**复 习**

#### 第一章

1. **计算机系统的组成**

**（1）软件系统**

基本的软件系统包括**系统软件**和**应用软件**两大类。

**（2）硬件系统：**

五大部件：输入设备、输出设备、存储器、运算器、控制器

总线

**（3）冯·诺依曼思想**

**1）采用二进制形式表示数据和指令：**指令是程序的基本单位，程序是若干指令的有序集合

**2）采用存储程序方式：**存储程序方式是冯·诺依曼思想的核心，是计算机能高速自动运行的基础

**存储程序方式：**在用计算机解题之前，事先编制好程序，并连同所需的数据预先存入主存储器中。在解题过程(运行程序)中，由控制器按照事先编好并存入存储器中的程序自动地、连续地从存储器中依次取出指令并执行，直到获得所要求的结果为止。

**3）计算机由输入设备、输出设备、运算器、存储器和控制器五大部件组成：**运算器+控制器=CPU；CPU+主存储器=主机；输入设备+输出设备+外存储器=外设（I/O设备）

**（4）现代计算机的结构**

以存储器为中心（冯诺依曼计算机是以运算器为中心）

1. **总线：概念、特点、分类、结构**

**（1）概念**：

一组可为多个功能部件共享的公共信息传送线路。

**（2）特点：**

单发，多收

**（3）分类：**

任务分：1）CPU内部总线；2）部件内总线；3）系统总线（地址线、数据线、控制线）；4）外总线

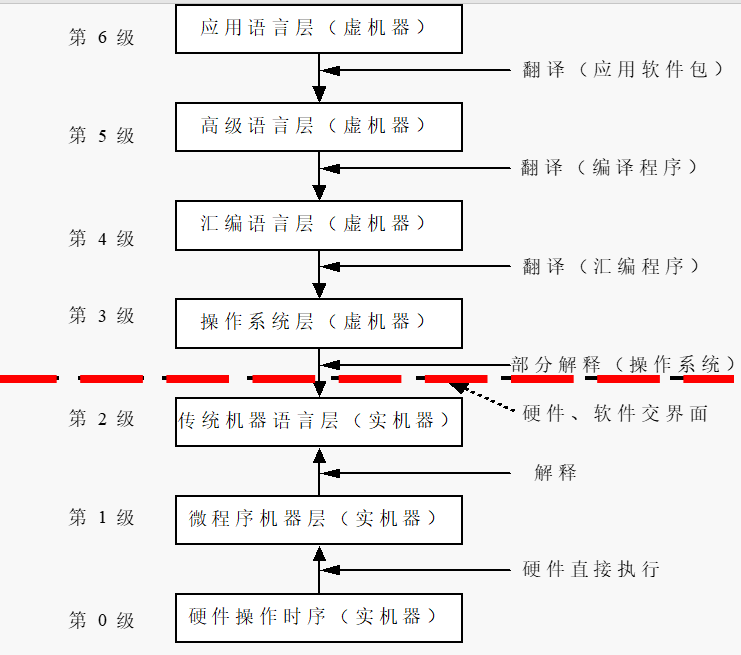
传输方向分：1）单向总线；2）双向总线

**（4）结构：**

1）单总线结构；2）双总线结构（以CPU为中心；面向存储器）；3）三总线结构；4）多总线结构

1. **计算机系统的层次结构**

对不同的对象而言，一个计算机系统就成为实现不同语言的、具有不同属性的机器。



1. **软件与硬件的关系**

计算机系统由硬件、软件两大部分组成。硬件和软件是紧密相关、缺一不可的整体。

硬件是计算机系统的物质基础

软件是计算机系统的灵魂

硬件、软件在逻辑功能上等效（软件“固化”甚至“硬化”；硬件“软化”）

1. **计算机系统在速度、容量、价格方面的主要性能指标（MIPS，Mflops，CPI，CPU时间）**

（1）基本字长（标志着计算机的运算精度；要保证 i 位十进制数的精度，至少要采用3.3倍 i 位二进制数的位数）

（2）主存容量

**（3）运算速度：（公式见ppt，重点！）**

MIPS：每秒百万条指令；Mflops：每秒百万次浮点运算；CPI：执行每条指令所需的平均时钟周期数；CPU时间：

1. **计算机系统的分类**

（1）按处理的信息形式分：1）电子数字计算机；2）电子模拟计算机

（2）按计算机字长分：如64位机、32位机

（3）按计算机应用范围分：1）专用机；2）通用机

（4）按计算机规模分：巨型机、大型机、中型机、小型机、微型机、单片机等等

**重点：**

计算机系统的组成的概念；各部件的作用；冯·诺依曼思想；计算机系统的层次结构的概念；软件与硬件逻辑上等效的概念；计算机系统主要性能指标的计算。

**出题形式：**

填空、选择、判断、简答

#### 第二章

1. **数制及数制转换**
2. **带符号数的表示：原码、反码、补码、移码以及各种码制与真值之间的转换**



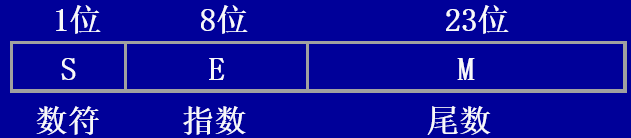
1. **定点数的表示格式和数据表示范围**

**浮点数的表示格式和数据表示范围、浮点数的规格化数、原码／补码的规格化规则**

1. **IEEE 754标准浮点数的表示格式（32位单精度）、与真值之间的转换规则**

M为原码，E为移127码



若E＝0，且M＝0，则 N为0（有±0）。

若E＝0，且M≠0，则 N＝(-1)S·2-126·(0.M)。为非规格化数。

若1≤E≤254，则 N＝(-1)S·2E-127·(1.M)。为规格化数。

若E＝255，且M≠0，则 N＝NaN (“非数值”)。

若E＝255，且M＝0，则 N＝(－1)S\*∞ (无穷大)。

1. **非数值符号的表示：字符的ASCII码、字符串的存放方法；汉字编码的关系、汉字字库的容量计算**

**（1）ASCII码共7位**

**（2）国标码和区位码的转化：**区位码的区号和位号各加32（+ 00100000）

1. **数据长度与存储方式；C语言简单数据类型的长度与存储**

**（1）大端排序和小端排序：**

**大端排序方式（big endian）：**数据的最高有效字节MSB存放在低地址单元中，最低有效字节LSB放在高地址单元中。

**小端排序方式（little endian）：**数据的最低有效字节LSB存放在低地址单元中，最高有效字节MSB存放在高地址单元中。

1. **奇偶校验码、海明校验码、循环校验码的编码方法和校验方法；海明校验码码长的计算公式、各种校验码的检错和纠错能力；校验码的检错、纠错能力与码距的关系。**

**（1）记码距为d，码距与校验码的检错和纠错能力的关系是：**

d≥e＋1 可检验e个错。

d≥2t＋1 可纠正t个错。

d≥e＋t＋1 且e＞t，可检e个错并能纠正t个错。

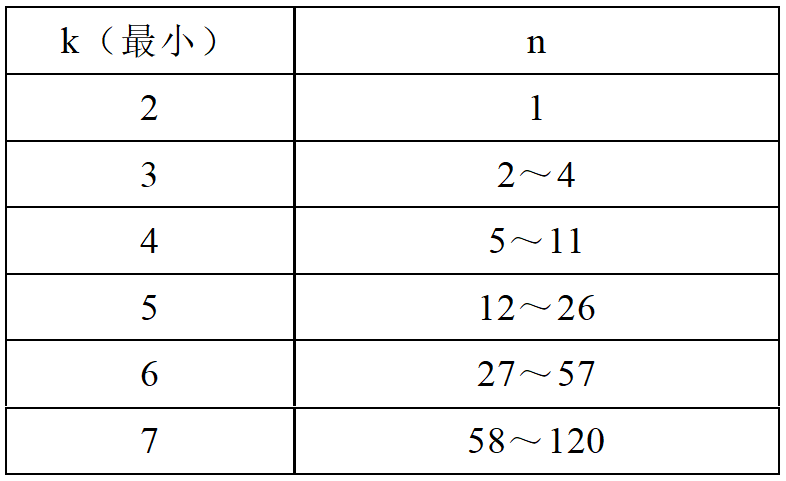
**（2）奇偶校验码：**

**1）概念：**在n位有效信息位上增加一个二进制位作为校验位P，构成n十1位的奇偶校验码。**奇校验Odd：**校验位P的取值（0或1）使n＋1位的奇偶校验码中“1”的个数为奇数；**偶校验Even：**校验位P的取值（0或1）使n＋1位的奇偶校验码中“1”的个数为偶数。

**2）检错能力：**只能发现奇数个错误，且不能发现错误位置，也不能纠错。

**（3）海明校验码：**

**1）概念：**设有效信息位的位数为n，校验位数为k，则能够检一纠一错的海明校验码应满足下面的关系：2k≥n＋k＋1



**2）检错能力：**检一纠一错，两个及以上的错误越纠越错

**（4）扩展海明校验码：**

**1）概念：**为了满足检二纠一错的要求，可将检一纠一错海明校验码再进行奇偶校验。

**2）检错方式：**

E0＝0，Ek…E1＝00…0 表示无错。

E0＝1，Ek…E1≠00…0 表示有一位出错，可根据Ek…E1的值确定出错位号并自动纠正。

E0＝0，Ek…E1≠00…0 表示有两位出错，此时无法确定出错位置，也无法纠错。

E0＝1，Ek…E1＝00…0 表示P0出错，将P0取反，即可自动纠正错误。

**3）检错能力：**检2个错或纠正1个错。

**（5）循环校验码：**

**1）位数关系：**k+1位生成多项式，有效信息左移k位（加k个0），余数k位

**2）生成多项式要求：**

（i）任何一位发生错误都应使余数不为0；

（ii）不同位发生错误应当使余数不同；

（iii）对余数作模2除法，应能使余数循环。

**重点：**

原码、反码、补码、移码以及各种码制与真值之间的转换方法；浮点数的规格化数、最大、最小数的表示；IEEE 754标准浮点数的表示规则、表示格式、IEEE 754标准浮点数与真值之间的转换；奇偶校验码、海明校验码（检一纠一、检二纠一）、循环校验码的编码方法；校验码的检错、纠错能力与码距的关系、海明校验码码长的计算公式、循环校验码的编码方法和校验方法、生成多项式的特点及对生成多项式的要求；高级语言的简单数据类型在机器中的存储。

