

## 《计算机组成原理》(白中英) 复习知识点总结

### 第一章 计算机系统概论

#### 电子数字计算机的分类(P1)

通用计算机(超级计算机、大型机、服务器、工作站、微型机和单片机)和专用计算机。

#### 计算机的性能指标 (P5)

#### 数字计算机的五大部件及各自主要功能 (P6)

五大部件：存储器、运算器、控制器、输入设备、输出设备。

存储器主要功能:保存原始数据和解题步骤。

运算器主要功能：进行算术、逻辑运算。

控制器主要功能：从内存中取出解题步骤（程序）分析，执行操作。

输入设备主要功能: 把人们所熟悉的某种信息形式变换为机器内部所能接收和识别的二进制信息形式。

输出设备主要功能: 把计算机处理的结果变换为人或其他机器所能接收和识别的信息形式。

#### 计算机软件 (P11)

系统程序——用来管理整个计算机系统

应用程序——按任务需要编制成的各种程序

### 第二章 运算方法和运算器

#### 课件+作业

### 第三章 内部存储器

#### 存储器的分类 (P65)

按存储介质分类:

易失性: 半导体存储器

非易失性: 磁表面存储器、磁芯存储器、光盘存储器

按存取方式分类:

存取时间与物理地址无关(随机访问):

随机存储器 RAM——在程序的执行过程中可读可写

只读存储器 ROM——在程序的执行过程中只读

存取时间与物理地址有关(串行访问):

顺序存取存储器 磁带

直接存取存储器 磁盘

按在计算机中的作用分类:

主存储器: 随机存储器 RAM——静态 RAM、动态 RAM

只读存储器 ROM——MROM、PROM、EPROM、EEPROM

Flash Memory

高速缓冲存储器(Cache)

辅助存储器——磁盘、磁带、光盘

#### 存储器的分级 (P66)

存储器三个主要特性的关系: 速度、容量、价格/位

多级存储器体系结构: 高速缓冲存储器(cache)、主存储器、外存储器。

#### 主存储器的技术指标 (P67)

存储容量: 存储单元个数  $M \times$  每单元位数  $N$

存取时间: 从启动读(写)操作到操作完成的时间

存取周期: 两次独立的存储器操作所需间隔的最小时间, 时间单位为 ns.

存储器带宽:单位时间里存储器所存取的信息量, 位/秒、字节/每秒, 是衡量数据传输速率的重要技术指标。

#### **SRAM 存储器 (P67)**

基本存储元:用一个锁存器(触发器) 作为存储元.

基本的静态存储元阵列 (P68)

双译码方式 (P68)

读周期、写周期、存取周期(P70)

#### **DRAM 存储器 (P70)**

基本存储元: 由一个 MOS 晶体管和电容器组成的记忆电路。

存储原理: 所存储的信息 1 或 0 由电容器上的电荷量来体现(充满电荷: 1; 没有电荷: 0)。

一个 DRAM 存储元的写、读、刷新操作 (P71)

DRAM 的刷新:集中式刷新和分散式刷新 (P73)

#### **存储器容量的扩充 (P73)**

位扩展——增加存储字长 (P73)

字扩展——增加存储字的数量 (P73)

字、位扩展 (P74)

例题 (P73)

#### **只读存储器 ROM (P80)**

掩模 ROM、PROM、EPROM、EEPROM、Flash 存储器 (P80-86)

#### **并行存储器 (P86)**

双端口存储器: 指同一个存储器具有两组相互独立的读写控制线路。

多模块交叉存储器:连续地址分布在相邻的不同模块内, 同一个模块内的地址都是不连续的。对连续字的成块传送可实现多模块流水式并行存取,大大提高存储器的带宽。

#### **cache 基本原理 (P92)**

避免 CPU “空等”现象

CPU 和主存 (DRAM) 的速度差异

程序访问的局部性原理

cache 由高速的 SRAM 组成

cache 的基本原理 (P93)

命中、未命中、命中率 (P93)

例题 (P94)

