

梦享考研系列

2016 年考研核心考点命题思路解密

计算机组成原理

梦享团队 组编

内容简介

《2016 年考研核心考点命题思路解密——计算机组成原理》严格按照最新计算机考研 408 统考大纲的计算机组成原理部分编写，涵盖大纲指定的所有考试内容。本书对统考大纲所涉及的知识点进行深入剖析和总结，并精心策划和部署每一个章节，对每一个章节的考点做了独家策划。

本书每一个考点中的命题，绝大部分来源于历年名校计算机考研真题和统考真题，少部分来源名校期末考试试题中的精华部分，是全国 408 统考大纲和高校考研真题的较好结合。为了提高考题的质量和解析的准确度，参考资料采用以考研权威教材、习题、考研真题为主，多方借鉴众多高校从事多年教育的教师课堂资料。梦享团队对每一个命题的思路和解题方法进行深入详细地讲解，并附上大量的图来帮助考生理解记忆，力求考生能够通过掌握一个题目而达到举一反三，有利于考生利用更少的时间掌握更多的知识。

本书可作为考生参加计算机专业研究生入学考试的备考复习用书，也可作为计算机专业的学生学习计算机组成原理的练习用书。

前言

梦享团队成立于 2013 年 10 月份, 目前共有 31 人, 队员以中科院、清华大学和北京交通大学 3 所高校的学生为主, 其他名校学生为辅, 都是上研不久的研究生, 以及一些考研论坛上参与答疑多年的版主等。在考研复习和辅导上, 梦享团队队员有着相对丰富的阅历。在考研的路上, 梦享团队队员也经历过和大家一样的坎坷辛苦。我们深切地体会到, 每一个考研的同学十分不容易。

计算机专业考研的命题, 侧重于考查同学们对基础知识的掌握, 考研书更应该侧重于培养同学们的实战能力。但目前的考研教材绝大多数倾向于知识点的讲解, 不注重培养考生的实战能力, 导致很多考生知识很丰富, 但是很难将这些知识很好地运用于解题。编写偏向于实战的参考书不同于知识讲解, 需要编者花费大量的时间来规划和布置章节、考点和解析考题。目前能找到的计算机考研命题解析类参考资料, 要么题目特别少但讲解特别详细啰嗦, 要么题目太多的而对命题的讲解十分粗略甚至只有一个最终答案。因而, 梦享团队决定写一套注重实战、解析详细、**直击重点**、严格参考大纲的参考书。

经过两年多的**努力**, “梦享考研系列”参考书终于一本一本和大家见面了, 到目前为止, 我们总共有 5 本图书已经出版。其中, 《计算机网络》完成于 2013 年 10 月, 《数据结构》完成于 2014 年 3 月, 《计算机操作系统》完成于 2014 年 9 月, 《计算机组成原理》完成于 2015 年 3 月。最后一本书, 《408 统考核心题型》, 完成于 2015 年 5 月。

为了提高图书的权威性, 本套图书严格按照 408 统考大纲编写, 涵盖了统考大纲所有指定的内容, 并融合了统考真题和历年名校考研真题的精华, 是全国 408 统考大纲、统考真题和高校考研真题的较好结合。为了提高考题的质量和解析的准确度, 参考资料采用以考研权威教材、习题、考研真题为主, 多方借鉴众多高校从事多年教育的教师课堂资料。

本书具有以下特色:

1. 组织严谨, 结构清晰

梦享考研系列图书通过对统考大纲和历年高校考研真题的深入剖析和总结, 精心规划和部署了各个章节, 对每一个章节的考点作了独家策划, 使得本套图书组织严谨, 结构清晰, 便于考生对各章考点逐个击破。

2. 突出重点, 注重实战

对于每一个计算机专业的考研同学而言, 复习任务是相当繁重的。除了四门统考专业

课之外，还有数学、英语和政治。所以，突出重点，让同学们把极其有限的时间都花在刀刃上，是我们的首要工作，而提高同学们的实战能力，是我们系列图书的最终目的。

为了直击命题、突出重点、给同学们节约时间、培养同学们的实战能力，在考题的挑选上，我们通过对统考和自主命题的高校常出现的考题类型和知识点进行深入的总结，抛开在统考或者自主命题的高校的考研真题上极少出现的极难、极易、极偏知识点，精心挑选了和考研难度相近的考题供大家练习。

此外，我们还根据考点的重要程度来完成考点内容分布，在较重要的考点部署较多的内容，在较重要的内容部署较多的命题，在较为不重要的知识点抓住重点布置核心题型。目的也在于突出考试难度、突出考试重点，方便大家进行实战训练，提高学习效率，让考生在更短的时间内掌握更多的知识点。

3. 解析详细，深入剖析

“梦享考研”系列图书一共 5 本，每一本都很厚，可能会吓怕很多同学。是不是题目太多了？不是的，其实考题并不多，我们并不提倡题海战术，也不提倡对于同一个知识点反复命题和赘述，我们提倡“少而精”。针对每一个考点可能出现的命题类型，我们精心挑选了极具代表性的命题供大家实战训练，并对这些习题进行详细、深入的剖析，揭露问题的本质和解题的精髓，有助于大家掌握解题方法和技巧，提高大家的实战能力，在较短的时间掌握更多的知识。

《2016 年计算机考研核心考点命题思路解密》系列图书是梦享团队 2 年多的汗水结晶，融入了梦享团队的集体智慧。另外，真诚感谢我们团队新成员张丽方、胡明明、刘春、白洋等 10 几位同学提供的建议和帮助！

在接下来的更长时间里，梦享团队将用最诚挚的心和最大的努力，写出更优秀、更有助于大家利用更短的时间考得更高分数的考研书。我们每年都会合理调整这套图书，使得这套图书更加受到大家青睐。

梦享团队会牢牢记住这样一句话——“助你们实现研究生梦想，是我们的梦想！”跟这套图书一样，我们愿伴随着 2016 年考研的同学一起度过艰辛的追梦季节，伴随着大家一起度过每一个难忘的日日夜夜！也祝福 2016 年考研的你们，获得圆满的成功！

安楠

2016 年 4 月于北京

目 录

第一章 计算机系统概述	7
考点 1 计算机发展历程.....	7
考点 2 计算机系统层次结构.....	8
考点 3 计算机性能指标.....	11
第二章 数据的表示和运算	15
考点 1 数制与编码	15
考点 2 定点数表示和运算.....	22
考点 3 浮点数的表示和运算.....	28
考点 4 算数逻辑单元 ALU	34
第三章 存储系统	37
考点 1 存储器的分类	37
考点 2 存储器的层次化结构.....	40
考点 3 半导体随机存取存储器.....	43
考点 4 主存储器与 CPU 的连接.....	47
考点 5 双口 RAM 和多模块存储器.....	53
考点 6 高速缓冲存储器 (Cache)	57
考点 7 虚拟存储器	62
第四章 指令系统	67
考点 1 指令格式	67
考点 2 指令的寻址方式.....	72

考点 3 CISC 和 RISC 的基本概念	79
第五章 中央处理器 (CPU)	82
考点 1 CPU 的功能和基本结构	82
考点 2 指令执行过程	84
考点 3 数据通路的功能和基本结构	86
考点 4 控制器的功能和工作原理	92
考点 5 指令流水线	98
第六章 总线	103
考点 1 总线概述	103
考点 2 总线仲裁	109
考点 3 总线操作和定时	114
考点 4 总线标准	116
第七章 输入输出 (I/O) 系统	118
考点 1 I/O 系统基本概念	118
考点 2 外部设备	118
考点 3 I/O 接口 (I/O 控制器)	122
考点 4 I/O 方式	124