**出题形式：**

填空、选择、判断、计算★

#### 第三章

* 1. **定点补码加减运算规则、溢出判断方法、定点补码加减运算的逻辑电路、算术逻辑运算部件的工作原理**
  2. **一位原码/补码的乘法运算规则以及乘法运算的硬件逻辑电路的结构和工作原理**
  3. **一位原码/补码不恢复余数除法运算规则、布斯除法运算规则以及除法运算的硬件逻辑电路的结构和工作原理**

**（1）原码不恢复余数除法中：**

**1）符号规则：**余数符号与被除数相同，商符号由被除数符号与除数符号异或得到；

**2）**当最后一步得到的余数为负时，要加上除数y恢复余数。

**（2）补码不恢复余数除法中：**

**1）上商规则：**每次加减所得的余数与除数**同号**时，**上商为1**；每次加减所得的余数与除数**异号**时，**上商为0**；

**2）余数加减规则：**每次加减所得的余数与除数同号时，将余数左移一位，减去除数；每次加减所得的余数与除数异号时，将余数左移一位，加上除数

**3）商的校正、余数恢复：**最后商若为负，则需要校正（末位横置1、校正法）；**若采用校正法，最后的余数与被除数异号则需要对余数进行恢复。**

**（3）在步斯除法中：**

**1）**最后要将商符取反，**若采用矫正法同样要对商进行校正，对余数进行恢复**

* 1. **浮点四则运算的方法和步骤**

**加减运算：求阶差、对阶、尾数加减、结果规格化、尾数的舍入规则**

**乘除运算：阶码加减、尾数乘除、结果规格化、尾数的舍入规则**

* 1. **逻辑运算、各类移位的移位规则**
  2. **AM2901A（4位ALU）芯片的功能及应用**

**重点：**

定点补码加减运算、溢出判断方法；一位原码、补码的乘法运算及硬件逻辑电路的结构；一位布斯除法的运算规则及除法运算的硬件逻辑电路；浮点四则运算的方法和步骤；算术、逻辑、循环移位的移位方法。

**出题形式：**

填空、选择、判断、计算★（结合定点加减乘除算法，进行浮点四则运算）

#### 第四章

1. **存储器的基本组成：存储体、读写控制、地址寄存及译码电路、数据寄存器**

**（1）存储元件（存储元、存储位）：**能够存储一位二进制信息的物理器件

**（2）存储单元：**由一组存储元件组成，可以同时进行读写

**（3）存储体（存储阵列）：**把大量存储单元电路按一定形式排列起来，即构成存储体。存储体一般都排列成阵列形式，所以又称存储阵列

**（4）地址寄存器：**用于存放所要访问的存储单元的地址

**（5）地址译码与驱动电路：**用于对地址寄存器中的地址进行译码，通过对应的地址选择线到存储阵列中找到所要访问的存储单元，并提供驱动信号驱动其完成指定的存取操作

**（6）读写电路：**根据CPU发出的读写控制命令，控制对存储单元的读写

**（7）数据寄存器：**暂存需要写入或读出的数据

**（8）时序控制电路：**用于接收来自CPU的读写控制信号，产生存储器操作所需的各种时序控制信号，控制存储器完成指定的操作

1. **存储系统的层次结构，程序局部性原理**

**（1）存储系统层次结构**：CPU-Cache-主存-硬盘-光盘、磁带（速度逐渐变慢，每位价格逐渐降低，存储容量逐渐变大）

**（2）程序局部性原理：**CPU在一个较短的时间间隔内，由程序产生的地址往往聚集在一个很小的区域内。

1. **主存与CPU之间数据传送的控制方式：同步控制、异步控制**

**（1）同步控制方式：**数据传送在固定的时间间隔内完成，即在一个存取周期内完成。（多用于CPU与主存）

**（2）异步控制方式：**数据传送的时间不固定，存储器在完成读/写操作后，需向CPU回送“存储器功能完成”信号（MFC），表示一次数据传送完成。（多用于CPU与外设）

1. **主存的主要性能指标：容量、速度（存取时间TA、存取周期TM、带宽的计算方法）、价格**

**（1）存储容量表示方式：**存储单元数与每个单元的位数的乘积表示（2地址位数×数据位数）

**（2）速度：**

**1）存取时间TA：**从启动一次存储器存取操作到完成该操作所需的全部时间。（即从存储器接到CPU发出的读/写命令和地址信号到数据读入MDR/从MDR写入MEM所需的时间）

**2）存取周期TM：**TM＝TA＋一定的恢复时间

**3）带宽：**带宽是指存储器单位时间内所存取的二进制信息的位数。带宽等于存储器总线宽度（数据总线根数）除以存取周期。Bm=W/TM（**注意区分存储器带宽与总线带宽！**）

**（3）价格：**设存储器容量为S位，总价格为C总，每位价格为c＝C总/S

1. **半导体存储器的存储原理**

**（1）SRAM、DRAM的存储原理：**SRAM是靠触发器电路存储信息的，所存的信息表现为双稳态电路的电平，所以不需要刷新；DRAM是靠电容存储信息的，所存信息表现为电容上的电荷，需要刷新。

**（2）DRAM的刷新方式：**按行刷行

**1）刷新最大周期（刷新最大间隔）：**Δt=C\*(Δu/I) （Δu为电容两端的电压变化；I为泄露电流；C为存储电容）

**（3）如何计算DRAM的刷新周期：**

1）集中式刷新 2）分散式刷新 3）异步式刷新 4）透明式刷新

1. **存储器与CPU的连接：芯片数的计算、地址、数据、控制线的连接、片选信号的产生、地址范围的确定（字扩展、位扩展）；当需要多种字长访存时的各种地址和片选信号的实现，数据的整数边界存储问题。地址线参与片选控制的安排**
2. **Cache的工作原理：Cache的命中率的计算，Cache－主存系统的平均访问时间的计算**

**（1）Cache工作原理：**根据程序局部性原理，把这一局部区域的程序和数据从主存复制到Cache中，使CPU能够高速地在Cache中读取指令和数据，就可大大提高CPU的访存速度。

**（2）Cache命中率：**H=Nc/(Nc+Nm) （访问Cache 的总命中次数为Nc，访问主存的次数为Nm，CPU访问Cache的命中率为H）

**（3）平均访问时间：**Ta＝H Tc＋(1－H) Tm（Tc为Cache的存取时间，Tm为主存的存储周期，H为Cache命中率，Ta为Cache－主存系统的平均访问时间）

**（4）Cache存储系统加速比Sp：Sp=** Tm/ Ta

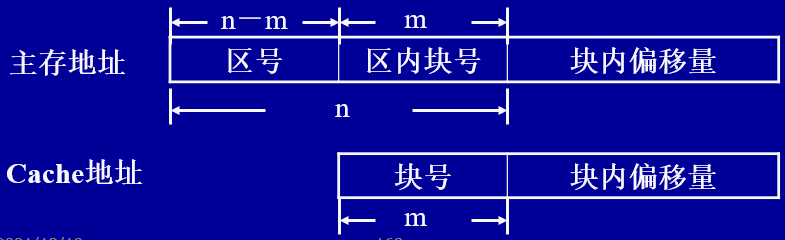
**（5）Cache主存系统的访问效率e：e=** Tc/ Ta

1. **主存与Cache之间的三种地址映射方式的实现原理和特点； Cache容量的计算（数据Cache和标识Cache的容量计算）**

**（1）直接映像方式：**直接映像是指任何一个主存块只能复制到Cache的某一固定块中。

**1）**Cache块号i和主存块号j采用十进制统一编号，则i与j的对应关系为：i＝j mod 2m

**2）地址格式：**



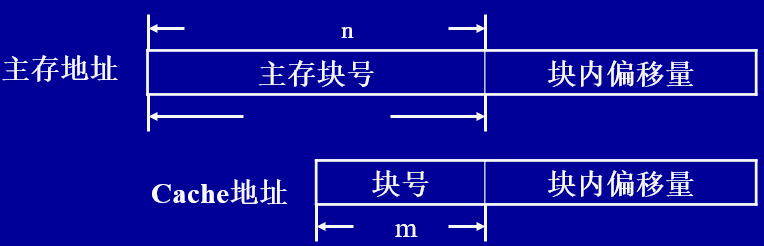
（其中，2m=Cache块数，2块内偏移量位数=Cache每块大小）

**3）标识Cache容量：**2m×(n-m+1)（即块数×(标识tag位数+有效位位数)）

**（2）全相联映像方式：**任何主存块可映像到任意一个Cache块

**1）**在全相联映像中，主存块号又称为**标识**，块内偏移量又称为**索引**。

**2）地址格式：**



（其中，2m=Cache块数，2块内偏移量位数=Cache每块大小）

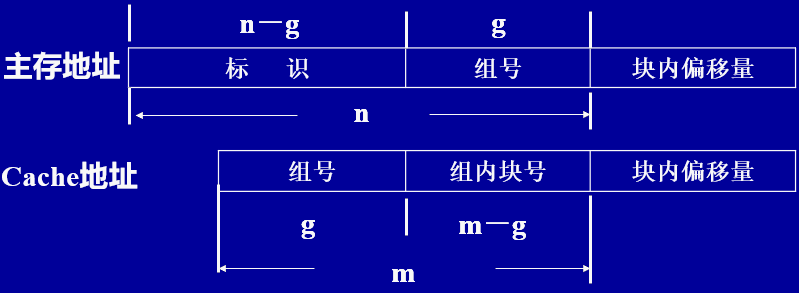
**3）标识Cache容量：**2m×(n+1)