#### **cache 与主存的地址映射(P94)**

全相联映像: 主存中的任一块可以映象到缓存中的任一块。

直接映像: 每个缓存块可以和若干个主存块对应;每个主存块只能和一个缓存块对应。

组相联映像:某一主存块  $j$  按模  $u$  映射到 缓存 的第  $i$  组中的 任一块。

#### **替换算法 (P98)**

先进先出算法 (FIFO): 把一组中最先调入 cache 的块替换出去, 不需要随时记录各个块的使用情况, 所以实现容易,开销小。

近期最少使用算法(LRU): 将近期内长久未被访问过的行(块)换出。每行设置一个计数器, cache 每命中一次, 命中行计数器清零, 其它各行计数器增 1。当需要替换时, 比较各特定行的计数值, 将计数值最大的行换出。

最不经常使用 (LFU):被访问的行计数器增加 1, 换值小的行, 不能反映近期 cache 的访问情况。

随机替换：从特定的行位置中随机地选取一行换出。

#### cache 的写操作策略 (P99)

写回法、全写法、写一次法 (P99-100)

### 第四章 指令系统

#### 指令系统(P103)

程序、高级语言、机器语言、指令、指令系统、复杂指令系统计算机 (CISC)、精简指令系统计算机(RISC) (P103)

#### 指令格式 (P105)

操作码：指令操作性质的二进制数代码

地址码：指令中的地址码用来指出该指令的源操作数地址(一个或两个)、结果地址及下一条指令的地址。

三地址指令、二地址指令、一地址指令、零地址指令；三种二地址指令(SS、RR、RS) (P106)

指令字长度、机器字长(P107)

例题 (P110)

#### 操作数类型 (P110)

地址数据、数值数据、字符数据、逻辑数据

#### 寻址方式 (P112)

确定本条指令的操作数地址，下一条欲执行指令的指令地址

指令寻址

顺序寻址——PC+1

跳跃寻址——转移类指令

数据寻址(P112-116)

立即寻址——形式地址就是操作数

直接寻址——有效地址由形式地址直接给出

隐含寻址——操作数地址隐含在操作码中

间接寻址——有效地址由形式地址间接提供

寄存器寻址——有效地址即为寄存器编号

寄存器间接寻址——有效地址在寄存器中

基址寻址——有效地址=形式地址+基址地址

变址寻址——有效地址=形式地址+变址寄存器的内容

相对寻址——有效地址=PC 的内容+形式地址

堆栈寻址——栈顶指针

段寻址

例题 (P118)

#### 指令的分类(119)

数据处理、数据存储、数据传送、程序控制

#### RISC 技术 (P121)

RISC——精简指令系统计算机

CISC——复杂指令系统计算机

RISC 指令系统的特点(P121)

### 第五章 中央处理器

#### CPU 的功能 (P127)

指令控制、操作控制、时间控制、数据加工



## **CPU 的基本组成(P127)**

控制器、运算器、cache

## **CPU 中的主要寄存器(P128)**

数据缓冲寄存器(DR)、指令寄存器 (IR)、程序计数器 (PC)、数据地址寄存器 (AR)、通用寄存器、状态字寄存器 (PSW)

## **操作控制器的分类(P130)**

时序逻辑型：硬布线控制器

存储逻辑型：微程序控制器

## **指令周期 (P131)**

取出并执行一条指令所需的全部时间。

指令周期、机器周期、时钟周期(P131)

一个指令周期含若干个机器周期

一个机器周期包含若干个时钟周期

取指周期 (数据流) (P132)

执行周期 (数据流) (P133—138)

## **时序信号的作用和体制 (P141)**

时序信号的基本体制是电位—脉冲制。数据加在触发器的电位输入端 D，打入数据的控制信号加在触发器的时钟脉冲输入端 CP。电位高低表示数据是 1 还是 0,要求打入数据的控制信号来之前电位信号必须已稳定。

节拍电位、节拍脉冲 (P142)

