计算机组成作业

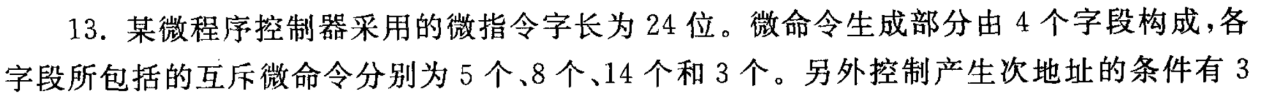
第六章 CPU设计



1. ****直接控制法（直接编码法）****：微指令的控制字段**每一位直接对应一个微命令**。一位为 “1”，就产生对应的微命令；为 “0” 则不产生，简单直观，微指令长，用于微命令少的情况。
2. ****字段编码法（分段编码法）****：把微命令分组（互斥微命令放一组），每组编码成一个字段，通过字段译码器译码产生微命令，可缩短微指令长度，有水平型（相容微命令分组，并行性强 ）和垂直型（类似机器指令操作码，一条微指令实现一种操作 ）之分。
3. ****混合编码法****：结合直接控制法和字段编码法，对部分重要、高频微命令直接控制，其他微命令字段编码，兼顾微指令长度和执行速度

**水平型，垂直型编码**

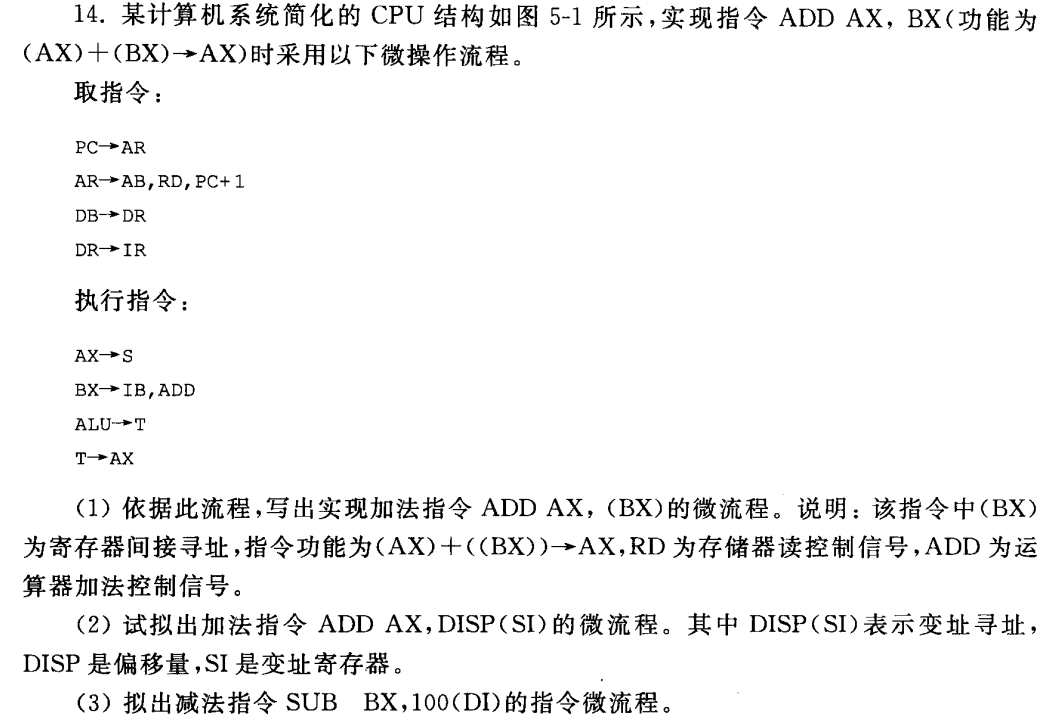
| ****维度**** | ****水平编码**** | ****垂直编码**** |
| --- | --- | --- |
| ****指令长度**** | 长（64 位 +），**包含所有操作细节** | 短（16/32 位），操作码**隐含逻辑** |
| ****译码复杂度**** | 无需 / 简单译码，直接对应硬件信号 | 需要复杂译码器解析操作码 |
| ****并行性**** | 高，可同时控制多个功能单元 | 低，一次指令通常仅执行单一操作 |
| ****编程难度**** | 高，需手动管理硬件细节 | 低，接近高级语言语义 |
| ****空间效率**** | 低（码长占用内存多） | 高（码长短，节省空间） |
| ****典型应用**** | DSP 微码、早期 CISC 指令、硬件底层控制 | x86/ARM 指令、RISC 架构、通用处理器 |



