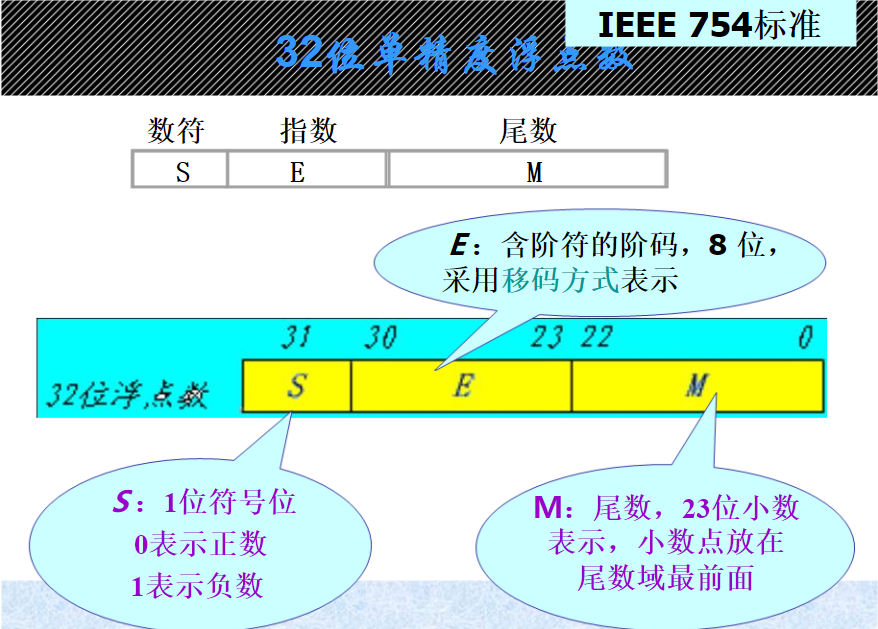
双符号位法（当结果两位符号位的值不一致的时候表示溢出）：

00：正数，11：负数，01：上溢，10：下溢

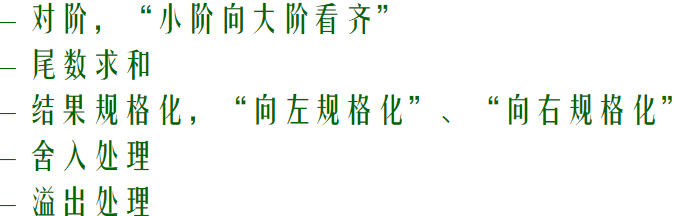
单符号位法（数值部分最高位产生的进位CS，符号部分向上产生的进位为CF，溢出公式：V=CFCS）：

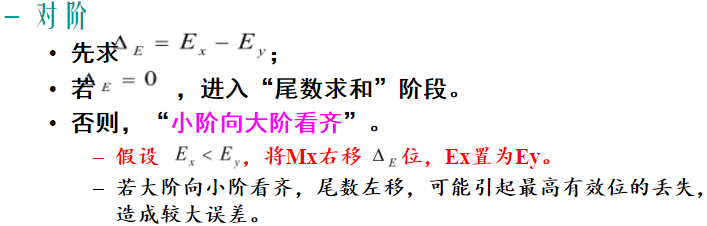
当最高有效位产生进位而符号位无进位时，产生上溢；当最高有效位无进位而符号位有进位时，产生下溢