第一章 计算机系统概述

考点 1 计算机发展历程

温馨提示：计算机的发展历程，在历年的考研中出现较少，请同学们注意计算机发展的几个历程。稍作了解即可。

一. 选择题部分

1. (原书第5题) 计算机问世至今，新型机器不断推陈出新，不管怎样更新，依然保有“存储程序”的概念，最早提出这种概念的是()。
- A. 巴贝奇 B. 冯·诺依曼
- C. 帕斯卡 D. 贝尔

【解析】 本题考查“冯·诺依曼”计算机体系结构的设计思想。

冯·诺依曼计算机的设计思想可以简要地概括为以下三点:

- (1). 计算机应包括运算器、存储器、控制器、输入和输出设备五大基本部件。
- (2). 计算机内部应采用二进制来表示指令和数据。每条指令由操作码和地址码两部分组成。其中操作码表示运算性质，地址码指出操作数在存储器中的地址。
- (3). 采用存储程序方式。将编写好的程序送入内存存储器中，然后启动计算机工作，计算机无需操作人员干预，能自动逐条取出指令并执行指令。

从以上 3 条可以看出, 以前所有的讨论都是针对冯·诺依曼设计思想论述的, 不过没有明确指出其人罢了。冯·诺依曼设计思想最重要之处在于明确地提出了“程序存储”的概念, 他的全部设计思想实际上是对“程序存储”概念的具体化。

参考答案: B

2. (原书第8题)1949年研制成功的世界上第一台存储程序式的计算机称为()。

A. EDVAC B. EDSAC

C. ENIAC D. UNIVAC- I

参考答案: B

2. (原书第8题)1949年研制成功的世界上第一台存储程序式的计算机称为()。
- A. EDVAC B. EDSAC
- C. ENIAC D. UNIVAC- I

【解析】 本题考查第一台存储程序的计算机名称。

EDVAC 是世界上第一台具有存储功能的计算机，于 1950 年在冯 诺依曼领导下研制成功。该计算机采用“存储程序和程序控制”原理，至今的计算机仍然遵从这个原理。

参考答案：A

3. **（原书 第 9 题）** 至今为止，计算机中所有信息仍以二进制方式表示，其原因是（ ）。
- A. 节约元件
 - B. 运算速度快
 - C. 物理器件性能决定
 - D. 信息处理方便

【解析】本题考查计算机以 2 进制存储程序和数据的原因。

至今为止，计算机中所有信息仍以二进制方式表示是由物理器件性能决定的。

参考答案：C

考点 2 计算机系统层次结构

温馨提示：本考点考查计算机系统的层次结构，把握计算机硬件的基本组成，计算机软件的分类和计算机的工作过程三个部分，都是基本的概念，请同学们熟记。

一. 选择题部分

1. **（原书 第 2 题）** 完整的计算机系统应包括（ ）。
- A. 运算器、存储器、控制器
 - B. 外部设备和主机
 - C. 主机和实用程序
 - D. 配套的硬件设备和软件设备

【解析】本题考查计算机系统的组成。

通常人们所说的一个完整的计算机系统应该包括硬件系统和软件系统。**硬件系统是指用电子器件和机电装置组成的物理实体，它包括组成微机的各部件和各种外部设备；软件系统是指计算机运行所需要的全部程序、数据和相关文档的总称。**硬件系统和软件系统共同决定了计算机的工作能力。通俗地说，硬件是计算机的躯体，软件是计算机的头脑和灵魂，两者缺一不可。

参考答案：D

2. (原书 第4题) 冯·诺依曼型计算机的设计思想是 ()。

- A. 存储数据并按地址顺序执行
- B. 存储程序并按地址逆序执行
- C. 存储程序并按地址顺序执行
- D. 存储程序并乱序执行

【解析】本题考查冯·诺依曼计算机的设计思想。

冯诺依曼计算机设计思想是存储程序并按地址顺序执行。指令由操作码和地址码两部分组成，操作码规定了操作的类型，地址码规定了操作的数据所存放的地址（可以是操作对象和操作结果的地址）。

【经典总结】

一般情况下，指令是顺序执行的，下一条指令的地址由程序计数器 PC 给出。当遇到转移指令或子程序调用等程序转移时，下一条要执行的指令的地址应该由当前指令指明。

参考答案：C

3. (原书 第7题) 描述汇编语言特性的概念中，有错误的句子是 ()。

- A. 对程序员的训练要求来说，需要硬件知识
- B. 汇编语言对机器的依赖性高
- C. 用汇编语言编制程序的难度比高级语言小
- D. 汇编语言编写的程序执行速度比高级语言快

【解析】本题考查汇编语言的特点。

学过汇编的同学都知道，汇编语言对编程的人来说，需要硬件知识。汇编语言对机器的依赖性很高。高级语言编写的程序（源程序）处于最高层，必须翻译成汇编语言，再由汇编程序汇编成机器语言（目标程序）之后才能被执行。所以，汇编语言的执行速度通常比高级语言快。

参考答案：C

4. (原书 第13题) 下列关于冯·诺依曼型计算机的描述，不正确的是 ()。

- A. 计算机硬件系统由运算器、存储器、控制器、输入设备、输出设备五大部件组成
- B. 指令和数据在存储器中都是二进制码存储
- C. 指令存储器和数据存储器独立分设在不同的存储器
- D. 存储程序并按地址顺序执行是 CPU 自动工作的关键