## **控制器的控制方式 (P144)**

同步控制方式:即固定时序控制方式,各项操作都由统一的时序信号控制, 在每个机器周期中产生统一数目的节拍电位和工作脉冲。

异步控制方式:不受统一的时钟周期(节拍)的约束;各操作之间的衔接与各部件之间的信息交换采取应答方式。

联合控制方式: 同步控制和异步控制相结合的方式,大部分指令在固定的周期内完成,少数难以确定的操作采用异步方式。

## **微程序控制原理 (P145)**

微程序控制是指运行一个微程序来实现一条机器指令的功能。微程序控制的基本思想:仿照计算机的解题程序,把微操作控制信号编制成通常所说的“微指令”,再把这些微指令按时序先后排列成微程序,将其存放在一个只读存储器里,当计算机执行指令时,一条条地读出这些微指令,从而产生相应的操作控制信号,控制相应的部件执行规定的操作。

微程序、微指令、微命令、微操作(P145)

机器指令与微指令的关系 (P150)

## **微命令的编码方法 (P151)**

直接表示法:微指令的每一位代表一个微命令,不需要译码。

编码表示法:把一组相斥性的微命令信号组成一个小组(即一个字段),然后通过小组(字段)译码器对每一个微命令信号进行译码,译码输出作为操作控制信号。

混合表示法: 把直接表示法与字段编码表示法混合使用,以便能综合考虑微指令字长、灵活性、速度等方面的要求。

## **微指令格式(P153)**

水平型微指令: 是指一次能定义并能并行执行多个微命令的微指令。

垂直型微指令: 微指令中设置微操作码字段, 采用微操作码编译法, 由微操作码规定微指令的功能, 称为垂直型微指令。垂直型微指令的结构类似于机器指令的结构。

### 硬连线控制器(P155)

基本思想：通过逻辑电路直接连线而产生的,又称为组合逻辑控制方式。这种逻辑电路是一种由门电路和触发器构成的复杂树形逻辑网络。

三个输入：来自指令操作码译码器的输出；来自执行部件的反馈信息；来自时序产生器的时序信号，包括节拍电位信号 M 和节拍脉冲信号 T。

一个输出:微操作控制信号

硬布线控制器的基本原理：某一微操作控制信号 C 用一个逻辑函数来表达。

### 并行处理技术(P161)

并行性的概念：问题中具有可以同时进行运算或操作的特性。

时间并行:让多个处理过程在时间上相互错开，轮流使用同一套硬件设备的各个部件，以加快硬件周转而赢得速度，实现方式就是采用流水处理部件。

空间并行：以数量取胜.它能真正的体现同时性

时间+空间并行:综合应用.Pentium 中采用了超标量流水线技术。

### 流水线的分类 (P163)

指令流水线:指指令步骤的并行。将指令流的处理过程划分为取指令、译码、取操作数、执行、写回等几个并行处理的过程段。

算术流水线：指运算操作步骤的并行。如流水加法器、流水乘法器、流水除法等。

处理机流水线：是指程序步骤的并行。由一串级联的处理机构成流水线的各个过程段，每台处理机负责某一特定的任务。

### 流水线中的主要问题 (P164)

资源相关：指多条指令进入流水线后在同一机器时钟周期内争用一个功能部件所发生的冲突。

数据相关：在一个程序中，如果必须等前一条指令执行完毕后，才能执行后一条指令。解决数据相关冲突的办法：为了解决数据相关冲突，流水 CPU 的运算器中特意设置若干运算结果缓冲寄存器，暂时保留运算结果，以便于后继指令直接使用，称为“向前”或定向传送技术。

控制相关：由转移指令引起的。解决控制相关冲突的办法：延迟转移法、转移预测法。

例题(P165)

## 第六章 总线系统

### 总线的概念 (P184)

总线是构成计算机系统的互联机构，是多个系统功能部件之间进行数据传送的公共通路。

### 总线的分类 (P184)

内部总线——CPU 内部连接各寄存器及运算部件之间的总线。

系统总线——CPU 和计算机系统中其他高速功能部件相互连接的总线。按系统传输信息的不同，又可分为三类：数据总线，地址总线和控制总线。

I/O 总线——中、低速 I/O 设备之间互相连接的总线。

### 总线性能指标(P185)

总线宽度:指数据总线的根数。

寻址能力：取决于地址总线的根数。PCI 总线的地址总线为 32 位，寻址能力达 4GB。

传输率：也称为总线带宽，是衡量总线性能的重要指标。

例题 (P193)