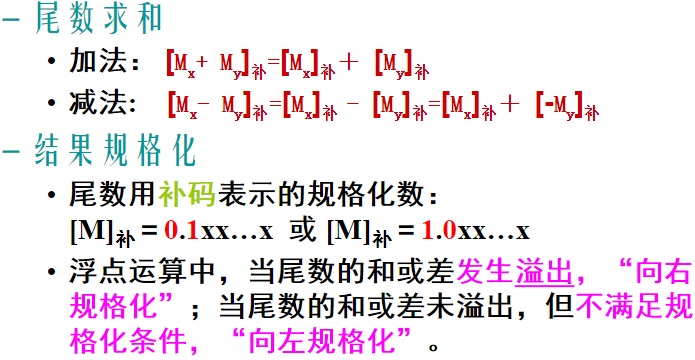
**（8）32位单精度浮点数IEEE754标准**

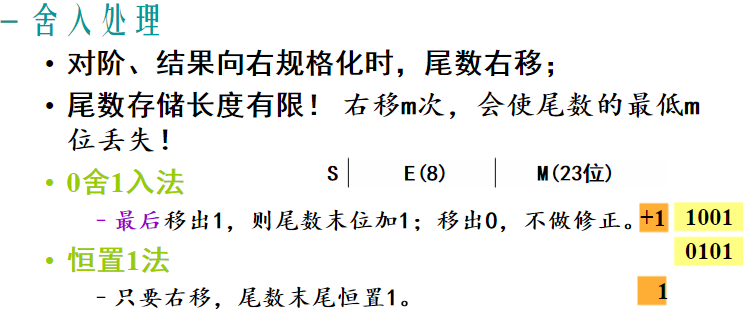


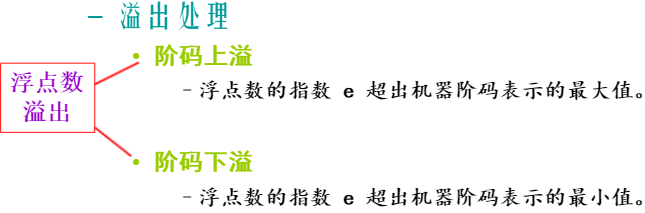
**（9）浮点数加减法运算的步骤与方法**











**（10）奇偶校验**

校验码：能够发现甚至纠正信息传输或存储过程中出现错误的编码

检错码：仅能检测出错误的编码

纠错码：能够发现并纠正错误的编码

最简单且应用广泛的检错码：奇偶校验码

偶校验：编码中1的个数为**偶数**时校验位为**0否则为1**

奇校验：编码中1的个数为**奇数**时校验位为**0否则为1**



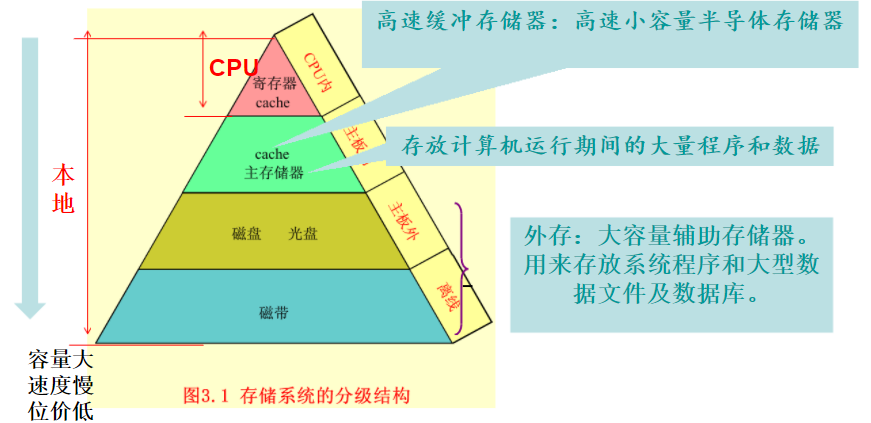
第三章  
**（1）不同存储器保存数据的条件：**



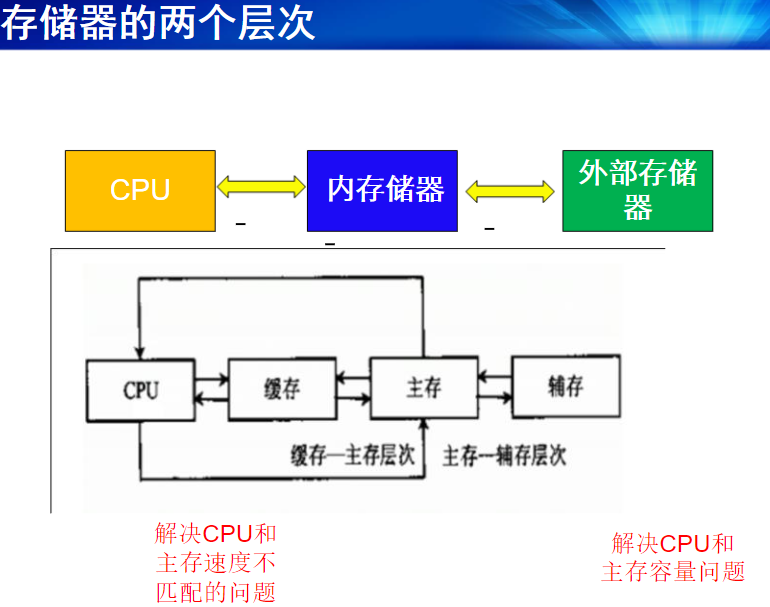
（SRAM）高速缓冲存储器中数据保存的条件是（电源 ）

（DRAM）内存条中数据保存的条件是（电源、刷新）

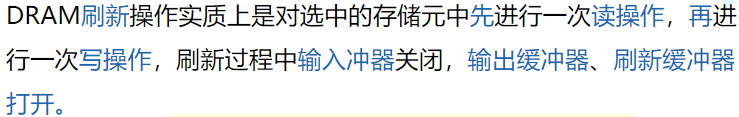
**（2）多层次存储器结构（作用目的、速度、容量）：**



|  |  |  |  |
| --- | --- | --- | --- |
|  | 目的 | 速度 | 容量 |
| 内存 | 用来存储计算机运行期间的大量程序和数据 | 中 | 中 |
| 高速缓冲存储器 | 为提高计算机的处理速度 | 快 | 小 |
| 辅助存储器 | 存放系统程序和大型数据文件及数据库 | 慢 | 大 |