【解析】冯·诺依曼机的指令和数据都以二进制形式存放在存储器中，显然 C 答案错误。

参考答案：C

二. 综合应用题部分

1. (原书 第 2 题) 说明现代计算机系统的层次结构。

【解析】本题考查计算机系统层次结构。

计算机系统的层次结构如表 1.1 所示。

表 1.1 计算机系统的层次结构

5 级	高级语言级	编译程序
4 级	汇编语言级	汇编程序
3 级	操作系统级	操作系统
2 级	一般机器级	微程序
1 级	微程序设计级	直接由硬件执行

2. (原书 第 5 题) 简单描述计算机的层次结构中，各层次的主要特点。

【解析】现代计算机系统是一个硬件与软件组成的综合体，可以把它看成是按功能划分的多级层次结构。

第 0 级为**硬件组成的实体**。

第 1 级是**微程序级**。这级的机器语言是微指令集，程序员用微指令编写的微程序一般是直接由硬件执行的。

第 2 级是**传统机器级**。这级的机器语言是该机的指令集，程序员用机器指令编写的程序可以由微程序进行解释。

第 3 级是**操作系统级**。从操作系统的基本功能来看，一方面它要直接管理传统机器中的软硬件资源，另一方面它又是传统机器的延伸。

第 4 级是**汇编语言级**。这级的机器语言是汇编语言，完成汇编语言翻译的程序叫做汇编程序。

第 5 级是**高级语言级**。这级的机器语言就是各种高级语言，通常用编译程序来完成高级语言翻译工作。

第 6 级是**应用语言级**。这一级是为了使计算机满足某种用途而专门设计的，因此这一级语言就是各种面向问题的应用语言。

3. (原书 第 8 题) 图 1.1 是计算机硬件组成框图, 说明图中各部件的功能。

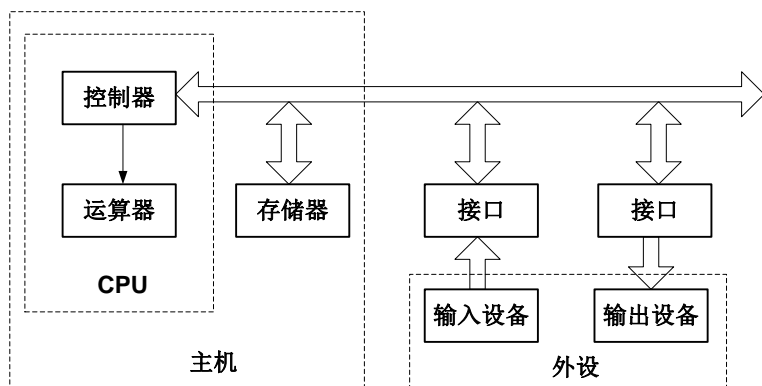


图 1.1 计算机硬件组成框图

【解析】本题考查计算机各部件的作用。计算机各部件的作用分别如下：

- (1). 控制器：整个计算机的指挥中心，它使计算机的各个部件自动协调工作。
- (2). 运算器：对数据信息进行处理部件，用来进行算术运算和逻辑运算。
- (3). 存储器：存放程序和数据，是计算机实现“存储程序控制”的基础。
- (4). 输入设备：将人们熟悉的信息形式转换成计算机可以接受并识别的信息形式的设备。
- (5). 输出设备：将计算机处理的结果（二进制信息）转换成人类或其它设备可以接收和识别的信息形式的设备。

考点 3 计算机性能指标

温馨提示：计算机的性能指标，包括吞吐量、响应时间、CPU 时钟周期、主频、CPI、CPU 执行时间、MIPS 和 MFLOPS。本考点涉及到简单的计算，同学们掌握计算机性能指标的计算方法。

一. 选择题部分

1. (原书 第 3 题) 若一台计算机的字长为 2 个字节，则表明该机器（ ）。
 - A. 能处理的数值最大为 2 位十进制数
 - B. 能处理的数值最多由 2 位二进制数组成

C. 400

D. 600

【解析】本题考查 CPI 的计算。

由题意，该计算机执行一条指令的平均时钟周期个数 CPI 计算如下：

$$CPI = 2 \times 50\% + 3 \times 20\% + 4 \times 10\% + 5 \times 20\% = 3$$

该机器的主频为 1.2GHz，即每一秒有 1.2G 个时钟周期，因为平均每执行一条指令需要 3 个时钟周期，所以平均每秒能执行 4 百万条指令，即 4MIPS。

参考答案： C

二. 综合应用题部分

1. (原书 第 1 题) 用一台 40MHz 处理机执行标准测试程序，它含的混合指令数和相应的所需时钟周期数如表 1.3 所示。

表 1.3 机器所含混合指令数及其所需相应的时钟周期数

指令类型	指令数	时钟周期数
整数运算	45000	1
数据传送	32000	2
浮点	15000	2
控制传送	8000	2

求有效 CPI（每条指令的平均时钟周期）、MIPS 速率和程序的执行时间。

【解析】指令的平均时钟周期为

$$CPI = \frac{\sum_{i=1}^n CPI_i \times I_i}{\sum_{i=1}^n I_i} = \frac{45000 \times 1 + 32000 \times 2 + 15000 \times 2 + 8000 \times 2}{45000 + 32000 + 15000 + 8000} = 1.55$$

$$MIPS = \frac{\text{时钟频率}}{CPI \times 10^6} = \frac{40MHz}{1.55 \times 10^6} = 25.8$$

$$\text{执行时间 } T_{execute} = (45000 + 32000 + 15000 + 8000) / (25.8 \times 10^6) = 0.0038s$$

2. (原书 第 5 题) 某 CPU 的主频为 8 MHz，若已知每个机器周期平均包含 4 个时钟周期，该机的平均指令执行速度为 0.8 MIPS。
- (1). 试求该机的平均指令周期及每个指令周期含几个机器周期？
 - (2). 若改用时钟周期为 0.4us 的 CPU 芯片，则计算机的平均指令执行速度为多少 MIPS？
 - (3). 若要得到平均每秒 40 万次的指令执行速度，则应采用主频为多少的 CPU 芯片？

【解析】

(1). 由主频为 8 MHz, 得时钟周期为 $1 / 8 = 0.125\mu\text{s}$, 机器周期为 $0.125 \times 4 = 0.5\mu\text{s}$ 。根据平均指令执行速度为 0.8 MIPS, 得平均指令周期为 $1 / 0.8 = 1.25\mu\text{s}$, 每个指令周期含 $1.25 / 0.5 = 2.5$ 个机器周期。

(2). 若改用时钟周期为 $0.4\mu\text{s}$ 的 CPU 芯片, 即主频为 $1 / 0.4 = 2.5 \text{ MHz}$, 则平均指令执行速度为: $(0.8 \text{ MIPS} \times 2.5 \text{ MHz}) / 8 \text{ MHz} = 0.25 \text{ MIPS}$ 。

若要得到平均每秒 40 万次的指令执行速度, 即 0.4 MIPS, 则 CPU 芯片的主频应为 $(8 \text{ MHz} \times 0.4 \text{ MIPS}) / 0.8 \text{ MIPS} = 4 \text{ MHz}$ 。

本章到此就结束了, 请问您有什么疑问吗? 任何问题, 欢迎您与我们作者进行交流!



梦享团队微信号



梦享团队官方微信公众号



梦享团队新浪微博

第二章 数据的表示和运算

考点 1 数制与编码

温馨提示：数制和编码部分，主要包括进位计数制及其相互转换、真值和机器数、BCD 码、字符和字符串以及校验码。其中，进制数相互转换、真值和机器数，都是自主命题高校常考的内容。校验码部分，请同学们注意奇偶校验码、循环冗余码和海明码，并学会利用这三种校验码解题。

一. 选择题部分

1. (原书第2题) 在下列机器数()中,零的表示形式是唯一的。
- A. 原码 B. 补码
- C. 反码 D. 原码和反码

【解析】 本题考查 0 在机器中的表示形式。

对于真值 0，原码和反码各有两种不同的表示形式，而补码只有唯一的一种表示形式。在补码表示中，真值 0 的表示形式是唯一的：

$$[+0]_{\text{补}} = [-0]_{\text{补}} = 00000$$

在反码表示中，真值 0 也有两种不同的表示形式：

$$[+0]_{\text{反}}=00000 \quad [-0]_{\text{反}}=11111$$

在原码表示中，真值 0 有两种不同的表示形式：

$[+0]_{\text{原}} = 00000$ $[-0]_{\text{原}} = 10000$

故而，本题选择 B 答案。

参考答案: B

2. (原书第6题) 计算机系统中采用补码运算的目的是为了()。
- A. 与手工运算方式保持一致 B. 提高运算速度
- C. 简化运算器的设计 D. 提高运算的精度

【解析】 本题考查计算机系统采用补码运算的目的。

计算机中的 CPU 仅有加法电路，没有减法电路。采用补码运算的目的之一，是将减法变成加法。同时，补码运算将符号位视为数共同参与运算，其结果仍然不会出错。所以，

计算机系统采用补码来运算。

参考答案：C

3. (原书 第 10 题) 关于数据表示和编码, 下列说法正确的是 ()。

- A. 奇偶校验码是一种功能很强的检错纠错码
- B. 在计算机中用无符号数来表示内存地址
- C. 原码、补码和移码的符号编码规则相同
- D. 用拼音从键盘上敲入汉字时, 使用的拼音码是汉字的字模码

【解析】我们一般认为, 奇偶校验码能检错, 但是不能纠错。比如, 产生了奇数位错误时, 奇偶校验码能检测出来, 但是发生了偶数位错误的时候, 奇偶校验码就检测不出来了。

例如, 数据 1010110 采用偶校验, 因为数据中 1 的个数为 4 个, 所以校验位为 0, 数据位和校验位是 10101100。若传输的过程中, 出现了两位错误, 即 1010 1100 变成 1001 1100, 则偶校验无法发现错误。

所以, A 答案错误。

原码、补码和移码的符号编码规则显然不同, 所以 C 答案错误。用拼音从键盘上敲入汉字时, 使用的编码是 ASCII 码。所以, D 答案错误。

参考答案：B

4. (原书 第 17 题) 假定下列字符码中有奇偶校验位, 但没有数据错误, 采用偶校验的字符码是 ()。

- A. 11001011
- B. 11010110
- C. 11000001
- D. 11001001

【解析】本题考查奇偶校验。

奇偶校验我们在计算机网络一书中的第二章也讲过, 其实道理很简单, 但是该怎么理解才容易记住呢?