**（3）组相联映像方式：**前两种方式的一种折衷。主存中的任何一块只能存放到Cache中的某一固定组中，但存放在该组的哪个位置是灵活的。

**1）**主存某个块映像到Cache对应组的公式：Cache组号＝主存块号 MOD Cache组数

**2）**如果一个**组里有k块**，则这种组相联映像方式称为**k路组相联**。

**3）地址格式：**



（其中，2m=Cache块数，2块内偏移量位数=Cache每块大小）

**4）标识Cache容量：**2m×(n-g+1)

1. **辅助存储器**

**磁表面存储器的存储原理、各种记录方式的特点、各种记录方式的评价标准（自同步能力、编码效率）**

**磁盘的常用技术指标：容量、平均存取时间、数据传输率的计算**

**（1）磁盘三维地址：**圆柱面号（磁道号）+盘面号+扇区号（磁盘信息的基本单位是扇段）

**（2）磁盘存储容量：**

**1）非格式化容量：**一个磁盘组有n个盘面存储信息，每个面有T条磁道，每条磁道分成S个扇段，每段存放B个字节，则存储容量C为：C＝n×T×S×B

**2）格式化容量：**在磁盘上划分记录区、写入各种标志信息和地址信息。一般为非格式化容量的60%-70%

**（3）平均存取时间：**平均存取时间=平均寻道时间＋平均旋转等待时间（转半圈所需时间）＋读写操作时间＋控制延时时间

**（4）数据传输率：**

设磁盘旋转速度为n转/秒，每条磁道容量为N个字节，则  
 Dr＝n×N(字节/秒)。

设D为位密度，v为磁盘旋转的线速度，则  
 Dr＝D×v(字节/秒)

1. **存储体系中单体多字并行存储器、****多体交叉存储器的概念、**

**多端口存储器及应用、单体多字存储系统、多体交叉存储系统**

**（1）多端口存储器：**

**1）单端口存储器：**单端口存储器每次只接收一个地址，访问一个编址单元，从中读取或存入一个字节或一个字。

**2）双端口存储器：**双端口存储器具有两个彼此独立的读/写口，每个读/写口都有一套独立的地址寄存器和译码电路，可以并行地独立工作。可以访问同一区间、同一单元。

**（2）单体多字存储系统：**多个并行存储器共用一套地址寄存器，按同一地址码并行地访问各自的对应单元。（单体多字并行主存系统适用于向量运算一类的特定环境）

**（3）多体交叉存储系统：**使用n个容量相同的存储器，或称为n个存储体，它们具有自己的地址寄存器、数据线、时序，可以独立编址地同时工作。各存储体的编址大多采用**交叉编址方式**，即将一套统一的编址，按序号交叉地分配给各个存储体。（适合支持流水线工作方式）

**（4）相联存储器（Associative Memory）：**又称为联想存储器。相联存储器是根据所存信息的全部特征或部分特征进行存取的，是一种**按内容寻址**的存储器。

**重点：**

存储器的基本组成；DRAM的刷新；主存的主要性能指标的计算；存储器与CPU的连接及整数边界存储；Cache容量的计算；三种地址映射方式下主存块与Cache块的对应关系；多体交叉存储器带宽的计算。

**出题形式：**

填空、选择、判断、问答、计算、设计★（存储器与CPU的连接，多种字长访存时的各种地址和片选信号的实现）

#### 第五章

1. **指令的格式、指令中地址的格式、操作码的编码方式**

**根据操作码和地址码计算指令的条数**

1. **寻址方式**

**各种寻址方式中有效地址的计算方法，RISC系统的寻址方式（寄存器直接寻址、立即数寻址、相对寻址、基址寻址）**

**（1）立即寻址：**立即寻址方式是指指令的地址码部分给出的不是操作数的地址而是操作数本身。

**（2）直接寻址：**操作数的有效地址在指令字中直接给出，指令的地址码部分给出的形式地址A就是操作数的有效地址EA。

**（3）间接寻址：**指令的地址码部分给出的是操作数的有效地址EA所在的存储单元的地址或是指示操作数地址的地址指示字。

**（4）寄存器直接寻址：**指令地址码中给出的是某一通用寄存器的编号（也称寄存器地址），该寄存器的内容即为指令所需的操作数。

**（5）寄存器间接寻址：**指令中地址码部分所指定的寄存器中的内容是操作数的有效地址。

**（6）变址寻址：**操作数的有效地址是由指令中指定的变址寄存器的内容与指令字中的形式地址相加形成的。

**（7）相对寻址：**程序计数器 PC 的当前内容与指令中给出的形式地址相加形成操作数的有效地址。

**（8）基址寻址：**操作数的有效地址等于指令中的形式地址与基址寄存器中的内容之和。

**（9）基址加变址寻址：**将两个寄存器的内容和指令形式地址中给出的偏移量相加后得到的结果作为操作数的有效地址。其中一个寄存器作为基址寄存器，另一个作为变址寄存器。

**（10）基址寻址与变址寻址差距：**

**1）**基址寻址是面向系统的，主要用于逻辑地址到物理地址的变换，用以解决程序定位问题。其对用户是透明的，且基址寄存器的内容一般不进行自动增量和减量。

**2）**变址寻址是面向用户的，主要用于访问数组、向量、字符串等成批数据，用以解决程序的循环控制问题。变址寄存器的内容可以进行自动增量和减量。

1. **指令类型**

**完备的指令系统应具有的基本指令类型，各种指令的实现过程。**

**（1）一个完善的指令系统应包括的基本指令有：**数据传送指令、算术逻辑运算指令、移位操作指令、堆栈操作指令、字符串处理指令、程序控制指令、输入/输出指令等。

1. **CISC和RISC系统的设计风格的特点**

**（1）CISC（Complex Instruction Set Computer）复杂指令系统计算机：**具备庞大且复杂的指令系统的计算机，简称CISC。

**（2）CISC指令系统的特点 ：**

1)指令系统复杂庞大，指令数目一般多达200～300条。

2)指令格式多，指令字长不固定，使用多种不同的寻址方式。

3)可访存指令不受限制。

4)各种指令的执行时间和使用频率相差很大。

5)大多数采用微程序控制器。

**（3）RISC（Reduced Instruction Set Computer）精简指令系统计算机：**用一套精简的指令系统取代复杂的指令系统，使机器结构简化，以达到用简单指令提高机器性能和速度、提高机器的性能价格比的目的。

**（4）RISC指令系统的特点 ：**

1)选取使用频率高的简单指令以及很有用但又不复杂的指令组成指令系统。

2)指令数目较少，指令长度固定，指令格式少，寻址方式种类少。

3)采用流水线技术，大多数指令可在一个时钟周期内完成；特别是在采用了超标量和超流水技术后，可使指令的平均执行时间小于一个时钟周期。

4)使用较多的通用寄存器以减少访存。

5)以寄存器 — 寄存器方式工作，只有取数/存数(LOAD/STORE)指令访问存储器，其余指令的操作都在寄存器之间进行。

6)控制器以组合逻辑控制为主，不用或少用微程序控制。

7)采用优化编译技术，力求高效率支持高级语言的实现。

1. RISC系统指令类型、格式及特点

**重点：**

指令格式；寻址方式；各种指令的实现过程；CISC和RISC系统的设计风格的特点。

**出题形式：**

填空、选择、判断、问答、设计★（指令格式设计；寻址方式分析；指令功能实现）

#### 第六章

1. **控制器的功能:**

（1）根据事先编好并存放在存储器中的解题程序，控制各部件有条不紊地、自动协调地进行工作。

（2）控制器是计算机的指挥和控制中心，由它把计算机的运算器、存储器、I/O设备等联系成一个有机的系统，并根据各部件具体要求，适时地发出各种控制命令，控制计算机各部件自动、协调地进行工作。