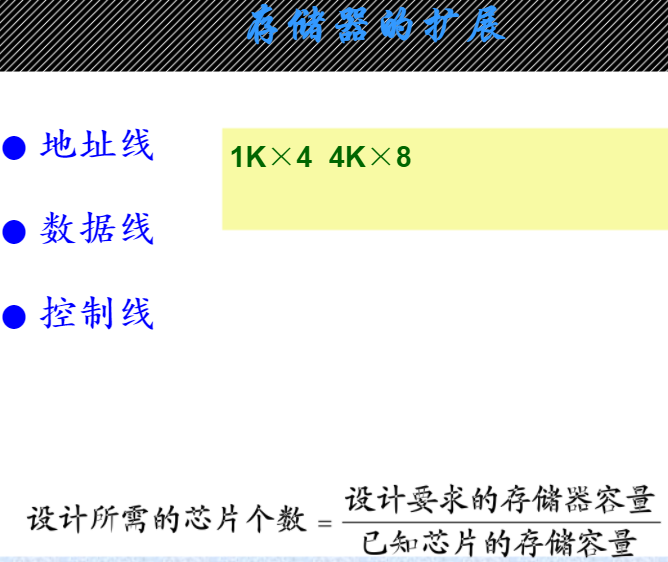


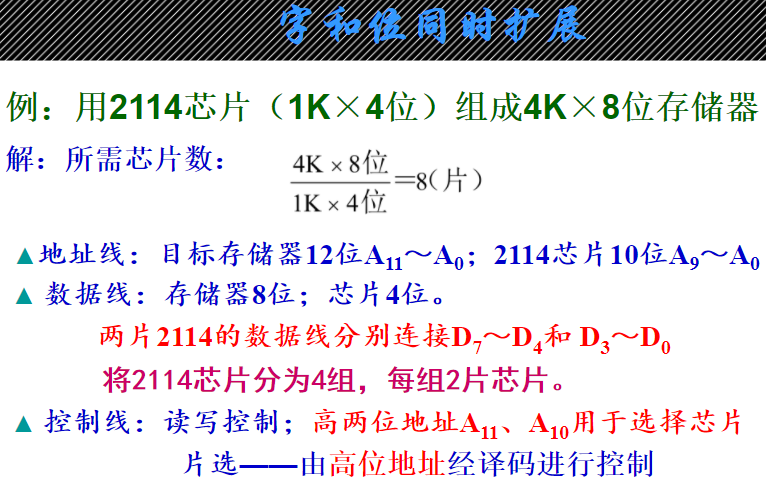
**（3）刷新 死区：**

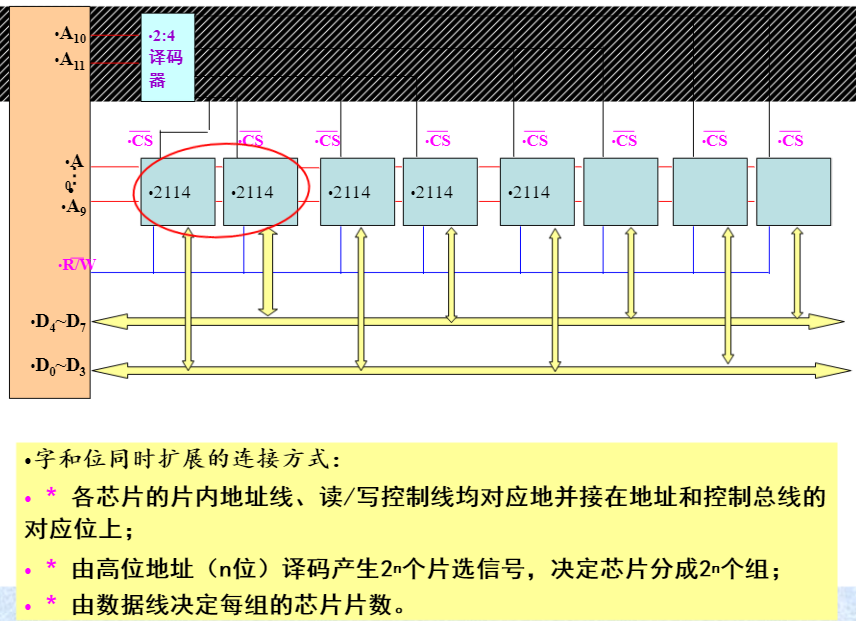




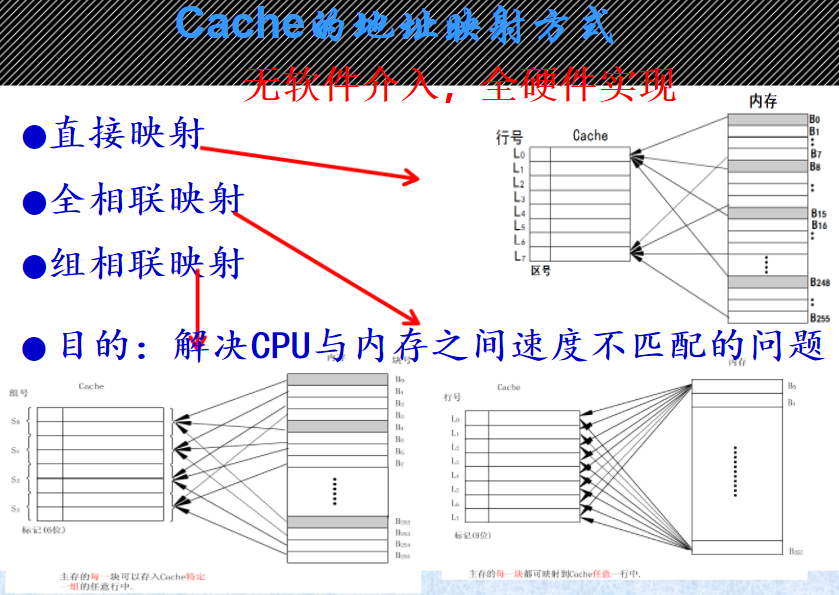
**（4）存储器的扩展（字长和容量扩展）：**

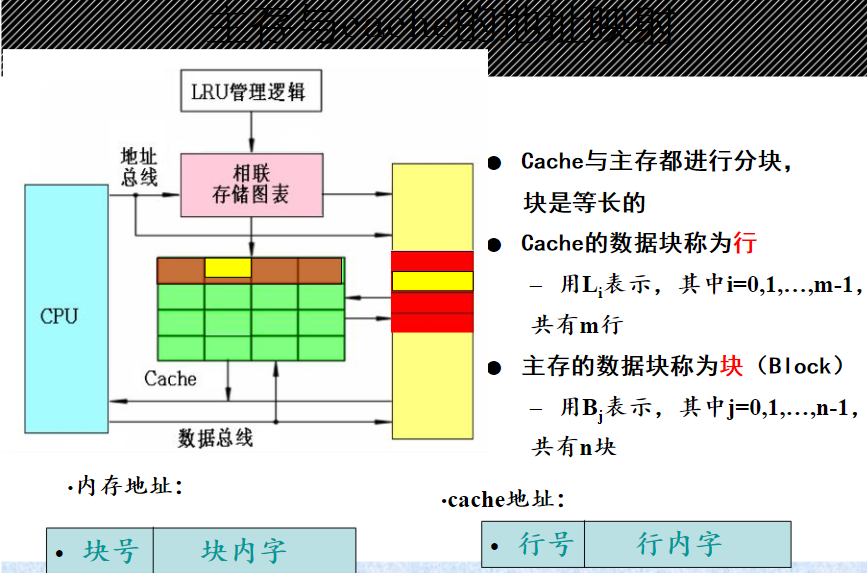




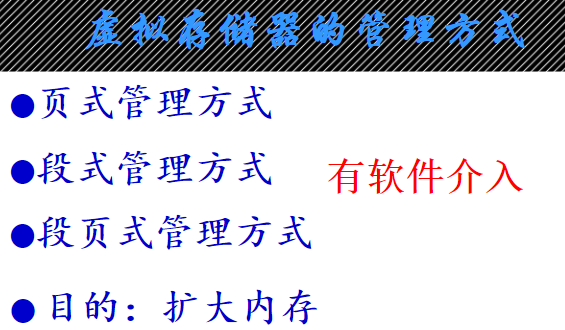


**（5）Cache地址映射方式的内存地址格式：**





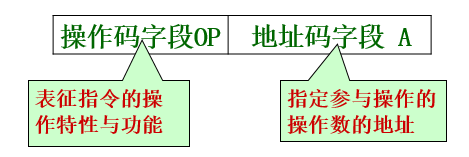
**（6）虚拟存储器的管理方式：**



第四章

1. **指令结构**

指令格式即指令字用二进制代码表示的结构形式，由操作码字段和地址码字段组成。



1. **指令和操作数的寻址方式**

寻址方式：形成指令或操作数地址的方式

指令寻址方式：用于形成指令在内存中的地址