解析：微指令通常包含微命令生成部分和次地址部分 ，微命令生成部分包含的各字段长度不一定相同，会根据互斥微命令的数量来确定字段长度

3+4+4+2位控制条件，次地址有3种，占2位，所以地址剩24-15=9位

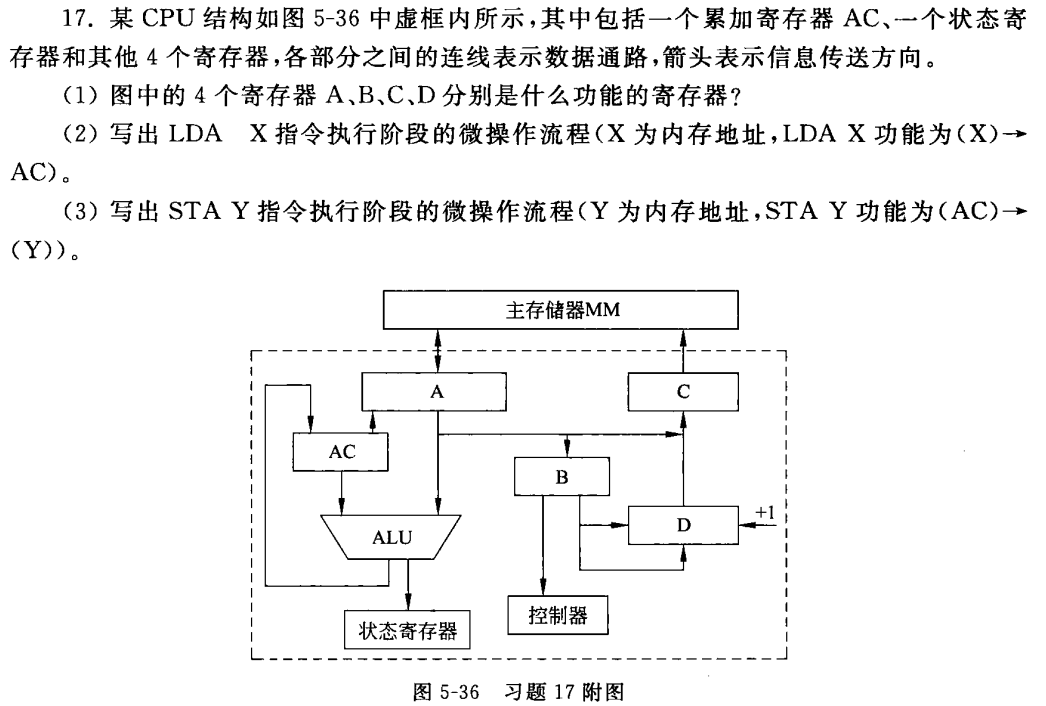
容量：一共能2^9 条指令，每条24位，所以一共1024\*24寄1K\*24位