1. **指令的执行步骤：**

取指令 -> 分析指令 -> 执行指令

1. **控制器组成部件：PC、IR、ID、操作信号形成部件等**

**（1）指令部件：**

**1）程序计数器 PC（指令计数器、指令地址寄存器）：**PC用于保证程序按规定的序列正确运行，并提供将要执行指令的指令地址。PC的位数应与MAR位数保持一致。PC的增量可以通过PC本身实现，也可以通过ALU实现

**2）指令寄存器 IR：**指令寄存器用于存放当前正在执行的指令。

**3）指令译码器 ID：**指令译码器是指令分析部件，对指令寄存器中的指令**操作码**进行译码分析，产生相应操作的控制电位，提供给微操作控制信号形成部件。对**寻址方式字段**进行译码分析，以控制操作数有效地址的形成。

**4）地址形成部件：**根据机器所规定的各种寻址方式，形成操作数有效地址。

**（2）时序控制部件：**用于产生一系列时序信号，为各个微操作定时，以保证各个微操作的执行顺序。（微操作是机器最简单的基本操作）

**（3）微操作控制信号形成部件：**根据指令部件提供的操作控制电位、时序部件所提供的各种时序信号以及有关的状态条件，产生机器所需要的各种微操作控制信号。

**（4）中断控制逻辑：**用于实现异常情况和特殊请求的处理。

**（5）程序状态寄存器PSR：**用于存放程序的工作状态（如管态、目态等）和指令执行的结果特征(如结果为零、结果溢出等)，把它所存放的内容称为程序状态字（PSW）

1. **控制器的组成方式：组合逻辑方式、微程序方式**

控制器的组成方式主要是指**微操作控制信号形成部件**采用何种组成方式产生微操作控制信号

**（1）组合逻辑方式：**组合逻辑控制器也称为**硬联逻辑**或**硬布线逻辑**。组合逻辑控制器是根据控制要求和状态，采用组合逻辑技术来实现的。其微操作信号发生器是由门电路组成的复杂树形网络构成的。（RISC机和巨型机采用）

**（2）存储逻辑方式：**存储逻辑型控制器称为**微程序控制器**。它是采用**存储逻辑**来实现的。把微操作信号代码化，使每条机器指令转化成为一段微程序存入控制存储器中。执行指令时，读出控存中的微指令，由微指令产生微操作控制信号。

1. **控制器的控制方式：同步控制、异步控制、联合控制**

**（1）同步控制：**任何指令的运行或指令中各个微操作的执行，均由确定的具有统一基准时标的时序信号所控制。即所有的操作均由统一的时钟控制，在标准的时间内完成。

**（2）异步控制：**没有统一的同步信号，采用问答方式进行时序协调，将前一操作的回答信号作为下一操作的启动信号。

**（3）联合控制：**在**功能部件内部**采用**同步方式**或以同步方式为主的控制方式；在**功能部件之间**采用**异步方式**。

**控制器的时序：指令周期、机器周期、节拍、脉冲**

**（1）指令周期：**从取指令、分析指令到执行完一条指令所需的全部时间。（不同指令其指令周期也不同）

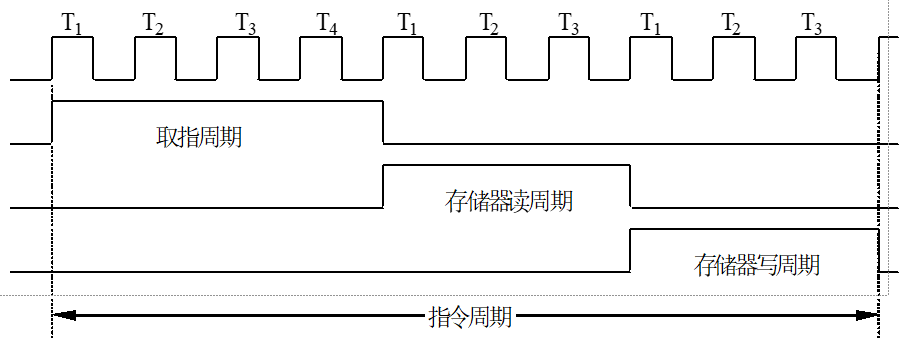
**（2）机器周期（CPU周期）：**指令周期中的某一工作阶段所需的时间。在指令执行过程中，各机器周期相对独立。一条指令的指令周期由若干个机器周期所组成，每个机器周期完成一个**基本操作**。所以机器周期也称为**基本周期**。一般机器的CPU周期有**取指周期**、**取数周期**、**执行周期**，**中断周期**等。

**（3）节拍：**把一个机器周期等分成若干个时间区间，每一时间区间称为一个节拍。一个节拍对应一个电位信号，控制一个或几个**微操作**的执行。

**（4）脉冲：**在一个节拍内，有时还需要设置一个或几个工作脉冲，用于寄存器的复位和接收数据等。

（**注：**指令和数据在CPU中的区分主要靠机器周期，取指周期取出的是指令，取数周期取的是数据。）

**时钟周期时序系统：指令周期-机器周期-时钟周期（一般微型机使用）**



1. **CPU的结构、CPU中的基本寄存器**

**（1）CPU包括：**运算器和控制器两大部分

**（2）CPU功能：**程序控制、操作控制、时序控制、数据加工

**（3）CPU的基本寄存器：**

1)指令寄存器 IR

2)程序计数器 PC

3)累加寄存器 AC

4)程序状态寄存器 PSR

5)地址寄存器 MAR

6)数据缓冲寄存器 MDR(或MBR)

7)堆栈指针SP

其中2）、3）、4）是用户可用的，1）、5）、6）是用户不可用的。

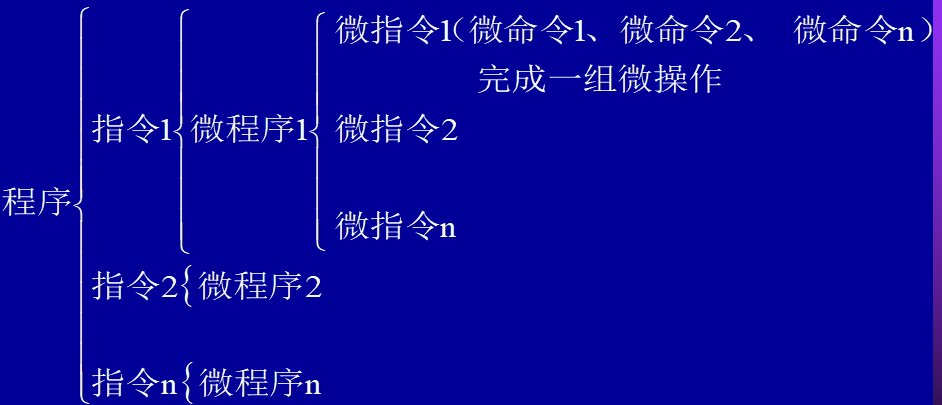
**（4）通用寄存器：**一组程序可访问的、具有多种功能的寄存器。在指令系统中，为通用寄存器分配了编号（寄存器地址），可以编程指定使用其中的某个寄存器。

1. **数据通路及指令流程分析**

**根据指令功能和CPU的数据通路结构写出指令流程、控制信号序列及一个指令周期的的访存次数**

1. **组合逻辑控制器的组成方式**
2. **微程序控制器**

**微程序控制基本概念：微命令、微操作、微指令、微程序、微周期、控制存储器**



**（1）微命令：**直接作用于部件或控制门电路的控制命令。是构成控制信号序列的最小单位。（比如模型机中的PC→BUS1，R0→BUS1、CPIR、R/W等控制信号都称为微命令。）

**（2）微操作：**由微命令控制实现的最基本的操作。微操作的定义可大可小，可由一个或多个微命令构成（比如微操作 (PC)→MAR；微操作打开PC与BUS1之间的控制门）

**（3）微指令：**用以产生一组微命令，控制完成一组微操作的二进制编码字称为微指令。（通常控制实现数据通路中的一步操作过程）

**（4）微程序：**一系列微指令的有序集合称为微程序。（一条机器指令对应一段微程序）

**（5）微周期：**从控制存储器中读取一条**微指令**并执行相应的微操作所需的时间称为微周期。

**（6）控制存储器（CM）：**存放微程序的存储器。简称控存，也称为微程序存储器。

**微程序控制器的组成方式**

**（1）控制存储器CM：**控存的每个单元存放一条微指令代码。

**（2）微指令寄存器µIR：**存放从控存中读取的微指令。微指令通常分为两大字段:**操作控制字段**和**地址控制字段**

**（3）微地址形成电路：**用于产生起始微地址和后继微地址，保证微程序的连续执行。

**（4）微地址寄存器µMAR：**接受微地址形成电路送来的地址，为读取微指令准备好控存的地址。

**（5）译码与驱动电路：**对µMAR中的微地址进行译码，找到被访问的控存单元并驱动其进行读取操作，读取微指令并存放于微指令寄存器中。

**微指令的编译方式（微指令格式的设计方法）：直接控制法、最短编码法、字段直接编码法**

**（1）直接控制法（不译法）：**微指令操作控制字段的每一位都直接表示一个微命令，该位为“1”，表示执行这个微命令，为“0”表示不执行该微命令。

**（2）最短编码法：**将所有的微命令进行统一编码，每条微指令只定义一个微命令。（L≥log2N）

**（3）字段直接编码法：**将微指令操作控制字段划分为若干个子字段，每个子字段的所有微命令进行统一编码。（互斥的放在同一字段，相容的放在不同字段，要注意为每个字段留一个状态表示无命令）

**微程序的顺序控制方式：初始微地址的形成方式；后继微地址的形成方式：增量方式、断定方式**

**（1）初始微地址的形成方式：**经过“取指令”后，机器指令从主存取到IR，然后根据机器指令操作码转换出该指令所对应的微程序入口地址，形成初始微地址。

1）一级功能转移

2）二级功能转移：先按指令类型标志转移，以区分出是哪一类指令。

3）用PLA电路实现功能转移

**（2）后继微地址的形成方式：**每条微指令执行完毕，都要根据要求形成后继微地址，以保证微程序的正常执行。

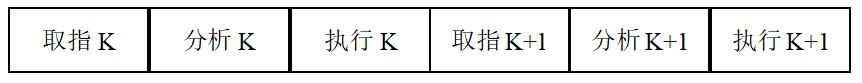
**1）增量方式：**当微程序按地址递增顺序一条条地执行微指令时，后继微地址是现行微地址加上一个增量（通常为1）

**2）增量方式下的地址转移：① 由微地址形成电路产生转移微地址：**由微地址形成电路根据执行的微指令和当前状态形成转移微地址。**② 由微指令的下地址控制字段设置转移微地址：**通常把微指令的地址控制字段分为两个部分：**转移地址字段BAF：**提供转移地址；**转移控制字段BCF：**规定地址的形成方式。

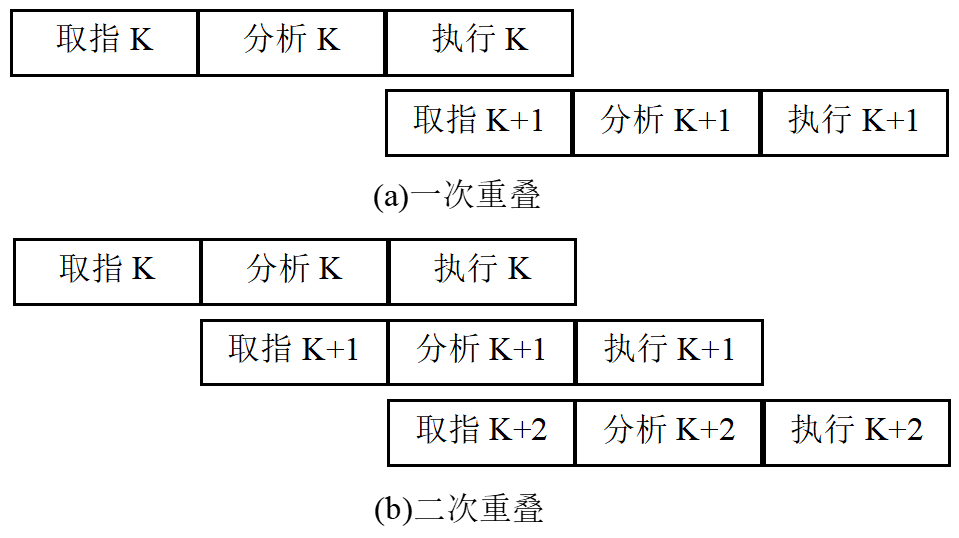
**3）断定方式：**后继微地址可由设计者指定或由设计者指定的测试判定字段控制产生。