### 总线上信息传送方式 (P190)

串行传送：使用一条传输线,采用脉冲传送(有脉冲为 1，无脉冲为 0)。连续几个无脉冲



的处理方法：位时间。

并行传送：每一数据位需要一条传输线，一般采用电位传送（电位高为 1，电位低为 0）。

分时传送：总线复用、共享总线的部件分时使用总线。

### 总线接口 (P192)

I/O 接口，也叫适配器，和 CPU 数据的交换一定是并行的方式，和外设数据的交换可以是并行的，也可以是串行的。

### 总线的仲裁 (P193)

集中式仲裁：有统一的总线仲裁器。

链式查询方式、计数器定时查询方式、独立请求方式 (P193—195)

分布式仲裁：不需要中央仲裁器，每个潜在的主方功能模块都有自己的仲裁器和仲裁号。(P195)

### 总线的定时 (P196)

同步定时：事件出现在总线上的时刻由总线时钟信号来确定。

异步定时：后一事件出现在总线上的时刻取决于前一事件的出现，即建立在应答式或互锁机制基础上。

### PCI 总线 (P200)

PCI：外围设备互连，PCI 总线：连接各种高速的 PCI 设备。PCI 是一个与处理器无关的高速外围总线，又是至关重要的层间总线。它采用同步时序协议和集中式仲裁策略，并具有自动配置能力。PCI 总线支持无限的猝发式传送。即插即用。

## 第七章 外围设备

### 外围设备的定义和分类(P209)

除了 CPU 和主存外，计算机系统的每一部分都可作为一个外围设备来看待。外围设备可分为输入设备、输出设备、外存设备、数据通信设备和过程控制设备几大类。

### 磁记录原理 (P210)

计算机的外存储器又称磁表面存储设备。所谓磁表面存储，是用某些磁性材料薄薄地涂在金属铝或塑料表面作载磁体来存储信息。磁盘存储器、磁带存储器均属于磁表面存储器。

磁性材料上呈现剩磁状态的地方形成了一个磁化元或存储元，是记录一个二进制信息位的最小单位。

### 磁表面存储器的读写原理(P211)

在磁表面存储器中，利用一种称为磁头的装置来形成和判别磁层中的不同磁化状态。通过电-磁变换，利用磁头写线圈中的脉冲电流，可把一位二进制代码转换成载磁体存储元的不同剩磁状态；通过磁-电变换，利用磁头读出线圈，可将由存储元的不同剩磁状态表示的二进制代码转换成电信号输出。

### 磁盘的组成和分类(P213)

硬磁盘是指记录介质为硬质圆形盘片的磁表面存储设备。它主要由磁记录介质、磁盘控制器、磁盘驱动器三大部分组成。

温彻斯特磁盘简称温盘，是一种采用先进技术研制的可移动磁头固定盘片的磁盘机。它是一种密封组合式的硬磁盘，即磁头、盘片、电机等驱动部件乃至读写电路等 组装成一个不可随意拆卸的整体。

### 磁盘上信息的分布 (P215)

记录面、磁道、扇区 (P215)

磁道编号 (P215)

磁盘地址由记录面号(也称磁头号)、磁道号和扇区号三部分组成。

### 磁盘存储器的技术指标(P216)

存储密度：存储密度分道密度、位密度和面密度。

道密度：沿磁盘半径方向单位长度上的磁道数，单位道/英寸。

位密度：磁道单位长度上能记录的二进制代码位数，单位为位/英寸。

面密度：位密度和道密度的乘积，单位为位/平方英寸。

平均存储时间=寻道时间+等待时间+数据传送时间 (P216)

数据传输率 (P217)

例题 (P217)

### 磁盘 cache (P218)

磁盘 cache 是为了弥补慢速磁盘和主存之间速度上的差异。

### 磁盘阵列 RAID (P218)

RAID：独立磁盘冗余阵列（廉价冗余磁盘阵列），或简称磁盘阵列。简单的说，RAID 是一种把多块独立的硬盘(物理硬盘)按不同方式组合起来形成一个硬盘组（逻辑硬盘），从而提供比单个硬盘更高的存储性能和提供数据冗余的技术。