指令的寻址方式：顺序寻址和跳跃寻址

数据寻址方式：用于形成操作数的地址

操作数的寻址：把操作数的形式地址，根据间址和变址等组合变换为操作数有效地址的过程。

1. **常用操作数寻址方式中数据的存放位置**

立即寻址：指令中

直接寻址：存储器

间接寻址：存储器

寄存器寻址：寄存器  
寄存器间接寻址：存储器

偏移寻址：存储器

1. **操作数寻址方式速度比较（从快到慢）**

1寄存器寻址：寄存器

2立即寻址：指令中

3直接寻址：存储器

4寄存器间接寻址：存储器

5间接寻址：存储器

6偏移寻址：存储器

1. **CISC指令系统与RISC指令系统**

CISC（出现较早，大而全）：

1. 指令系统复杂
2. 各种指令都能访问存储器
3. 指令周期长
4. 有专用寄存器
5. 采用微程序控制

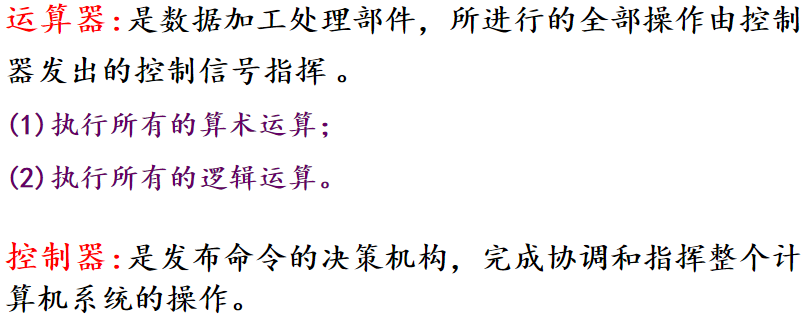
RISC（小而精）：

1. 简化的指令系统
2. 以寄存器-寄存器方式工作
3. 指令周期短
4. 采用大量通用寄存器，以减少访存次数
5. 采用组合逻辑电路控制，不用或少用微程序控制