1. **指令的执行方式：顺序方式、重叠方式、流水方式**

**（1）顺序方式：**各条指令之间顺序串行执行。即一条指令执行完以后，才取下条指令来执行。



**（2）重叠方式：**前一条指令的解释执行完成之前，就开始下一条指令的解释执行，即相邻两条指令在时间上相互重叠。



**（3）流水方式：**把指令的执行过程划分成若干个复杂程度相当、处理时间大致相等的子过程，每个子过程由一个独立的功能部件来完成。同一时间，多个功能部件同时工作，完成对不同子过程的处理。

1. **流水线的分类：操作部件级、指令级和处理机级；单功能流水线和多功能流水线；静态流水线和动态流水线；线性流水线和非线性流水线**

**（1）按处理级别分：**

**1）操作部件级流水线：**将复杂的运算过程组成流水线的工作方式

**2）指令级流水线：**将指令的整个执行过程分成若干个子过程

**3）处理机级流水线：**是一条宏流水线。多个处理机通过共享存储器串接起来处理同一数据流，每个处理机完成专一任务，并将结果输出到共享存储器。

**（2）按功能分类：**

**1）单功能流水线：**只完成一种功能

**2）多功能流水线：**可完成两种以上功能，其流水线控制较为复杂

**（3）按工作方式分类：**

**1）静态流水线：**在同一时间内只能以一种工作方式工作

**2）动态流水线：**允许在同一时间内将不同的功能段组合成具有多种功能的流水子集

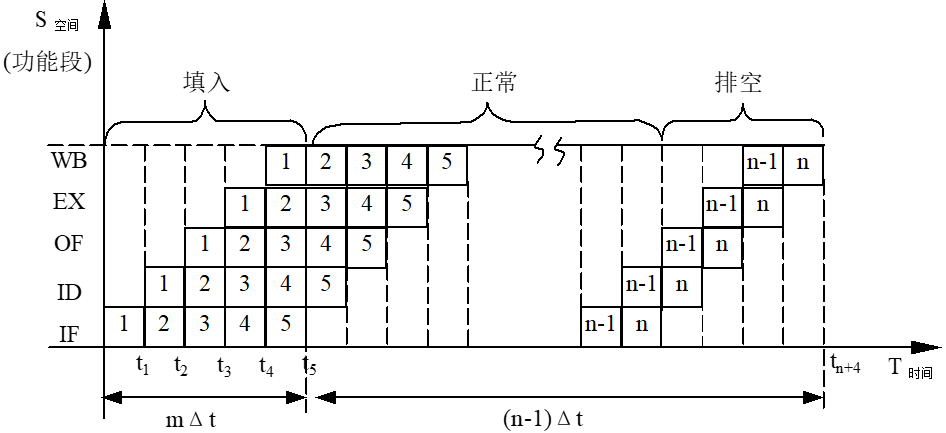
**（4）按流水线结构分：**

**1）线性流水线：**每个功能段在处理流水任务时，最多只经过一次，没有反馈回路。

**2）非线性流水线：**允许流水线的功能段可以通过反馈回路多次被使用

1. **线性流水线的性能：流水线时空图，线性流水线的吞吐率、效率和加速比的计算。**

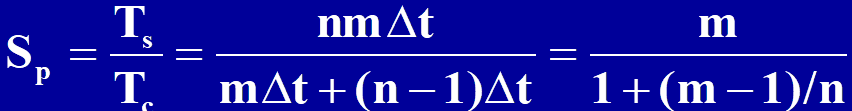
**（1）流水线时空图：**



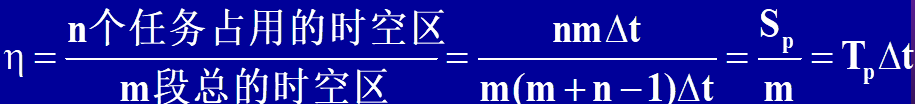
**（2）吞吐率：**单位时间内流水线能完成的指令数、任务数和输出结果的数量。



**（3）加速比：**采用流水线后的工作速度与等效的顺序串行方式的工作速度之比。



**（4）效率：**流水线中各功能段的利用率。



1. **线性流水线的三种相关及处理。**

**（1）资源相关（结构冒险）：**当有多条指令进入流水线后，在同一机器周期内争用同一功能部件而导致流水不能继续运行的现象。

**（2）解决资源相关方法：**

1）增加缓冲部件（如Cache，且分别设I-Cache、D-Cache），将指令提前预取到缓冲区。

2）针对存储冲突，可设两个不同的存储器（指令存储器、数据存储器）。RISC-V系统采用此方法。

3）使后续的相关指令延迟一节拍进入流水线。（插入气泡） 流水线阻塞一个步骤，称为流水线阻塞，经常被称为气泡。

**（3）数据相关（数据冒险）：**由于多条指令进入流水线后，各条指令的操作重叠进行，使得原来对操作数的访问顺序发生了变化，产生了错误的运行结果，而导致的数据冲突。

**（4）解决数据相关方法：**

1）增加快速直接通道（零延迟量法、定向技术-设置专用旁路）。

2）使后续相关的指令延迟进入流水线（推后法）。

**（5）控制转移相关（控制冒险）：**控制相关是指有分支指令、转子指令和中断等引起的相关冲突。

**（6）解决控制转移相关方法：**

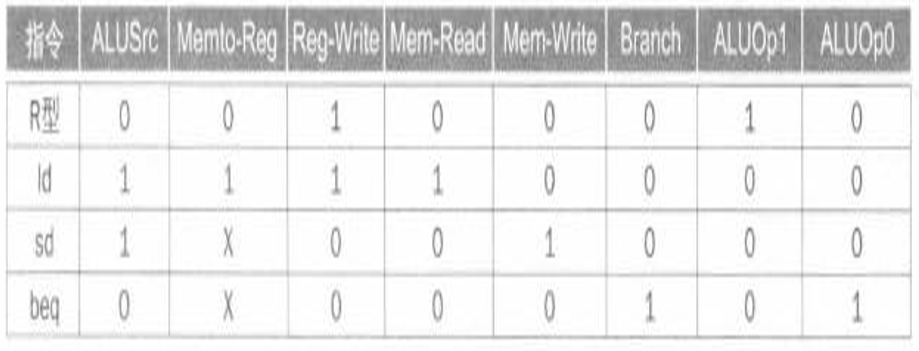
1）加快和提前形成条件码 （缩短分支延迟）

2) 预取转移成功（或不成功）分支序列的指令（假定分支不发生）

3) 采用延迟转移技术

4）动态转移预测（动态分支预测）（一种具有较高猜准率的动态方法是：考虑以前两次转移的历史。）

1. **RISC-V系统的处理器：数据通路及多选器的作用，常用RISC指令流程分析（指令流程、微控制信号序列等），ALU指令、Load/Store指控制相关是指有分支指令、转子指令和中断等引起的相关冲突。**
2. **令、分支指令特点及五个阶段的操作，流水线处理等**



1. **RISC-V各类指令的执行周期、使用频率等对性能提升的影响。**

RISC-V为了实现流水处理技术，将指令的执行分成五个阶段（5个周期），不同类型指令在不同阶段完成相应的工作





**重点：**

控制器组成及组成方式；CPU中各寄存器的作用；指令流程分析；程序功能分析；微指令的编译方式；微地址的形成方式；指令的执行方式；流水线的分类；线性流水线的性能计算、流水线的三种相关问题及分析；RISC-V指令的执行过程、性能评价。