组成磁盘阵列的不同方式成为 RAID 级别。RAID 0 提高存储性能的原理是把连续的数据分散到多个磁盘上存取，这样，系统有数据请求就可以被多个磁盘并行的执行，每个磁盘执行属于它自己的那部分数据请求。这种数据上的并行操作可以充分利用总线的带宽，显著提高磁盘整体存取性能。

## 第八章 输入输出系统

### 外围设备的速度分级 (P236)

在 CPU 和外设之间数据传送时加以定时：

速度极慢或简单的外设：CPU 只需要接受或者发送数据即可。

慢速或者中速的设备：可以采用异步定时的方式。

高速外设：采用同步定时方式。

### I/O 和主机信息交换方式(P237)

程序查询方式、程序中断方式、直接内存访问(DMA) 方式、通道方式

#### 程序查询方式 (P239)

数据在 CPU 和外围设备之间的传送完全靠计算机程序控制。当需要输入/输出时，CPU 暂停执行主程序，转去执行设备输入/输出的服务程序，根据服务程序中的 I/O 指令进行数据传送。

这是一种最简单、最经济的输入/输出方式，只需要很少的硬件。但由于外围设备动作很慢，程序进入查询循环时将浪费 CPU 时间。

#### 中断的概念(P242)

中断是指 CPU 暂时中止现行程序，转去处理随机发生的紧急事件，处理完后自动返回原程序的功能和技术。

#### 程序中断方式的原理(P242)

在程序中断方式中，某一外设的数据准备就绪后，它“主动”向 CPU 发出请求中断的信号，请求 CPU 暂时中断目前正在执行的程序而进行数据交换。当 CPU 响应这个中断时，便暂停运行主程序，并自动转移到该设备的中断服务程序。当中断服务程序结束以后，CPU 又回到原来的主程序。

#### 中断处理过程中的几个问题(P243)

CPU 只有在当前一条指令执行完毕后，即转入公操作时才受理设备的中断请求。

保存现场(P243)

中断屏蔽 (P243)

中断处理过程 (P243)



### 单级中断和多级中断 (P245)

单级中断系统中，所有的中断源都属于同一级，所有中断源触发器排成一行，其优先次序是离 CPU 近的优先权高。当响应某一中断请求时，执行该中断源的中断服务程序。在此过程中，不允许其他中断源再打断中断服务程序，即使优先权比它高的中断源也不能再打断。

多级中断系统是指计算机系统中有相当多的中断源，根据各中断事件的轻重缓急程度不同而分成若干级别，每一中断级分配给一个优先权。优先权高的中断级可以打断优先权低的中断服务程序，以程序嵌套方式工作。

一维多级中断是指每一级中断里只有一个中断源，

二维多级中断是指每一级中断里又有多个中断源。

### DMA 的基本概念 (P253)

直接内存访问(DMA) 是一种完全由硬件执行 I/O 交换的工作方式。在这种方式中，DMA 控制器从 CPU 完全接管对总线的控制，数据交换不经过 CPU，而直接在内存和 I/O 设备之间进行。DMA 方式一般用于高速传送成组数据。

DMA 方式的优点 (P253)

### DMA 能执行的一些操作 (P254)

从外围设备发出 DMA 请求；CPU 响应请求，把 CPU 工作改成 DMA 操作方式，DMA 控制器从 CPU 接管总线的控制；由 DMA 控制器对内存寻址，即决定数据传送的内存单元地址及数据传送个数的计数，并执行数据传送的操作；发中断，向 CPU 报告 DMA 操作的结束。

### DMA 传送方式 (P254)

停止 CPU 访问内存、周期挪用、DMA 与 CPU 交替访内 (P254)

### DMA 数据传送过程(P257)

传送前预处理；正式传送；传送后处理。(P257)

### 通道的基本概念(P261)

通道是一个特殊功能的处理器，它有自己的指令和程序专门负责数据输入输出的传输控制，而 CPU 将“传输控制”的功能下放给通道后只负责“数据处理”功能。这样，通道与 CPU 分时使用内存，实现了 CPU 内部运算与 I/O 设备的平行工作。