【经典总结】

奇校验就是看数值位中 1 的个数是不是奇数, 若是奇数则校验位为 0, 若是偶数则校验位为 1, 保证最后的数值位+校验位得到的二进制数中 1 的个数为奇数。

同理, 偶校验就是看数值位中 1 的个数是不是偶数, 若是偶数则校验位为 0, 若是奇数则校验位为 1, 保证最后的数值位+校验位得到的二进制数中 1 的个数为偶数。

下面我们来看该题的 4 个选项在偶校验情况下正确的字符编码:

对于 A 答案, 二进制数 1100 1011 中 1 的个数为 5 个, 显然 1 的个数 5 是奇数。若是

偶校验，则该数字有数据错误。

对于 B 答案，二进制 1101 0110 中 1 的个数为 5 个，显然 1 的个数 5 也是奇数。若是偶校验，则该数字一定有数据错误。

对于 C 答案，二进制 1100 0001 中 1 的个数为 3 个，显然 1 的个数 3 也是奇数。若是偶校验，则该数字一定有数据错误。

对于 D 答案，二进制 1100 1001 中 1 的个数为 4 个，显然 1 的个数 4 是偶数。若数据正确，则一定是偶校验。

参考答案：D

5. (原书 第 23 题) 一个 16×16 点阵的汉字，需要()字节的存储空间。

A. 24

B. 64

C. 256

D. 32

【解析】字模码是用点阵表示的汉字字形代码，它是汉字的输出形式。字模点阵只能用来构成汉字库，用于汉字的显示输出或打印输出。例如，“英”字的点阵表示如下图 2.1 所示。

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
0						●						●					04H,10H
1						●						●					04H,10H
2		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	7EH,FFH
3						●						●					04H,10H
4						●			●			●					04H,10H
5									●								04H,90H
6				●	●	●	●	●	●	●	●	●	●				0EH,FCF
7				●					●					●			10H,84H
8				●					●					●			10H,84H
9				●					●					●			10H,84H
10		●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	7EH,FFH
11								●		●							0EH,40H
12						●					●						0EH,20H
13						●						●					04H,10H
14					●								●				0EH,08H
15		●	●	●										●	●	●	70H,07H

图 2.1 汉字“英”的点阵表示

【经典错误】

汉字的输入编码(常见的输入编码有数字编码、拼音码和自行编码,不需要详细掌握)、汉字内码、字模码是计算机中用于输入、内部处理、输出三种不同用途的编码,不要混为一谈。

参考答案：D

二. 综合应用题部分

1. (原书 第 2 题) 一种 (7, 4) 海明码的定义为: 由 4 个信息位 x_1, x_2, x_3, x_4 按以下

模 2 加法运算方式构成 3 个校验位 c_1, c_2, c_3 :

$$c_1 = x_1 + x_2 + x_3 \quad (\text{mod } 2)$$

$$c_2 = x_2 + x_3 + x_4 \quad (\text{mod } 2)$$

$$c_3 = x_1 + x_2 + x_4 \quad (\text{mod } 2)$$

将这些信息位和校验位构成码字 w , 即

$$w = \{x_1, x_2, x_3, x_4, c_1, c_2, c_3\} = \{w_1, w_2, w_3, w_4, w_5, w_6, w_7\}$$

- (1). 计算相应的 (7, 4) 海明码的全部码字。
- (2). 用组合逻辑设计一个海明码的编码器和译码器。

【解析】

(1). 根据 (7, 4) 海明码的定义, 可知 4 位信息位共有 16 种二进制编码。我们来求每一个二进制编码对应的校验位。例如, 编码 $x_4 \sim x_1$ 为 0000 时, 校验位

$$c_1 = x_1 + x_2 + x_3 = 0 + 0 + 0 \quad (\text{mod } 2) = 0$$

$$c_2 = x_2 + x_3 + x_4 = 0 + 0 + 0 \quad (\text{mod } 2) = 0$$

$$c_3 = x_1 + x_2 + x_4 = 0 + 0 + 0 \quad (\text{mod } 2) = 0$$

再比如, 取 $x_4 \sim x_1$ 依次为 0011 时, 校验位

$$c_1 = x_1 + x_2 + x_3 = 1 + 1 + 0 \quad (\text{mod } 2) = 0$$

$$c_2 = x_2 + x_3 + x_4 = 1 + 0 + 0 \quad (\text{mod } 2) = 1$$

$$c_3 = x_1 + x_2 + x_4 = 1 + 1 + 0 \quad (\text{mod } 2) = 0$$

同理, 我们可求得 $x_1 \sim x_4$ 对应的 16 个 2 进制编码所产生的校验位, 如表 2.1 所示。

表 2.1 $x_1 \sim x_4$ 对应的 16 个 2 进制编码所产生的校验位

x_1	x_2	x_3	x_4	c_1	c_2	c_3	x_1	x_2	x_3	x_4	c_1	c_2	c_3
0	0	0	0	0	0	0	1	0	0	0	1	0	1
0	0	0	1	0	1	1	1	0	0	1	1	1	0
0	0	1	0	1	1	0	1	0	1	0	0	1	1
0	0	1	1	1	0	1	1	0	1	1	0	0	0
0	1	0	0	1	1	1	1	1	0	0	0	1	0
0	1	0	1	1	0	0	1	1	0	1	0	0	1
0	1	1	0	0	0	1	1	1	1	0	1	0	0
0	1	1	1	0	1	0	1	1	1	1	1	1	1

(2). 根据题目告知的海明码的用数值位计算校验位的方程:

$$c_1 = x_1 + x_2 + x_3$$

$$c_2 = x_2 + x_3 + x_4$$

$$c_3 = x_1 + x_2 + x_4$$

可知该 (7,4) 海明码的编码器如图 2.2 所示。

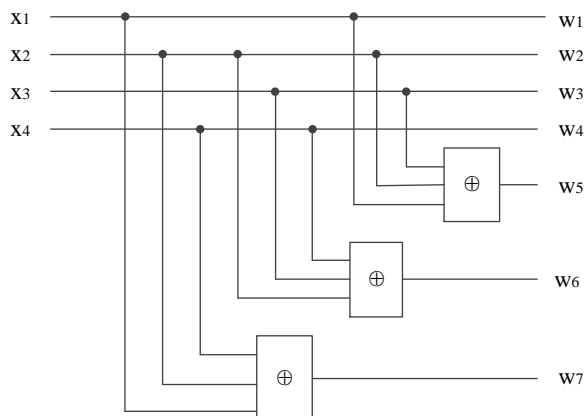


图 2.2 (7,4) 海明码编码器

(7,4) 海明码解码时, 必须能用校验位来表示数值位, 利用题目告知的数值位表示校验位的三个方程, 求解用校验位表示数值位的方程, 最后得到图 2.3 所示的解码电路。