**出题形式：**

填空、选择、判断、问答、设计★（指令流程及程序功能分析与实现；微指令格式设计；微程序的实现；线性流水线的性能分析）

#### 第七章

* 1. **总线的分类**

**（1）按总线所承担的任务：**

**内部总线：**用于实现主机系统内部各功能模块（部件）之间的互联。

**外部总线：**用于实现主机系统与外部设备或其它主机系统之间的互联。

**（2）按总线所处的物理位置：**

**片内总线：**实现芯片内部功能部件之间的连接。

**功能模块（板）内总线：**实现电路板上各个集成电路芯片之间的互连，

**功能模块（板）间总线：**用于把各个功能模块（如CPU、主存储器、I/O接口适配器等等）连接到一起，构成主机系统，所以也称为系统总线 。

**外部总线**

**（3）按总线所传送的信息类型：**

**地址总线**

**数据总线**

**控制总线**

**（4）按总线一次传送数据的位数：**

**串行总线**

**并行总线**

**（5）按总线操作的定时方式：**

**同步总线**

**异步总线**

* 1. **总线的性能：带宽、宽度、时钟频率、负载能力**

**（1）总线带宽：**单位时间内总线所能传输的最大数据量，也称总线传输率，用兆字节/秒（MB/s）来表示。

对于同步总线有：**总线带宽＝总线时钟频率×(总线宽度/8)**（总线宽度指数据总线的宽度）

**（2）总线宽度：**一个总线所设置的通信线路（或线缆）的数目称为该总线的宽度。

**1）数据总线宽度：**数据总线的宽度决定了一个总线内设置的用于传送数据的信号线的数目。

**2）地址总线宽度：**地址总线的宽度决定了计算机系统的寻址能力。

**（3）总线时钟频率：**对于同步总线来说，由于采用统一的时钟脉冲作为定时基准，因此总线的时钟频率越高，总线上的操作就越快。在数据总线宽度相同的情况下，较高的总线时钟频率，会带来较大的数据吞吐量。

**（4）总线负载能力：**总线上可以连接模块的最大数目。

* 1. **总线上的设备分类：总线主设备和总线从设备；总线源设备和总线目的设备；**

**（1）按逻辑功能分：**

**1）总线主设备：**总线操作的发起者，负责全面的总线控制。

**2）总线从设备：**总线操作的对象 。

**（2）按在信息交换中的地位分：**

**1）总线源设备：**发送数据的设备。

**2）总线目的设备：**接受数据的设备。

（**注意：**源设备不一定就是主设备，目的设备也不一定就是从设备。如,CPU访存取数。）

* 1. **总线仲裁的方法：**

**集中仲裁和分布仲裁；**

**（1）集中仲裁：**在系统中设置一个仲裁电路集中处理连接到总线上的各个设备所提出的使用总线的请求信号，集中对它们的优先级进行比较，由此确定总线的主控设备。

**（2）分布仲裁：**系统中每一个总线设备中都有较为复杂的总线访问请求控制逻辑，优先级比较电路也是分布在各个总线设备中，由各个已连接到总线上的并且目前有总线请求的设备共同来决定下面应该由哪个设备成为总线的主控设备。

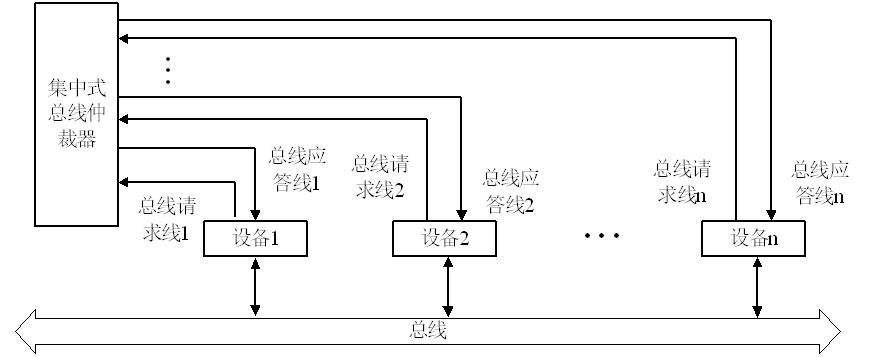
**并行仲裁和串行仲裁；**

**（1）并行仲裁：**连接到总线上的每个设备与总线仲裁电路之间都有独立的总线请求线和总线允许信号线。

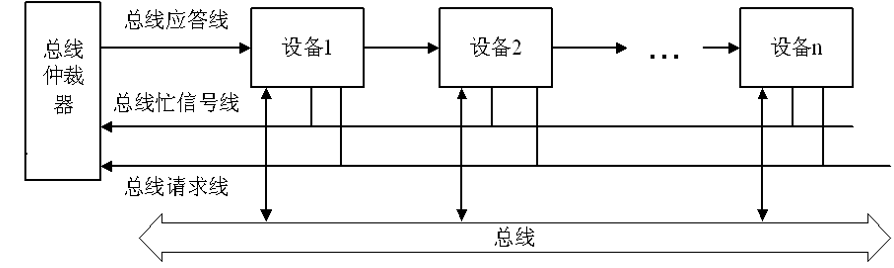
**（2）串行仲裁：**连接到总线上的设备共用一条总线请求信号线或（和）一条总线允许信号线。

**集中式总线控制器的仲裁方式**

**（1）集中式并行总线控制器：**系统中设置集中总线仲裁器，连接到总线上的每个设备分别有一条总线请求信号线和一条总线允许信号线连接到总线仲裁器上。

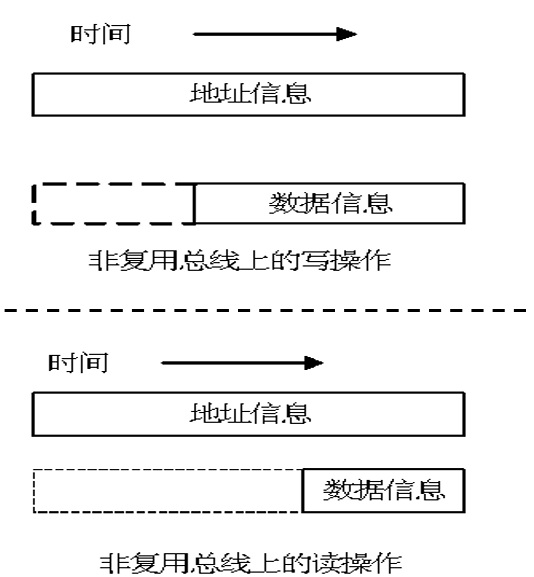


**（2） 集中式串行总线控制器：**系统中设置集中总线仲裁器，连接到总线上的设备共用一条总线请求信号线或（和）一条总线允许信号线。

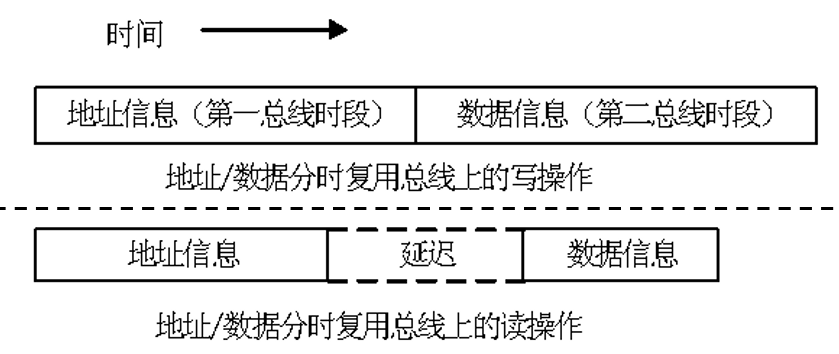


* 1. **总线的数据传输类型**

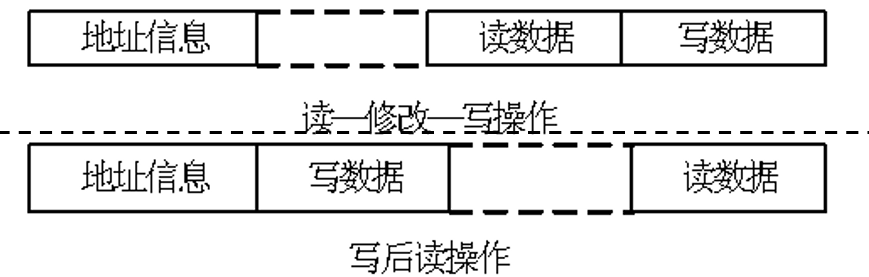
**（1）非复用总线上的读写操作：**



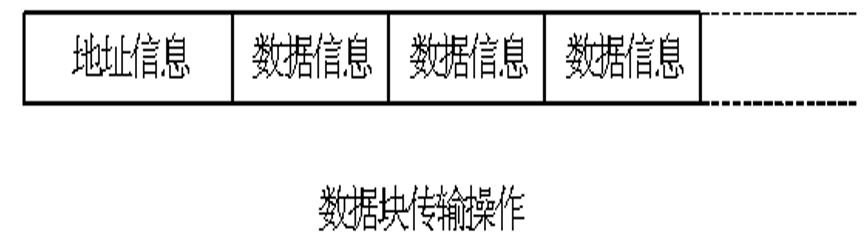
**（2）分时复用总线上的读写操作：**



**（3）联合操作：**



**（4）数据块传输方式：**



在给出了第一个数据所在存储单元的地址之后，可以有多个数据传送操作，即可以读/写连续的多个数据单元。也称为**连续数据传输方式、突发（猝发或迸发）数据传输方式、成组数据传输方式**。

**重点：**

总线上的设备分类；总线仲裁的方法；总线带宽的计算。

**出题形式：**

填空、选择、判断、问答、计算。

#### 第九章

1. **主机与外设的连接方式**

**（1）总线型连接方式：**CPU通过系统总线与主存储器，I/O控制器，I/O接口电路相连接，通过I/O接口电路进一步实现对外设的控制。

**（2）通道控制连接方式：**由通道控制器执行专门的通道程序，通过I/O总线控制接口与外设。（**四级连接方式：**主机→ 通道→ I/O接口(设备控制器)→ 外设）

**（3）I/O处理机控制连接方式(IOP方式)：**I/O处理机(IOP)是一种具有比I/O通道更强独立性的专用CPU。IOP有自己的指令系统，可编程控制，适应性强、通用性好。其程序的执行可与CPU并行，可使CPU彻底摆脱对I/O的控制处理。