第五章

1. **CPU的组成**





1. **CPU内部的寄存器**

通用寄存器：存放参与运算的源操作数或运算结果

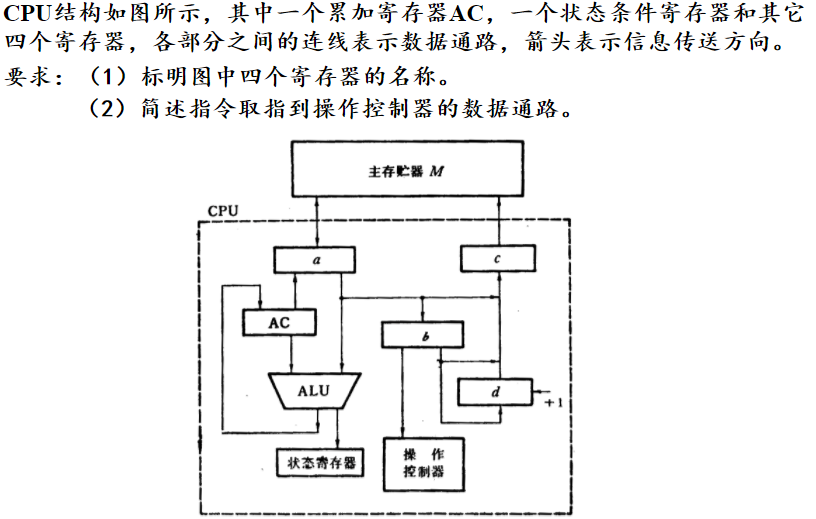
DR（数据缓冲存储器）：暂时存放由内存、外部接口读出的数据或ALU的结果

PSW（状态字寄存器）：保存由算术和逻辑指令的结果建立的各种条件码

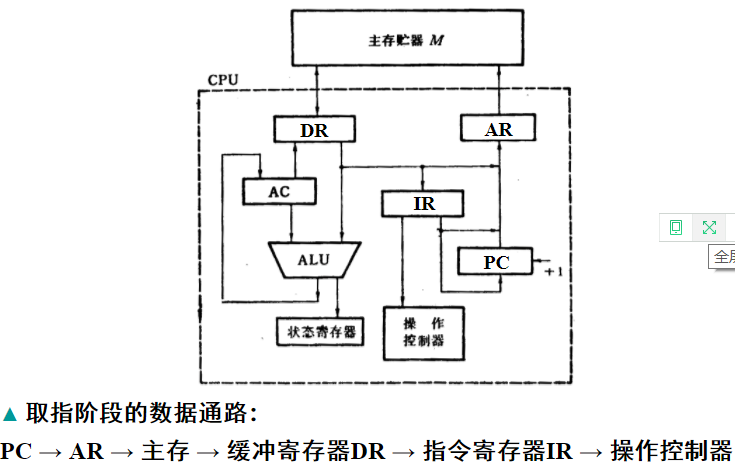
AR（数据地址寄存器）：保存当前CPU所访问的内存单元的地址

PC（程序计数器）：保存下一条指令的地址

IR（指令寄存器）：保存当前正在执行的一条指令

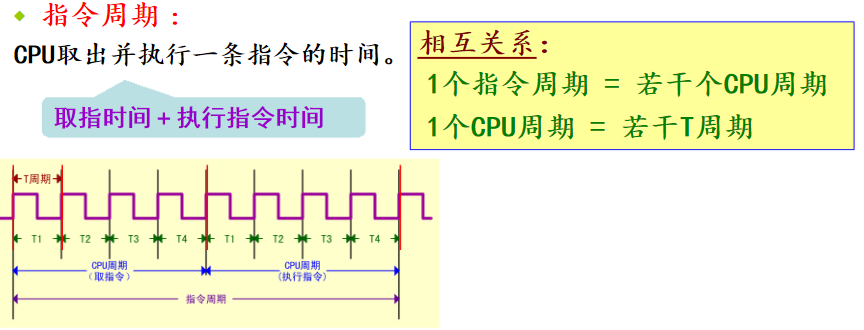


答案：

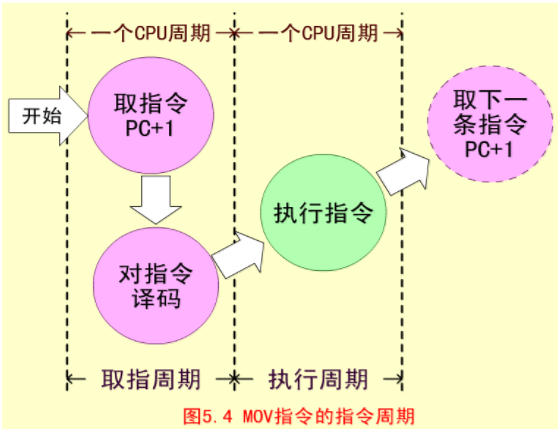


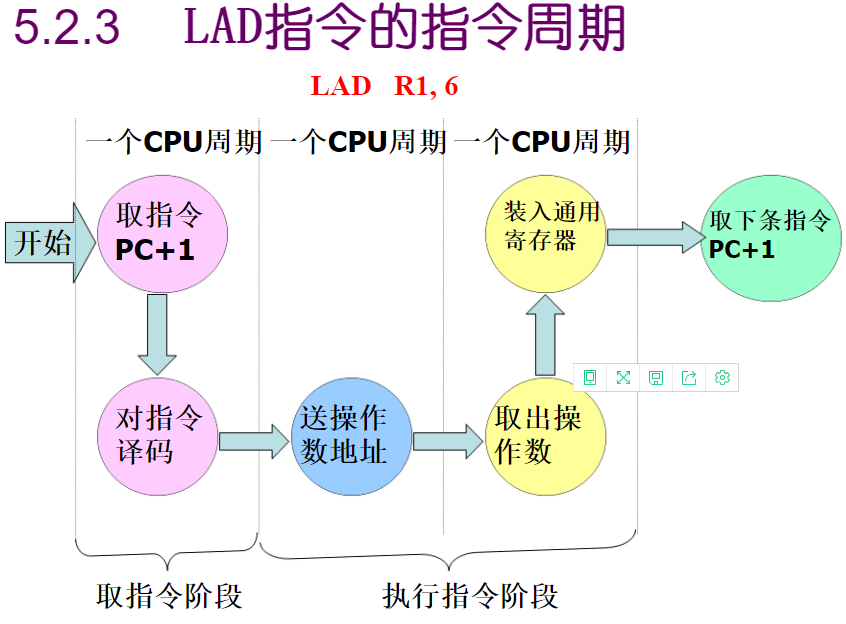
注：有+1的是PC，与操作控制器相连是IR，与主存储器相连双向箭头是DR，单向箭头是AR，有进入ALU的箭头的是AC，有从ALU出来的箭头指向的是PSW。