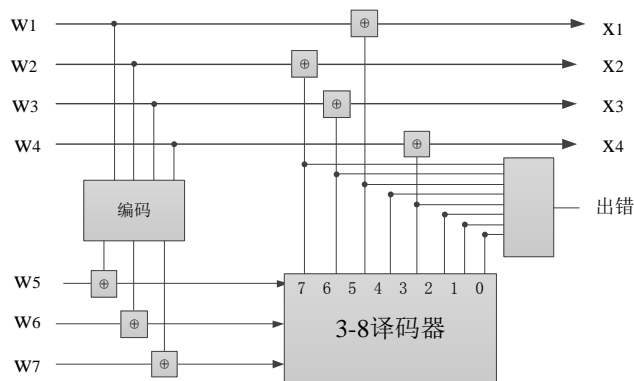


图 2.3 (7,4) 海明码译码器

2. (原书 第 6 题) 写出 $X=1011\ 1101$, $Y=-0010\ 1011$ 的双符号位原码、反码、补码表示, 并用双符号补码计算两个数的和与差。

【解析】X 和 Y 的原码、反码、补码的双符号位表示如表 2.2 所示。

表 2.2 X 和 Y 的原码、反码、补码的双符号位表示

	原码	反码	补码
$X=10111101$	00 1011 1101	00 1011 1101	00 1011 1101
$Y=-00101011$	11 0010 1011	11 1101 0100	11 1101 0101
$X-Y$	00 1110 1000	00 1110 1000	00 1110 1000

下面，我们利用双符号位的补码计算 X 与 Y 的和，即 $X+Y$ 。

$$\begin{array}{r}
 00\ 1011\ 1101 \\
 +\ 11\ 1101\ 0101 \\
 \hline
 00\ 1001\ 0010
 \end{array}$$

接下来，计算 $[X-Y]_{\text{补}}$ ，因为 $[X-Y]_{\text{补}}=[X]_{\text{补}}-[Y]_{\text{补}}=[X]_{\text{补}}+[-Y]_{\text{补}}$ ，而 $[-Y]_{\text{补}}=00\ 0010\ 1011$ ，故而计算过程如下：

$$\begin{array}{r}
 00\ 1011\ 1101 \\
 +\ 00\ 0010\ 1011 \\
 \hline
 00\ 1110\ 1000
 \end{array}$$

可知 $[X-Y]_{\text{补}}=00\ 1110\ 1000$ 。

3. (原书 第 7 题) 设生成多项式为 X^3+X^2+1 ，

- (1). 求数据信息 1011 的 CRC 编码、循环余数和出错模式；
- (2). 如果在接收端收到的信息是 1100000，该信息有没有错误？传送的正确信息是什么？

【解析】

- (1). 多项式为 X^3+X^2+1 ，对应的二进制码为 1101B， $r=3$ ，可计算如下：

$$\begin{array}{r}
 1000 \\
 1101 \overline{) 1011000} \\
 \underline{1101} \\
 1100 \\
 \underline{1101} \\
 100 \dots\dots\dots(1) \\
 1000 \\
 \underline{1101} \\
 101 \dots\dots\dots(2) \\
 1010 \\
 \underline{1101} \\
 111 \dots\dots\dots(3) \\
 1110 \\
 \underline{1101} \\
 11 \dots\dots\dots(4) \\
 110 \dots\dots\dots(5) \\
 1100 \\
 \underline{1101} \\
 1 \dots\dots\dots(6) \\
 10 \dots\dots\dots(7) \\
 100 \dots\dots\dots(1)
 \end{array}$$

从计算过程可以看出，CRC 码为 1011 100，前面 4 位为数据位，后面 3 位为校验位，传送的数据长度是数据位和校验位之和。

如果循环码有 1 位出错，用 $G(x)$ 作模 2 除将得到一个不为 0 的余数。如果对余数补 0 继续除下去，我们将发现一个有趣的结果；各次余数将按如下的顺序循环。

$$\begin{array}{c}
 011 \rightarrow 110 \rightarrow 001 \rightarrow 010 \\
 \uparrow \qquad \qquad \qquad \downarrow \\
 \leftarrow 111 \leftarrow 101 \leftarrow 100
 \end{array}$$

例如，第一位出错，循环余数将为 001，补 0 后再除，第二次余数为 010，以后依次为 100，101...，反复循环，这就是“循环码”名称的由来。

故而，出错模式如表 2.3 所示。

表 2.3 出错模式

传输的信息	循环余数	出错位
1011 100	000	无错
1011 101	001	1
1011 110	010	2
1011 000	100	3
1010 100	101	4
1001 100	111	5
1111 100	011	6
0011 100	110	7

1100 000 除以 1101 的余数是 101，即循环余数为 101，显然第 4 位出错了。于是，将第 4 位的 0 变为 1，传送的正确信息为：1101。

考点 2 定点数表示和运算

温馨提示：定点数的表示部分，主要包括有符号数和无符号数的表示；定点数的运算部分，主要包括定点数的位移运算、原码定点数的加/减运算、补码定点数的加/减运算、定点数的乘/除运算，以及溢出概念和判别方法。其中，unsigned int、short、int、long、float、double 等在 C 语言中表示方法、运算以及溢出判断、隐式类型转换，是 408 统考的常考内容。本考点也是计算机组成原理常出题的地方，请同学们掌握。

一. 选择题部分

1. (原书 第 1 题) 机器运算发生溢出的根本原因是 ()。
 - A. 数据的位数有限
 - B. 运算中将符号位的进位丢弃
 - C. 运算中将符号位的借位丢弃
 - D. 数据运算中的错误

【解析】本题考查机器运算溢出的根本原因。

无论采用何种机器数，只要运算的结果大于数值设备所能表示数的范围，就会产生溢

出。

参考答案：A

2. (原书 第6题) 在整数定点机中, 下述第 () 种说法是正确的。

- A. 原码和反码不能表示 -1 , 补码可以表示 -1
- B. 三种机器数均可表示 -1
- C. 三种机器数均可表示 -1 , 且三种机器数的表示范围相同
- D. 三种机器数均不可表示 -1

【解析】定点整数, 故原码、反码、补码均可以表示 -1 。假设字长为 8, 首位为符号位, 则 -1 分别表示为 10000001、11111110、11111111。

参考答案：B

3. (原书 第7题) 设 $X = -0.1011$, 则 $[X]_{\text{补}}$ 为 ()。

- A. 1.1011
- B. 1.0100
- C. 1.0101
- D. 1.1001

【解析】本题考查负(小)数的补码表示。

$X = -0.1011$, 即 $[X]_{\text{原}} = 1.1011$, 其补码是除符号位外, 每位求反, 再往最低位加 1。我们先将 $[X]_{\text{原}}$ 除符号位之外每位求反, 得到 1.0100, 再往最低位加 1, 得到 $[X]_{\text{补}} = 1.0101$ 。故而, 选择 C 答案。

参考答案：C

4. (原书 第21题) 在下述有关不恢复余数法何时需恢复余数的说法中 () 是正确的。

- A. 最后一次余数为正时, 要恢复一次余数
- B. 最后一次余数为负时, 要恢复一次余数
- C. 最后一次余数为 0 时, 要恢复一次余数
- D. 任何时候都不恢复余数