1. **接口的功能、组成、分类**

**（1）接口的功能：**

1）实现数据的传送、缓冲、隔离和锁存。

2）实现信号形式和数据格式转换。

3）控制主机与外设之间的通信联络，实现控制命令和状态信息的交换，保证时序协调。

4）寻址，即识别设备地址，选择指定的设备和I/O端口。

**（2）接口的组成：**

**1）设备选择电路：**用于接收总线传来的地址信息，经译码后，决定选择哪个设备或I/O接口内部的部件。

**2）数据缓冲寄存器（数据端口）：**用于存放主机与外设之间要传递的数据信息。

**3）命令寄存器（控制端口）：**用于存放主机向外设发送的控制命令。

**4）状态寄存器（状态端口）：**用于存放外设或接口的工作状态。

**5）其它有关部件：**如中断控制逻辑、DMA控制逻辑以及各类特殊部件。

**（3）接口的分类：**

**1）按数据传送格式分：**

**串行接口：**接口与设备之间的信息传送是逐位串行进行的。

**并行接口：**接口与设备之间的信息传送是将一个字或一个字节的所有位同时并行地进行传送的。

**2）按时序的控制方式分：**

**同步接口：**一般与同步总线相连，接口与总线采用统一时钟信号，无论CPU与I/O设备，还是存储器与I/O设备交换信息，都与总线同步时钟脉冲同步。

**异步接口：**与异步总线相连，接口与系统总线之间采用异步应答方式。

**3）按信息传送交换的控制方式分：**

**程序控制I/O接口：**采用直接程序控制方式进行信息交换的接口。

**程序中断I/O接口：**主机与外设以程序中断方式进行信息交换控制的接口。

**直接存储器存取I/O接口：**以直接存储器存取(DMA)方式控制信息传送的接口。

1. **I/O的寻址方式**

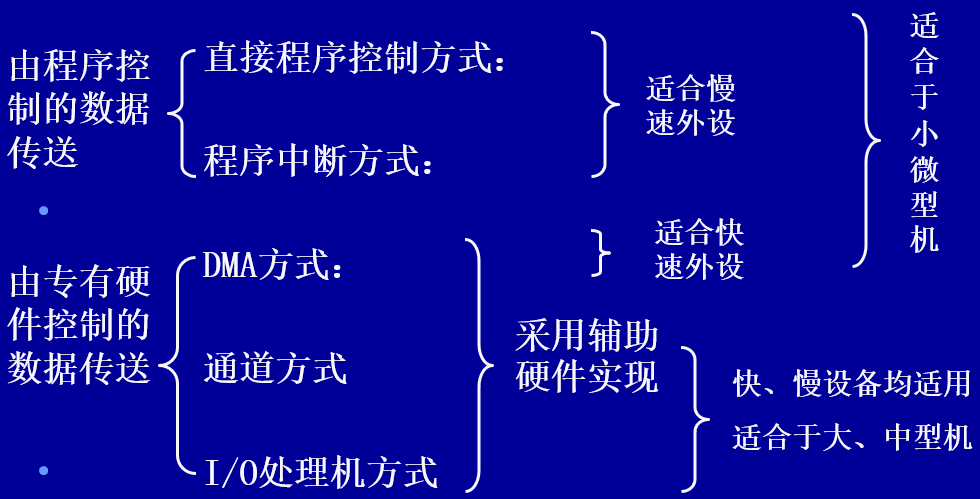
**（1）I/O端口与主存统一编址（存储器映像编址）：**

将一个I/O端口作为存储器中的一个单元对待，每一个I/O端口占用一个存储器单元地址。编址时将I/O端口与存储器单元一起进行编址。

**（2）I/O端口独立编址：**

将I/O端口与存储器单元分别独立进行编址，CPU访问外设时，需使用专门的I/O指令，并需要有与接口电路联系的单独的控制信号。

1. **I/O信息的传送方式**



直接程序控制方式分为：直接数据传送方式、轮询方式

1. **中断的功能和工作过程**

**中断请求、中断响应的条件，中断屏蔽、中断禁止、中断判优的条件，中断响应过程，向量中断的实现过程，中断嵌套的原则及处理过程**

**（1）中断处理过程：**中断请求 -> 中断判优及响应 ->保护现场 -> 中断服务 -> 恢复现场 -> 中断返回

**（2）中断响应条件：**

1）有中断源请求中断。

2）CPU允许响应中断，即处于开中断状态。

3）现行指令不是停机指令

4）一条指令执行结束(响应外部中断的条件)。

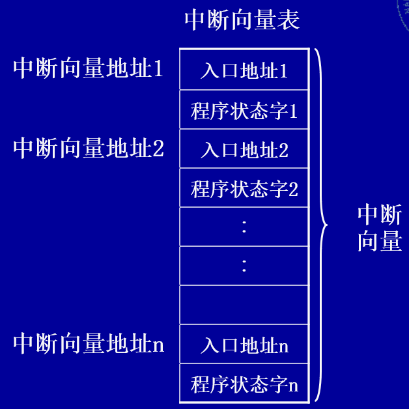
**（3）中断响应过程：**

1）关中断

2）保存断点地址和程序状态字

3）转入中断服务程序入口，以便执行相应的中断服务程序，完成中断处理任务。（直接由硬件完成，称为中断隐指令）

**（4）向量中断：**

**1）中断向量：**每个中断源对应的中断服务程序的入口地址及其程序状态字PSW。

**2）中断向量表：**存放中断向量的表

**3）中断向量地址：**访问中断向量表的地址码，即读取中断向量所需的地址，也称为中断指针。

**4）向量中断原理：**将各个中断服务程序的入口地址(或包括状态字)组织成中断向量表；响应中断时，由硬件直接产生对应于中断源的向量地址；据此访问中断向量表，从中读取中断服务程序入口地址，由此转向中断服务程序。（中断周期直接由硬件完成）

**（5）向量中断实现过程：**

1）中断源提出中断请求。

2）若CPU允许中断，则发出中断回答信号。

3）优先级编码电路( 如,中断控制器8259A)形成优先级最高的中断请求的中断向量地址存入中断向量地址寄存器VAR。

4）保护断点和现场，PC、PSW入栈。

5）根据中断向量地址，将对应的中断服务程序入口地址和PSW送入PC和PSR。

6）转入中断服务程序，进行中断服务。

7）中断返回，将保存的PC和PSW弹回PC和PSR。

1. **中断对流水线的影响及处理**

**（1）不精确断点法：**不论在第i条指令中的哪个流水功能段上发生中断请求，都不再允许此时未流入流水线的后继指令进入流水线。但已在流水线中的所有指令，允许它们**继续流动到执行完毕**，然后再去执行中断处理程序。

**（2）精确断点法：**不论在第i条指令的哪个流水功能段上发生的中断请求，流水线均停止流动，用大量后援寄存器**保存所有现场**。即中断处理程序对现场的处理是精确对应第i条指令的。在第i条之后已进入流水线的所有指令的原有现场，在中断处理完毕后都能恢复。

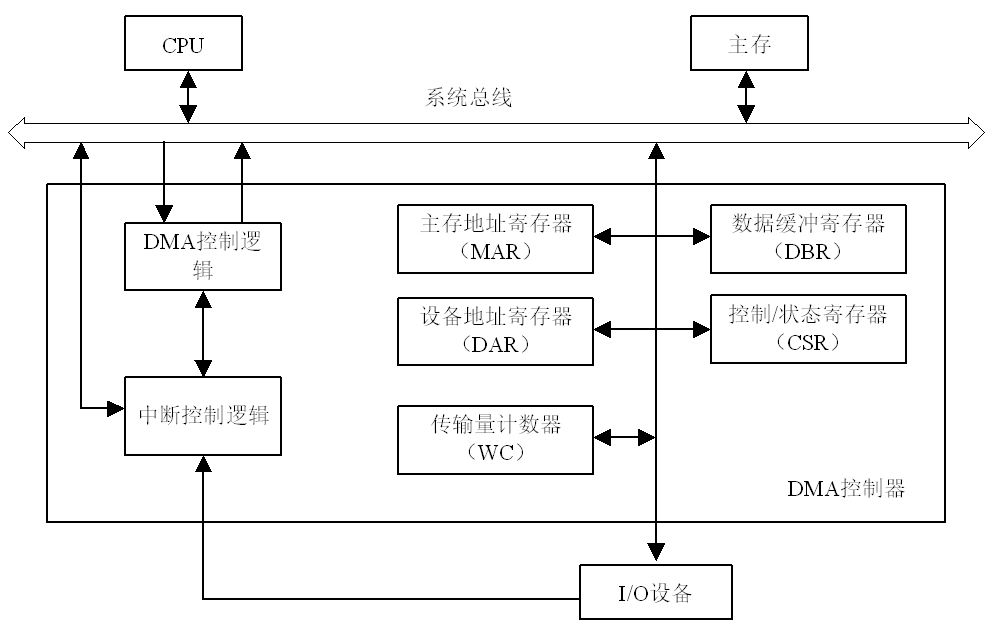
1. RISC-V系统的中断流水处理方法

采用精确断点法

1. **DMA的功能和工作过程**

**（1）DMA方式：**以主存为中心，采用硬件手段在主存与I/O设备之间建立直接的数据传送通路，由DMA控制器（DMAC）取得总线控制权，控制主存与I/O设备之间的数据传送，在传送过程中不需要CPU的程序干预的数据传送控制方式。

**DMAC的组成**



**（1）主存地址寄存器MAR：**用于存放主存中需要交换数据的地址。由CPU通过**执行程序**将数据在主存中的首地址送到主存地址寄存器。可以自增或自减，指向下一单元。

**（2）传输量计数器 WC：**用于记录传送数据的总字数。以负数补码表示，每传送一个字节或字增量1，为0时表示传送完成

**（3）数据缓冲寄存器DBR：**用于暂存每次传送的数据。

**（4）设备地址寄存器DAR：**存放I/O设备的设备码或表示设备信息存储区的寻址信息。（编码形式取决于外设）

**（5）控制/状态寄存器 CSR：**存放有关控制和状态信息，如传送方式、读/写状态、传送完毕与否等。

**（6）DMA控制逻辑：**DMA控制逻辑负责完成DMA的预处理（初始化各类寄存器）、接收设备控制器送来的DMA请求信号、向设备控制器回答DMA允许（应答）信号、向系统申请总线以及控制总线实现DMA传输控制等工作。

**（7）DMA中断控制逻辑：**DMA中断控制逻辑负责在DMA操作完成后向CPU发出中断请求，申请CPU对DMA操作进行后处理或进行下一次DMA传送的预处理。

**DMA传送方式：CPU暂停方式、周期挪用方式、交替访存方式**

**（1）CPU停机方式：**用CPU停机方式实现DMA传送时，CPU停止工作，让出对总线的控制权，而由DMAC接管总线，进行数据传送。数据传送结束后，再将总线交还给CPU。

**（2）周期挪用(周期窃取)方式：**当I/O设备无DMA传送请求时，CPU正常访问主存。当I/O设备需要使用总线传送数据时，产生DMA请求，DMAC把总线请求发给CPU。

1）若CPU本身无使用总线的要求，CPU就把总线交给DMAC，由DMAC控制I/O设备使用总线

2）如果此时CPU也要使用总线，则CPU自身进入一个或几个“空闲总线周期”状态，即CPU让出一个或几个总线周期给DMAC（也称DMAC“挪用”一个总线周期），DMAC利用此总线周期控制传送一个数据字后，再把总线交还给CPU，以便CPU可以继续执行总线操作。