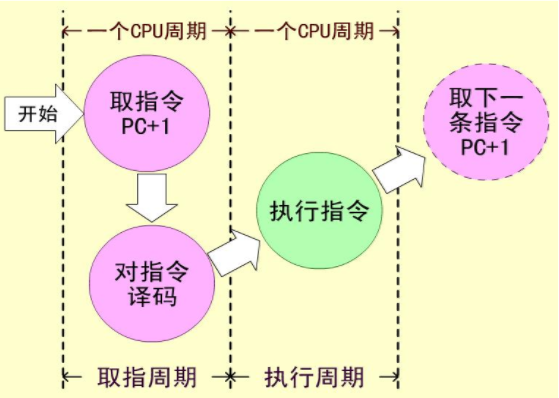
1. **指令周期**

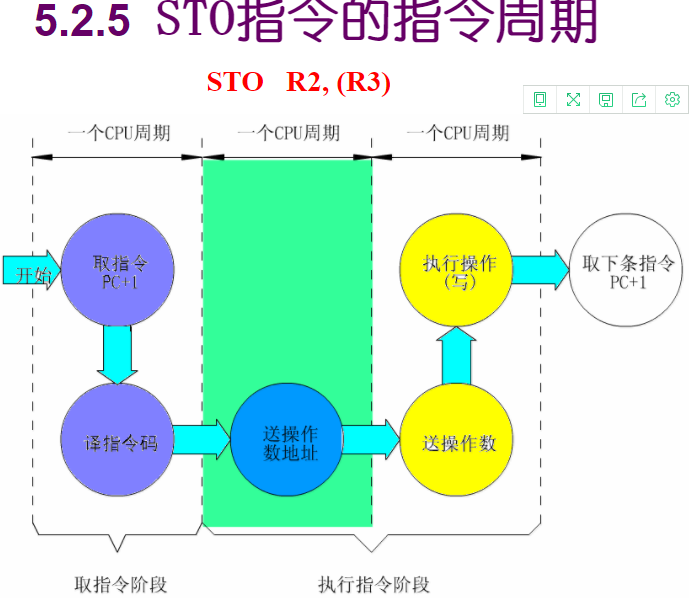


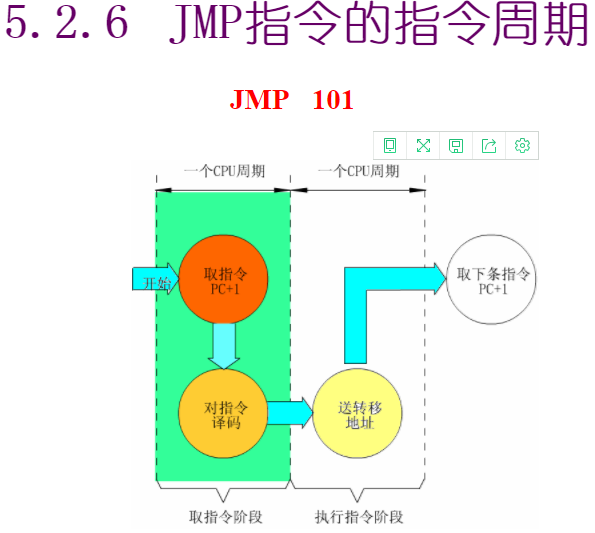
1. **控制信号、数据通路**



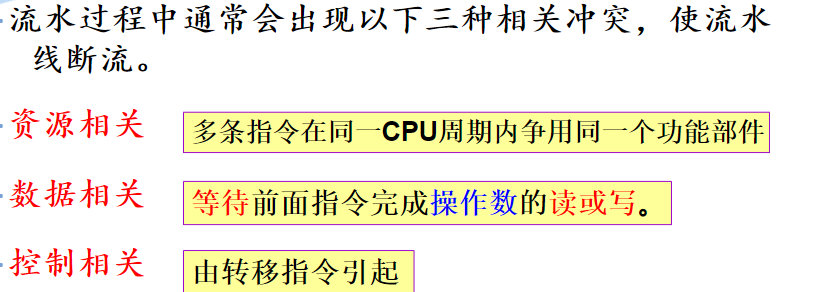




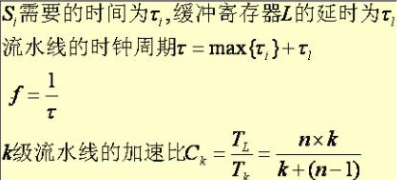




1. **流水中的问题**



1. **流水线的时钟周期**



1. **流水线的加速比**



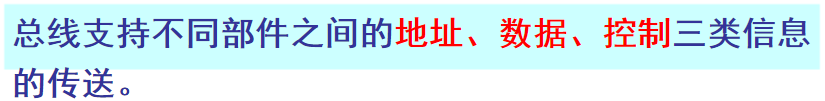
1. **程序、指令、微程序、微指令、微命令、微操作**