【解析】本题考查不恢复余数法恢复余数的合理时机。

在定点除法运算时, 为了防止溢出, 要求被除数的绝对值小于除数的绝对值, $|X| < |Y|$ ($|X| = |Y|$ 除外), 且除数不能为 0。因此第一次减除数肯定是不够减的, 如果我们采用先移位后减除数的方法, 得到的结果也是相同的。另外, 在原码加减交替法中, 当最终余数为负数时, 必须恢复一次余数, 使之变为正余数, 注意此时不需要再左移了。

参考答案：B

5. (原书 第 27 题)【2009 年 408 统考】一个 C 语言程序在一台 32 位机器上运行。程序中定义了三个变量 x 、 y 和 z ，其中 x 和 z 为 int 型， y 为 short 型。当 $x = 127$ ， $y = -9$ 时，执行赋值语句 $z = x + y$ 后， x 、 y 和 z 的值分别是 ()。
- A. $x = 0000007FH$ ， $y = FFF9H$ ， $z = 00000076H$
- B. $x = 0000007FH$ ， $y = FFF9H$ ， $z = FFFF0076H$
- C. $x = 0000007FH$ ， $y = FFF7H$ ， $z = FFFF0076H$
- D. $x = 0000007FH$ ， $y = FFF7H$ ， $z = 00000076H$

【解析】本题考查十进制整数到二进制的转换，数据的补码表示。

【经典总结】

该类题型只看运算结果，不看运算过程，我们可以用以下两种方法来解题。

方法 1：把 x 和 y 都转换成 2 进制的补码，再用补码的加法进行计算；

方法 2：直接用原码计算，再将计算结果转换成补码。

这两种方法的结果都是一样的，但是后者的计算速度快一些。

采用方法 1 时，先将 x 和 y 转换成补码。对这两个补码进行加法运算时，需要对两个不同位数的补码数进行符号扩展。

$$x = 127, [x]_{\text{补}} = 0000\ 007FH (\text{int 型})$$

$$y = -9, [y]_{\text{补}} = FFF7H (\text{short 型})$$

先将 y 的符号位进行扩展，得 $[y]_{\text{补}} = FFFF\ FFF7H$

$$[z]_{\text{补}} = [x]_{\text{补}} + [y]_{\text{补}} = [x]_{\text{补}} + [y]_{\text{补}} = 0000\ 007FH + FFFF\ FFF7H = 0000\ 0076H。$$

采用方法 2，先求出 $127 + (-9) = 118$ ，再变成补码的 $0000\ 0076H$ 。

参考答案：D

6. (原书 第 30 题)【2012 年 408 统考】某计算机存储器按字节编址，采用小端方式存放数据。假定编译器规定 int 型和 short 型长度分别为 32 位和 16 位，并且数据按边界对齐存储。某 C 语言程序段如下：

```
struct
{
    int a;
    char b;
    short c;
} record;
record.a = 273;
```


(2). 定点原码小数表示时, 最大正数为多少? 最小负数为多少?

【解析】已知机器字长为 16 位, 定点表示时, 尾数 15 位, 阶符 1 位。那么,

(1). 定点原码整数表示时, 当表示最大的整数, 小数点应该在数值位的后面, 阶符 1 位, 整数位 15 位, 即: 0, 111 1111 1111 1111, 所以最大正数为: $(2^{15}-1)_{10} = (32767)_{10}$; 最小的负整数时, 也是和最大的整数一样的, 只不过阶符为 1, 表示负数, 即 1, 111 1111 1111 1111。显然, 该最小负数为: $-(2^{15}-1)_{10} = (-32767)_{10}$ 。

(2). 定点原码小数表示时, 当表示最大的正数, 则该数为 0.111 1111 1111 1111, 即: $(1-2^{-15})_{10}$; 当表示最小的负数时, 该数为 1.111 1111 1111 1111, 即: $-(1-2^{-15})_{10}$;

2. (原书 第 4 题) 已知 $X=-0.1101$, $Y=0.1011$, 试用补码一位乘法计算 $X \times Y$ 。

【解析】补码一位乘法, 也就是 Booth 算法, 是一种带符号的乘法, 采用相加和相减的操作, 计算补码数据的乘积。设 $[X]_{\text{补}}=x_s.x_1x_2x_3\dots x_n$, $[Y]_{\text{补}}=y_s.y_1y_2y_3\dots y_n$, Booth 算法的运算规则如下:

- (1). 参加运算的数用补码表示;
- (2). 符号位参加运算。由于符号位要参加运算, 部分积累加时最高有效位产生的进位可能会侵占符号位, 故被乘数和部分积应取双符号位, 而乘数只需要一位符号位。
- (3). 乘数最低位后面增加一位附加位 Y_{n+1} , 其初值为 0;
- (4). 由于每求一次部分积要右移一位, 所以乘数的最低两位 Y_n 、 Y_{n+1} 的值决定了每次应执行的操作 (Booth 算法的移位规则如表 2.5 所示);

表 2.5 Booth 算法的移位规则

判断位	操作
0 0	原部分积右移一位
0 1	原部分积加 $[X]_{\text{补}}$ 后右移一位
1 0	原部分积加 $[-X]_{\text{补}}$ 后右移一位
1 1	原部分积右移一位

(5). 移位按补码右移规则进行;

(6). 共需做 $n+1$ 次累加, n 次移位, 第 $n+1$ 次不移位。

$[X]_{\text{补}}=11.0011$, $[-X]_{\text{补}}=00.1101$, $[Y]_{\text{补}}=0.1011$ 。接下来, 我们解析本题, 过程如下:

高位部分积	低位部分积/乘数	说明
00.0000	0.1011 <u>0</u>	初始情况
+ 00.1101		$Y_4Y_5=10$, $Y_5-Y_4=-1$, 所以 $+[-X]_{补}$
00.1101		
右移 00.0110	1.0101 <u>10</u>	右移部分积和乘数
+ 00.0000		$Y_4Y_5=11$, $Y_5-Y_4=0$, 所以 $+0$
00.0110		
右移 00.0011	0.1010 <u>110</u>	右移部分积和乘数
+ 11.0011		$Y_4Y_5=01$, $Y_5-Y_4=1$, 所以 $+ [X]_{补}$
11.0110		
右移 11.1011	0.0101 <u>0110</u>	右移部分积和乘数
+ 00.1101		$Y_4Y_5=10$, $Y_5-Y_4=-1$, 所以 $+ [-X]_{补}$
00.1000		
右移 00.0100	0.0010 <u>10110</u>	右移部分积和乘数
+ 11.0011		$Y_4Y_5=01$, $Y_5-Y_4=1$, 所以 $+ [X]_{补}$
11.0111		

其中, 计算过程中加粗部分构成了 $[X \times Y]_{补}$, 而“低位部分积/乘数”的右侧部分表示移位丢失的位。可见, 其实最开始移到低位的数是最终运算结果的最低位。

从计算过程可以看出, $[X \times Y]_{补}=1.01110001$, 将补码换成原码, 可知 $X \times Y = -0.10001111$ 。

3. (原书 第6题) 已知 $X=0.1001$, $Y=0.1011$, 试用补码不恢复余数法求 $[X \div Y]_{补}$ 。

【解析】补码不恢复余数法的运算规则如下。

- (1). 符号位参与运算, 被除数与除数均用双符号位补码表示;
- (2). 被除数与除数同号, 被除数减去除数。被除数与除数异号, 被除数加上除数。商的符号取值见(3);