（**注：**CPU让出总线必须在存储周期结束之后；外设访存优先于CPU访存）

**（3）交替访存方式：**将一个CPU周期分为两个分周期，与DMA分别使用。其中一个专供DMA访存，另一个专供CPU访存。

**DMA的接口类型**

**DMA控制方式下的数据传送过程：DMA预处理、数据交换操作、DMA后处理**

**（1）预处理：**在DMAC开始工作之前，CPU通过**执行输入输出程序**完成下面工作：

1） 控制寄存器中写DMA操作命令。给DMA控制逻辑指明数据传送方向是输入(主存写)还是输出(主存读)。

2）向DMA设备地址寄存器DAR送入设备号，并启动设备。

3）向DMA主存地址寄存器MAR送入交换数据的主存起始地址。

4）向传输量计数器WC送入交换数据的个数。

**（2）数据交换操作：**DMAC获得总线后，即可按规定的传送方式，进行数据的输入或输出操作，直到将所有数据传输完毕，DMAC将总线交还给CPU。需要时还向CPU发出中断请求。

**（3）DMA后处理：**CPU**响应中断**后，为DMA传送作结束处理工作。

1）校验送入主存的数据是否正确

2）决定是否继续用DMA方式传送，还是结束传送

3）测试在传送过程中是否发生了错误

4）判断传送工作是否正常结束

**DMA与中断的比较**

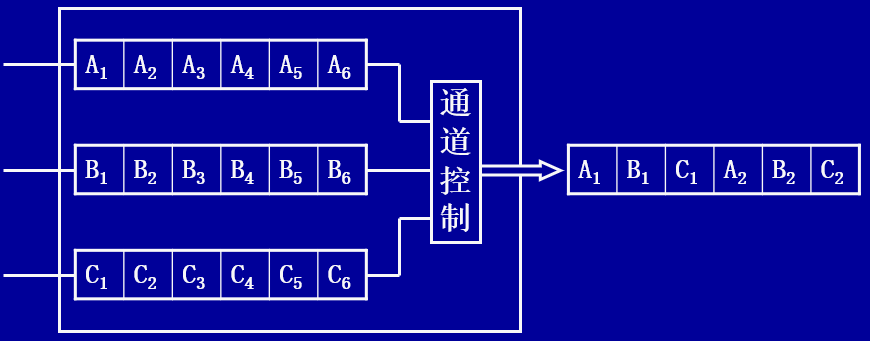
|  |  |
| --- | --- |
| **程序中断** | **DMA方式** |
| 以CPU为中心，采用软硬结合，以软件为主的方式，控制设备与主机之间的数据传送。 | 以主存为中心，采用硬件手段，控制设备与主存间直接进行数据传送。 |
| 因为需要程序切换，所以需要保护与恢复现场。 | 由DMA控制器直接控制数据传送。在数据传送期间，不需要CPU干预，不需保护与恢复现场。 |
| 适合于慢速外设。 | 适合于快速外设。 |
| 必须在一条指令执行结束后才能响应。 | 在一个访存周期结束后即可响应。 |
| 可实现多种处理功能 | 仅用于数据传送 |

1. **I/O通道控制方式的基本概念**

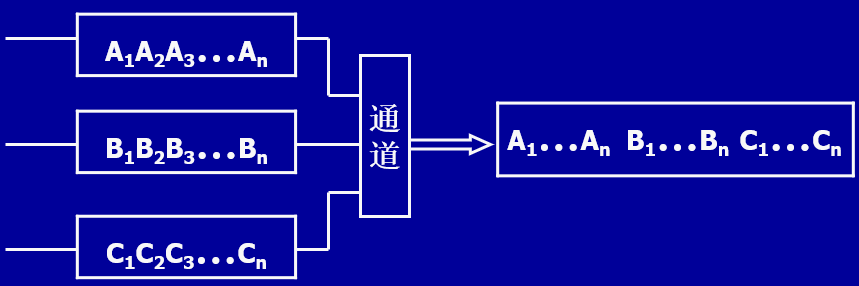
通道作为一个独立的I/O控制部件，能执行有限的I/O通道指令，代替CPU管理和控制外设。通道使主机与I/O设备之间能够达到更高的并行程度。

1. **通道的类型：字节多路通道、选择通道、数组多路通道**

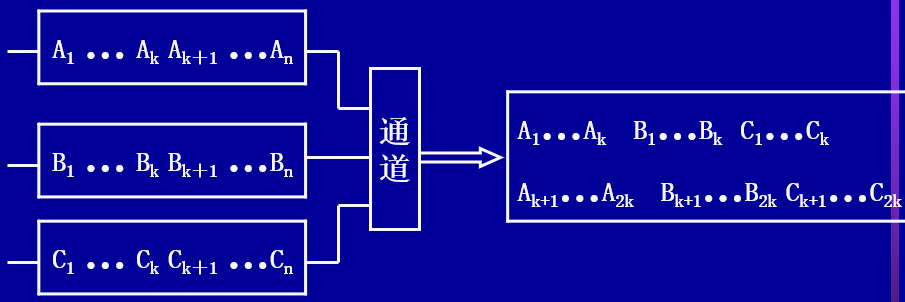
**（1）字节多路通道：**字节多路通道是一种简单的共享通道，可以依靠通道与CPU之间的高速数据通路分时地为多台设备服务。（字节交叉模式工作）



**（2）选择通道：**只有一套完整的硬件，以独占的方式工作，逐个轮流地为物理上连接的几台高速外设服务。（独占模式工作）



**（3）数组多路通道：**数组多路通道将字节多路通道和选择通道的特性结合起来。一个通道可带有多个子通道，各子通道以成组交叉模式轮流使用通道。（成组交叉模式工作）

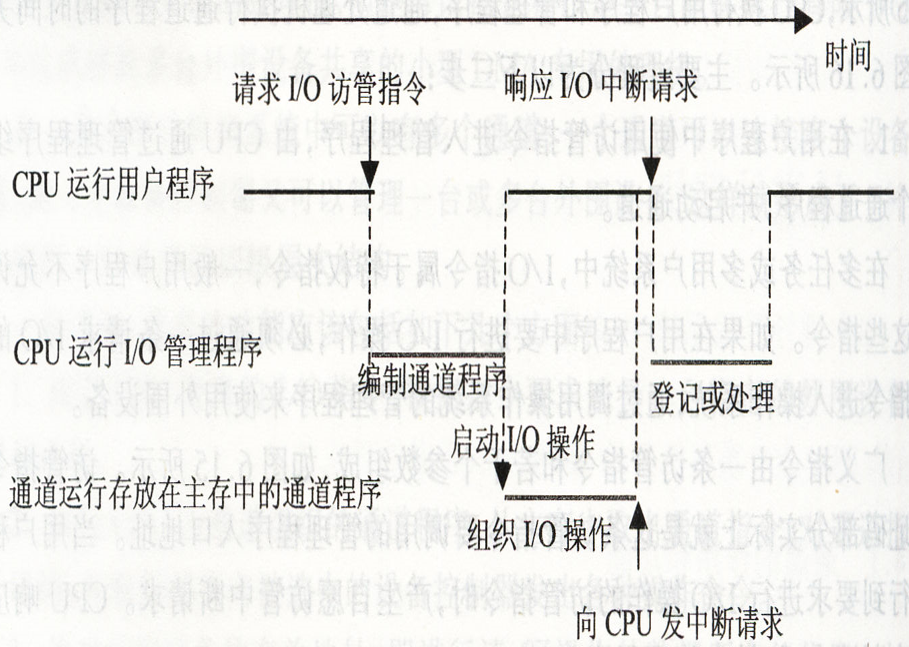


**（4）字节多路通道和数组多路通道的异同：**

**相同点：**都是多路通道，在一段时间内可以交替地执行多个设备的通道程序。

**不同点：**1）数组多路通道允许多个设备同时工作，但只允许一个设备进行传输型操作，其他设备只能进行控制型操作。字节多路通道不仅允许多个设备同时操作，而且允许它们同时进行传输型操作2）数组多路通道传输数据基本单位是数据块，而字节多路通道传输数据的基本单位是字节

1. **I/O通道的工作原理**



**重点：**

接口的功能；中断的功能和工作过程；向量中断的实现过程；DMA的功能和工作过程。中断、DMA方式下数据传输的特点及性能计算；通道的类型。多种I/O设备中断或DMA处理的优先响应及时间计算

**出题形式：**

填空、选择、判断、问答、计算、设计★（中断及DMA的功能和实现）

主要参考资料

1. 《计算机组成原理》（修订版，2016）---教材

2. 《计算机组成与设计 硬件/软件接口》（原书RISC-V版） ---pdf文件

（第1章、第2章、第4章）

#### 附加

1. Cache使用的是SRAM；CPU使用的是DRAM

2. 因为(-0.5)10＝(－0.1)2，写成补码为1.1，不符合规格化规定，所以(-0.5)10不是规格化数；(－1)10的补码是(1.000)2，符合规格化规定，所以－1是规格化数。

3.定点补码加减时，只有同号数相加才有可能溢出

4. (－1)×(－1)是定点小数补码乘法中唯一的溢出情况

5. **超流水技术：**流水线进一步细分；

**超长指令字技术：**一条指令实现多个操作；

**超标量技术：**多条流水线

6.当外设向CPU发出DMA请求时，CPU将在当前总线周期结束时给予响应

7.总线概念：总线作为计算机传送信息的通道，是连接各个功能部件的纽带。总线由多条通信线路（或线缆）组成。使用总线实现部件互连的优点：

（1）可减少各个部件之间的连线数量，降低成本。

（2）便于系统构建、系统性能扩充和产品更新换代。

8.中断概念：中断是指处理机暂时中止执行现行程序而转去执行处理更加紧迫事件的服务程序，待处理完毕后，再自动返回执行原来的程序的过程。

9.常见的集中式总线控制有三种：链式查询、计数器定时查询、独立请求；

其中：链式查询方式连接简单，易于扩充，对电路故障敏感；

计数器定时查询方式优先级设置较灵活，对故障不敏感，连线及控制过程较复杂；

独立请求方式速度最快，但是硬件器件用量大，连线多，成本较高。