**程序**是由**指令**组成的，**指令**是指挥计算机执行各种操作的命令，一台计算机上，指令的全体(集合)称为指令系统。

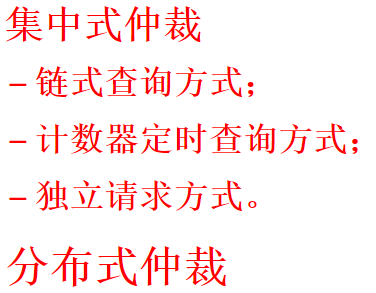
一条机器指令就是一个**微程序**，执行机器指令需要几个CPU周期就包含几个**微指令**，即一个CPU周期整体是一个**微指令**，微指令包含若干**微命令**，**微命令**发送给CPU内部部件执行**微操作**。

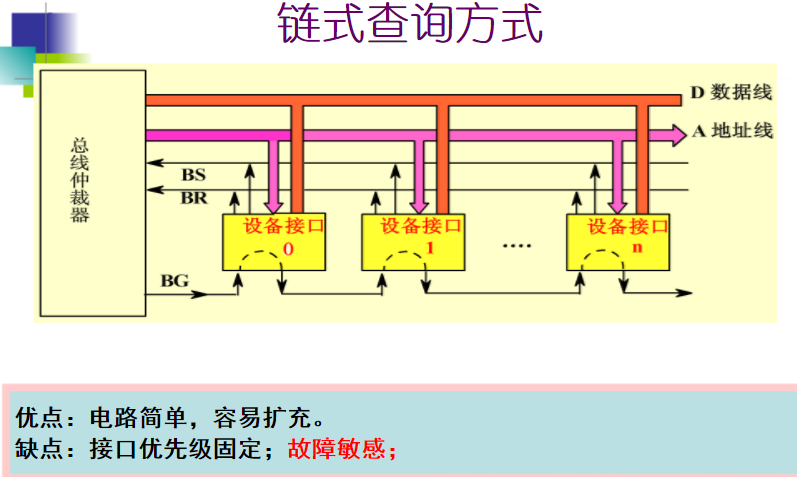
第六章

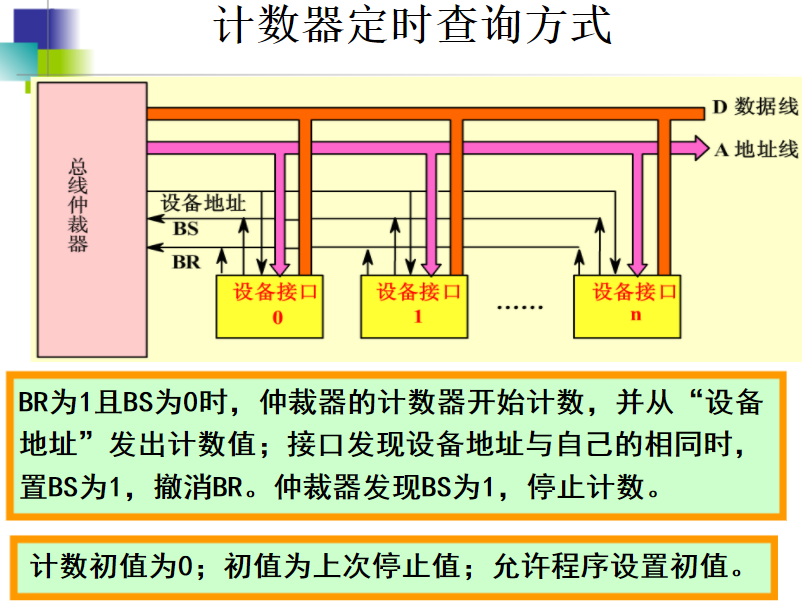
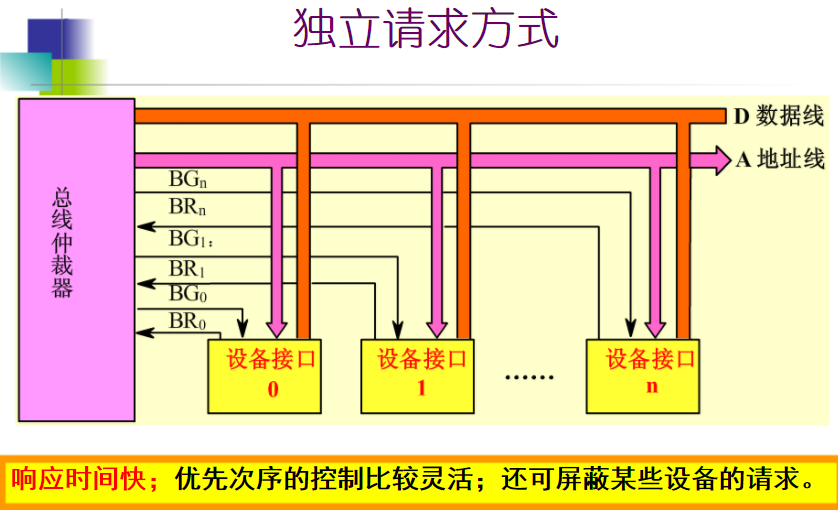
1. **总线的仲裁方式**