(3). 余数与除数同号, 商上 1, 余数左移一位减去除数; 余数与除数异号, 商上 0, 余数左移一位加上除数, 余数左移加/减除数后得到了新余数;

(4). 重复(3), 若采用末位“恒置 1”法, 则包括符号位在内共重复(3) n 次; 若采用校正法包括符号位在内, 则应重复(3) $n+1$ 次, 第 $(n+1)$ 次不左移, 然后决定是否校正。

$[X]_{补}=00.1001$, $[Y]_{补}=00.1011$, $[Y]_{补}=11.0101$, 下面是详细的补码不恢复余数算法计算过程:

	被除数/余数	商	上商位	说明
	00.1001	0.000	<u>0</u>	
$+[-y]_{\text{补}}$	11.0101			同号, 减除数比较
	11.1110	0.000	<u>0</u>	余数 r_0 与除数异号, 商上 0
左移	11.1100	0.00 <u>0</u>		左移一位
$+ [y]_{\text{补}}$	00.1011			加除数比较
	00.0111	0.00 <u>0</u>	<u>1</u>	r_1 与除数同号, 商上 1
左移	00.1110	0.00 <u>1</u>		左移一位
$+ [-y]_{\text{补}}$	11.0101			减除数比较
	00.0011	0.00 <u>1</u>	<u>1</u>	r_2 与除数同号, 商上 1
左移	00.0110	0.0 <u>1</u> 1		左移一位
$+ [-y]_{\text{补}}$	11.0101			减除数比较
	11.1011	0.0 <u>1</u> 1	<u>0</u>	r_3 与除数异号, 商上 0
左移	11.0110	<u>0.1</u> 10		左移一位
$+ [y]_{\text{补}}$	00.1011			加除数比较
	00.0001	<u>0.1</u> 10	<u>1</u>	r_4 与除数同号, 商上 1 ($n+1=5$ 步)

故 $[x/y]_{\text{补}}=0.1101$, 余数 $[r]_{\text{补}}=0.0000001$ 。因未除尽, 商又为正, 因此商不必修正。商为正且余数又与被除数同号, 故余数也不必修正。

考点 3 浮点数的表示和运算

温馨提示: 本考点考查: 1、浮点数的表示和运算, 我们在例题 1 中, 给出了浮点数加法的运算, 请同学们注意解题过程, 并能将解题方法运用自如; 2、IEEE 754 标准, 该考试内容历年 408 统考命题较多, 我们给出了几个例题, 请 408 统考的同学掌握解题方法。

一. 选择题部分

1. (原书 第 1 题)【2009 年 408 统考真题】浮点数加减运算过程一般包括对阶、尾数运算、规格化、舍入和判溢出等步骤。设浮点数的阶码和尾数均采用补码表示, 且位数分别为 5 位和 7 位(均含 2 位符号位)。若有两个数 $X=2^7 \times 29/32$, $Y=2^5 \times 5/8$, 则浮点加法计算 $X+Y$ 的最终结果是: ()。

A. 00111 1100010

B. 00111 0100010

C. 01000 0010001

D. 发生溢出

【解析】两个浮点数相加或相减，首先要把小数点的位置对齐，而浮点数的小数点的实际位置取决于阶码的大小，因此，对齐两数的小数点，就是使两数的阶码相等，这个过程称为对阶。

【经典总结】浮点运算中的对阶方法

要对阶，首先应求出两数阶码 E_A 和 E_B 之差 $\Delta E = E_A - E_B$ 。若 $\Delta E = 0$ ，表示两数阶码相等，即 $E_A = E_B$ ；若 $\Delta E > 0$ ，表示 $E_A > E_B$ ；若 $\Delta E < 0$ ，表示 $E_A < E_B$ 。

当 $E_A \neq E_B$ 时，要通过尾数的移位来改变 E_A 或 E_B 。对阶的规则是：小阶向大阶看齐。要使小阶的阶码增大，则相应的尾数右移，直到两数的阶码相等为止。每右移一位，阶码加 1。

$E_A > E_B$ ，则 M_B 右移。每右移一位， $E_B + 1 \rightarrow E_B$ ，直至 $E_A = E_B$ 为止。

$E_A < E_B$ ，则 M_A 右移。每右移一位， $E_A + 1 \rightarrow E_A$ ，直至 $E_A = E_B$ 为止。

尾数右移后，应对尾数进行舍入。

根据题目已知， X 表示成 00,111; 00,11101， Y 的浮点数规格化为 00,101; 00,10100。下面，我们来详细解答一下运算过程。

(1). 对阶

$X = 2^7 \times 29/32$ ， $Y = 2^5 \times 5/8$ ， X 的阶码是 7， Y 的阶码是 5，将 Y 的阶向 X 阶对齐，即 $Y = 2^7 \times 5/32$ 。

在机器中，阶码相减，即 $00,111 - 00,101 = 00,111 + 11,0111 = 00,010$ 。机器判定 X 的阶码大于 Y 的阶码。根据小阶向大阶对齐的原理，应将 Y 的阶码加 2，尾数右移两位，即 00,111; 00,00101。

(2). 尾数相加

接下来，计算 $X + Y = 5/32 + 29/32 = 34/32$ ，即 00 要进行规格化处理，右移一位，阶码加 1，得 $2^8 \times 34/64$ 。显然，计算结果得 01,000; 00,10001。

在机器中，

$$\begin{array}{r} X+Y=00,111; 00,11101 \\ +00,111; 00,00101 \\ \hline =00,111; 01,00010 \end{array}$$

(3). 规格化

根据第二步的运算结果，可知尾数符号位为 01，需要右规。将尾数右移一位，阶码加 1，得 01,000; 00,10001。

(4). 判断是否溢出

显然，运算结果得到的 01,000; 00,10001 的阶码符号位是 01，发生了溢出。

参考答案：D

2. (原书 第 4 题) 浮点数的表示范围和精度取决于 ()。

- A. 阶码的位数和尾数的机器数形式
- B. 阶码的机器数形式和尾数的位数
- C. 阶码的位数和尾数的位数
- D. 阶码的机器数形式和尾数的机器数形式

【解析】本题考查浮点数的阶码和尾数的作用。

一个浮点数可分为阶码和尾数两个部分，阶码用于表示小数点在该数中的位置，尾数用于表示数的有效数值（如图 2.4 所示）。由于阶码表示小数点的位置，所以阶码总是一个整数，而且既可以是正整数也可以是负整数；尾数可以采用整数或者纯小数两种形式。

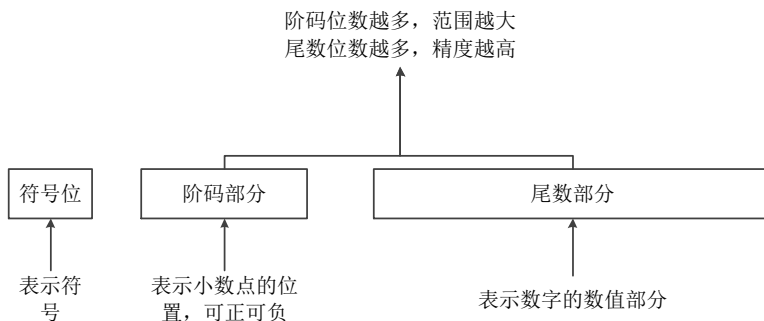


图 2.4 浮点数的表示

浮点数的阶码和尾数占用的尾数可以灵活设定，而且阶码确定数的表示范围，尾数确定数的精度。当字长一定时，阶码分配到的位数越多，表示范围越大；尾数分配到的位数越多，表示精度越高。