取指令的过程是不会变的

**关于括号：**从内向外看！参考题目给的注释：（AX）表示，AX里面的东西；所以**（BX）**是BX里面的东西，（（BX））是（BX）里面的东西，也就是根据BX里面是新的地址，然后再根据地址来查

“功能为”（X）-> AC相当于 MOV AC， X！？



AC：暂存器

A：数据寄存器

C：地址寄存器【单向】

**B：IR**

D：控制

第七章 I/O系统



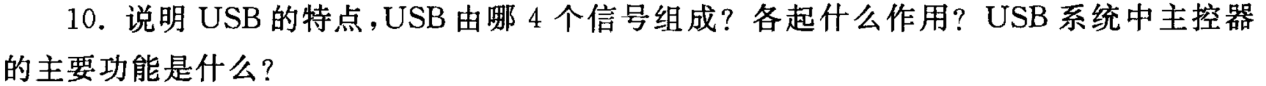
**PCI特点：同步时钟、高性能、即插即用、允许多负载、允许不同电压设备**

PCI和PCIE两大类

PCI：并行，集中式仲裁

PCIE：串行，交换机架构

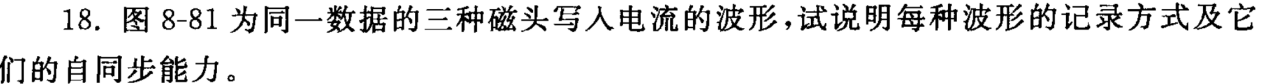
插上以后，系统自动完成识别、连接、配置等操作，无需手动配置

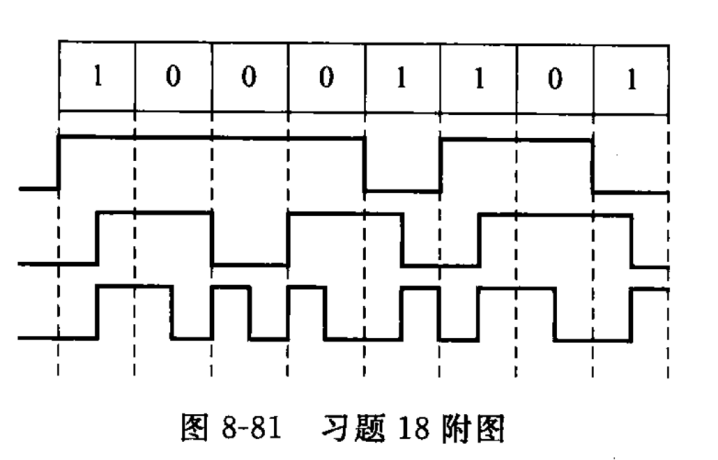


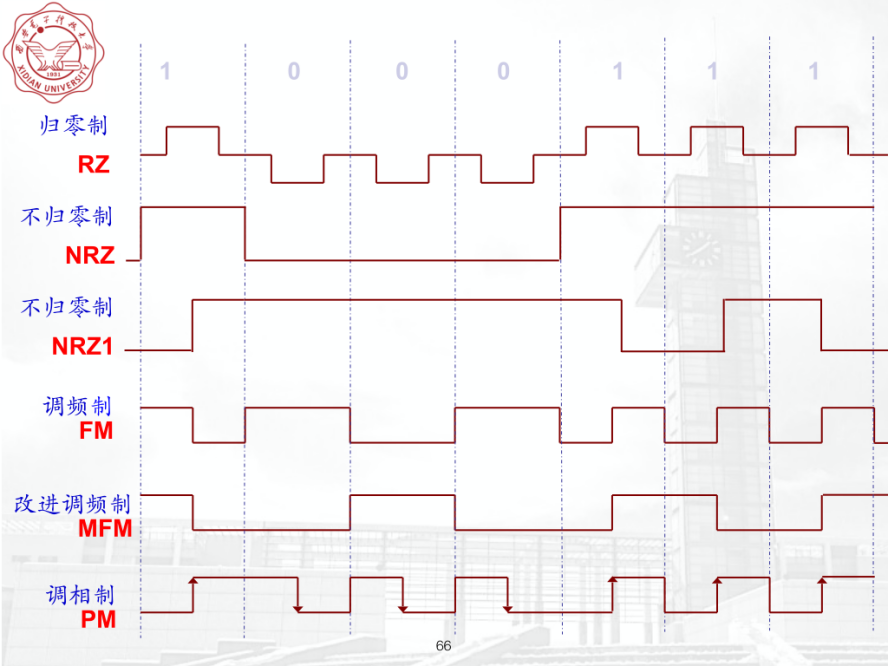
特点：热插拔，即插即用、统一接口、支持多种传输速率、可扩展

四个信号：VBus（供电），GND（地线），D+，D-（配合着数据传输）

USB主控器作用：识别和管理USB设备；管理数据和电流传输；向主机汇报





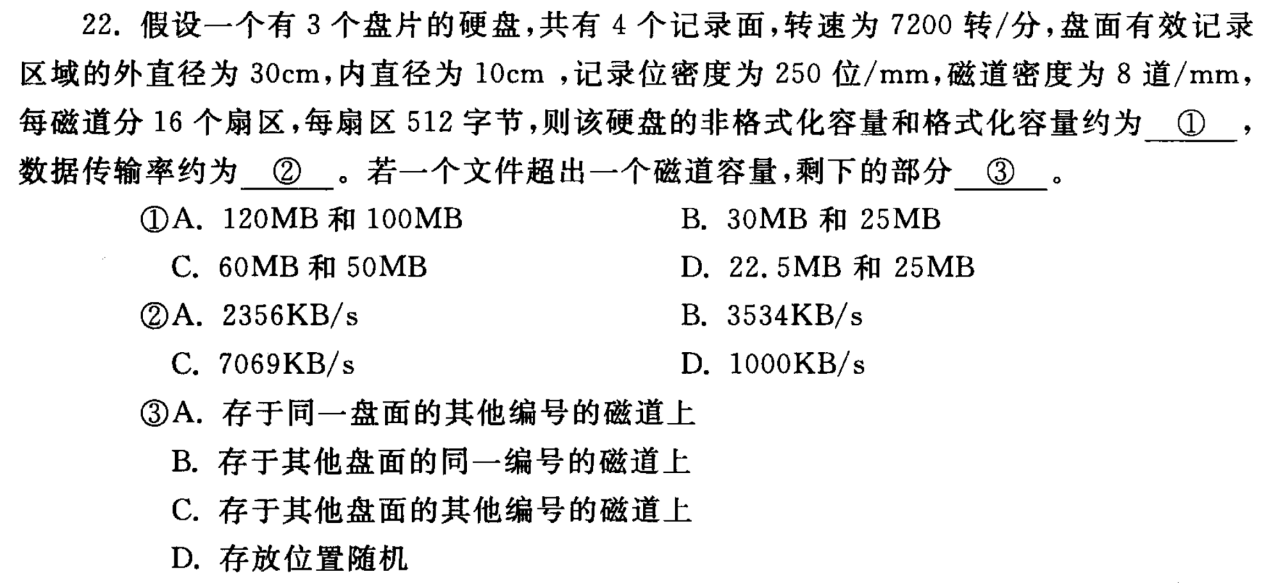


区分直径半径

区分字节和B和bit！**1B = 8bit!**

盘片，记录面

文件超出容量同磁道存储！



？非格式化容量2pai\*10^7是60MB？