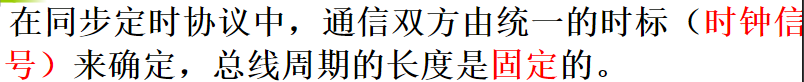
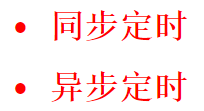


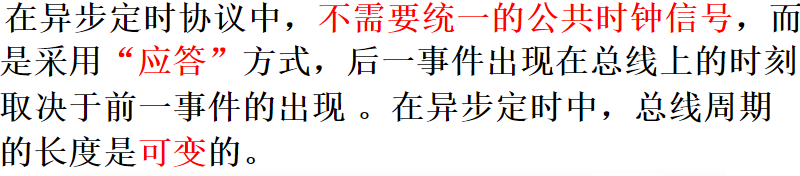






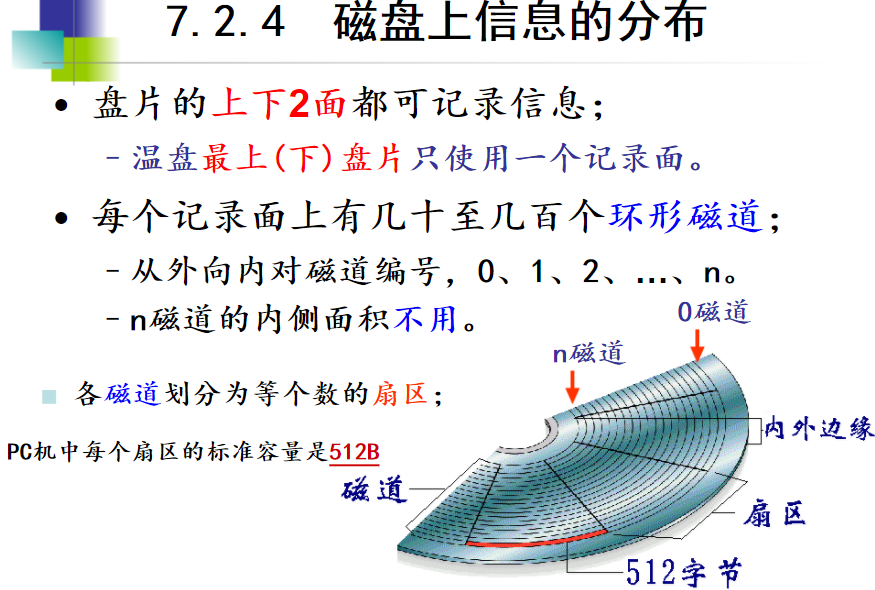
1. **总线定时的方式**



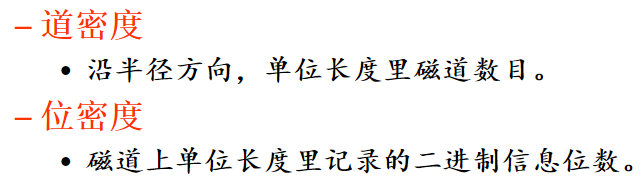
 

第七章

1. **磁盘盘片**

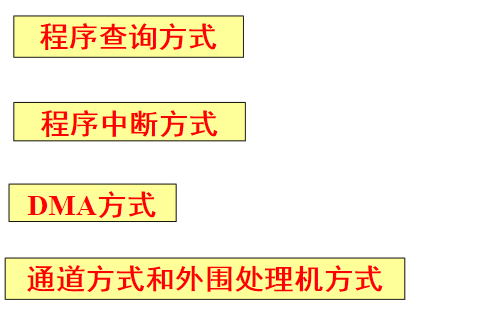


1. **磁道、扇区、道密度、位密度**



第八章

1. **信息交换方式**



eg：