### 通道的功能 (P253)

通道具有两种类型的总线：存储总线：承担通道与内存、CPU 与内存之间的数据传输任务。通道总线即 I/O 总线，承担外围设备与通道间的数据传送任务。

从逻辑结构上讲，I/O 系统一般具有四级连接：CPU 与内存→通道→设备控制器→外围设备

优先级别：由于大多数 I/O 设备的读写信号具有实时性，不及时处理会丢失数据；所以通道与 CPU 同时要求访内时，通道优先权高于 CPU。

### CPU 对通道的管理 (P262)

CPU 是通过执行 I/O 指令以及处理来自通道的中断，实现对通道的管理。

来自通道的中断有两种，一种是数据传送结束中断，另一种是故障中断。

### 通道对 I/O 模块的管理 (P262)

通道通过使用通道指令控制 I/O 模块进行数据传送操作，并以通道状态字接收 I/O 模块反映的外围设备的状态。

### 通道的类型 (P262)

选择通道、数组多路通道、字节多路通道(P263)

## 第九章 操作系统支持

### 虚拟存储器的概念 (P282)



虚拟存储器是借助于磁盘等辅助存储器来扩大主存容量,使之能为更大或更多的程序所使用。是一个容量非常大的存储器的逻辑模型,不是任何实际的物理存储器。它指的是主存-外存层次。以透明的方式给用户提供了一个比实际主存空间大得多的程序地址空间。

实地址:或物理地址,计算机物理内存的访问地址,由 CPU 引脚送出,是用于访问主存的地址,对应的存储空间——物理存储空间或主存空间。

虚地址:或逻辑地址,在编制程序时独立编址,使用的地址,对应的存储空间——虚存空间或逻辑地址空间。

虚地址到实地址的转换过程——程序的再定位。

### 虚存的访问过程 (P283)

虚拟存储器的用户程序以虚拟地址编址并存放在辅存中;程序运行时 CPU 以虚地址访问主存,由辅助硬件找出虚地址和物理地址的对应关系,判断这个虚地址指示的存储单元是否已装入主存:如果在主存,CPU 就直接执行已在主存的程序;如果不在,要进行辅存向主存的调度。

### 虚存与 cache 的异同 (P283)

#### 几种虚拟存储器 (P284)

段式、页式、段页式

#### 页式虚拟存储器 (P284)

页、页表:页式虚拟存储系统中,虚地址空间被分成等大小的页,称为逻辑页;主存空间也被分成同样大小的页,称为物理页。相应地,虚地址分为两个字段:高字段为逻辑页号,低字段为页内地址(偏移量);实存地址也分两个字段:高字段为物理页号,低字段为页内地址。通过页表可以把虚地址(逻辑地址)转换成物理地址。

页式虚存地址映射:地址变换时,用逻辑页号作为页表内的偏移地址索引页表,并找到相应物理页号,用物理页号作为实存地址的高字段,再与虚地址的页内偏移量拼接,就构成完整的物理地址。

虚页内容若没有调入主存,则计算机启动输入输出系统,把虚地址指示的一页内容从辅存调入主存,再提供 CPU 访问。

转换后援缓冲器 (P285)

#### 段式虚拟存储器 (P286)

段式虚拟存储器,是以程序的逻辑结构所形成的段(如主程序、子程序、过程、表格等)作为主存分配单位的虚拟存储器管理方式的存储器。

每个段的大小可以不相等。每个程序都有一个段表(映像表),用于存放该道程序各程序段从辅存装入主存的状况信息。段表一般驻留在主存中。

段式虚存地址映射 (P287)

#### 段页式虚拟存储器 (P287)

把程序按逻辑单位分段以后,再把每段分成固定大小的页。程序对主存的调入调出是按页面进行的,但它又可以按段实现共享和保护,兼备页式和段式的优点。

### 虚存的替换算法 (P289)

虚拟存储器中的替换策略一般采用 LRU (Least Recently Used) 算法、LFU 算法、FIFO 算法,或将两种算法结合起来使用。

例题 (P289)