参考答案：C

3. (原书 第 8 题) 长度相同但格式不同的 2 种浮点数，假定前者阶段长、尾数短，后者阶段短、尾数长，其他规定均相同，则它们可表示的数的范围和精度为 ()。

- A. 两者可表示的数的范围和精度相同
- B. 前者可表示的数的范围大但精度低
- C. 后者可表示的数的范围大且精度高
- D. 前者可表示的数的范围大且精度高

【解析】本题紧接第 4、5 题，在长度相同的情况下，不同的两种浮点数，阶码长的表

示数的范围大，但是尾数的位数就少了，精度也就降低了，阶码短的表示数的范围小了，但是尾数的位数变多了，精度也就变高了。

参考答案: B

4. (原书第10题) 假设阶码、尾数均为 5 位(含符号位), 二进制浮点数 $2^{-11} \times 0.1011$ 的补码表示是 ()。
- A. 10011 01011 B. 11101 01011
C. 11101 10101 D. 00011 01011

【解析】阶码和尾数均为 5 位，含有一位阶符，那么阶码和尾数的数值位都是 4 位。二进制浮点数 $2^{-11} \times 0.1011$ 是一个正数，而且尾数已经规格化，所以尾数为 01011。该数的阶码是 -11，符号位为 1，数值为 11 即 2 进制的 1011。那么，该数的阶码对应的补码为 1.0101。

【经典总结】

假设纯整数 x 有 n 位，包括符号位。根据移码（浮点数是数字表示方式是阶码用移码表示，尾数用补码表示）计算规则，可得阶码的移码为

$$[X]_{\text{移}} = 2^{n-1} + [X]_{\text{补}}$$

由以上的经典总结,可求得 $[x]_{\text{移}}=1,1101$ 。显然,所求的补码表示是1 1101 01011,故而B答案正确。

参考答案: B

5. (原书 第 13 题) 在浮点机中, 判断补码规格化形式的原则是 ()。
- A. 尾数的第一数位为 1, 数符任意
- B. 尾数的符号位与第一数位相同
- C. 尾数的符号位与第一数位不同
- D. 阶符与数符不同

【解析】 本题考查补码规格化的表示形式。

补码规格化,是因为表示的数字第1位在原码表示中不是1。即0.0xx...xx 或者 1.1xx...xx 形式。规格化之后,正数应为0.1xxx...xxx 形式,负数应为1.0xxx...xxx 形式。

参考答案: C

6. (原书 第 17 题)【2013 年 408 统考】某数采用 IEEE 754 单精度浮点数格式表示为 C640 0000H, 则该数的值是 ()。
- A. -1.5×2^{13} B. -1.5×2^{12}
C. -0.5×2^{13} D. -0.5×2^{12}

【解析】本题考查 IEEE 754 单精度浮点数表示方法。

IEEE 754 单精度浮点数表示方法在近几年的考题里面常出现，请同学们多注意一下这个知识点。学会将浮点数转换成 IEEE 754 规格化数，以及将 IEEE 754 规格化数转换成一般的浮点数的方法。

本题的浮点数为 C640 0000H，其对应的 2 进制数如下表 2.6 所示。

表 2.6 浮点数 C640 0000H 对应 IEEE 754 的 2 进制数

1 为数符	8 位阶码（含阶符，移码表示）	23 位尾数
1	100 0110 0	100 0000 0000 0000 0000

从上表可以看出，该数是一个负数，阶码为 100 0110 0B-111 1111 1B=000 0110 1B，尾数为 1.100 0000 0000 0000 0000，即 -1.5×2^{13} 。

参考答案：A

二. 综合应用题部分

1. (原书 第 1 题) 有两个浮点数 $X=2^{(+01)}_2 \times (-0.111)_2$ ， $Y=2^{(+01)}_2 \times (+0.101)_2$ ，设阶码 2 位，阶符 1 位，数符 1 位，尾数 3 位，用补码运算规则计算 $X-Y$ 的值。

【解析】设 S_x 为 x 的尾数， S_y 为 y 的尾数，则 $S_x=(-0.111)_2$ ， $[S_x]_{\text{补}}=1.001$ ， $S_y=(+0.101)_2$ ， $[S_y]_{\text{补}}=0.101$ 。又设 E_x 为 x 的阶码， E_y 为 y 的阶码，则 $E_x=(+01)_2$ ， $[E_x]_{\text{补}}=001$ ， $E_y=(+10)_2$ ， $[E_y]_{\text{补}}=010$ 。

(1). 对阶

$E_x-E_y=(01)_2-(10)_2=(-01)_2$ ，阶码不相等，故小阶的尾数 S_x 右移一位， $S_x=(-0.0111)_2$ ， E_x 阶码加 1，则 $E_x=(10)_2=E_y$ ， S_x 经舍入后得 $S_x=(-0.100)_2$ ，对阶完毕。

X 的补码浮点格式： 010 ， 1100

Y 的补码浮点格式： 010 ， 0101

(2). 尾数相减

$[S_y]_{\text{补}}=11.100$ ， $[-S_y]_{\text{补}}=11.011$ ，那么，

$$\begin{array}{r} [S_x]_{\text{补}}=11.100 \\ + \quad [S_y]_{\text{补}}=11.011 \\ \hline [S_x-S_y]_{\text{补}}= 10.111 \end{array}$$

(3). 规格化与舍入

因为 $[S_x-S_y]_{\text{补}}= 10.111$ ，显然需要右规。于是，将尾数右移 1 位，最低有效位舍掉，阶

码加 1 (右规) 则

$$[S_x - S_y]_{\text{补}} = 11.011$$

$$[E_x]_{\text{补}} = [E_y]_{\text{补}} = 011$$

可知, 规格化结果为: 011, 1011。

2. (**原书 第 3 题**) 设浮点数字长为 16 位, 其中阶码为 5 位(含 1 位阶符), 尾数为 11 位(含 1 位数符), 写出 -23/128 对应的浮点规格化数的原码形式、补码形式、反码形式和阶码用移码, 尾数用补码的形式。

【解析】根据题意, 浮点数字长是 16 位, 阶码为 5 位(含一位阶符), 尾数为 11 位(含 1 位数符)。-11 写成 2 进制数即 -1011, -11/128 只需将 -1011 右移 7 位即可, 移位的结果为 -0.0001011, 即 -0.1011×2^{-11} 。

则浮点规格化数的原码形式为:

$$1, 0011; 1.1011000000$$

浮点规格化数的补码形式为:

$$1, 1101; 1.0101000000$$

反码的运算规则为原码除符号位(包括阶码和尾数的符号位)之外, 每位求反(包括阶码和尾数的数值位)。浮点规格化数 1, 0011; 1.1011000000 的反码形式为:

$$1, 1100; 1.0100111111$$

移码(又叫增码)是符号位取反的补码, 一般用做浮点数的阶码, 引入的目的是为了保证浮点数的机器零为全 0。

浮点规格化数阶码用移码, 尾数用补码是常见的一种表述形式。1, 0011; 1.1011000000 的阶码为 1, 0011, 用移码表示为 0, 1101。所以该数的阶码用移码, 尾数用补码表示为:

$$0, 1101; 1.0101000000$$

3. (**原书 第 6 题**) 某机器字长 32 位, 浮点表示时, 阶码占 8 位, 尾数占 24 位, 各包含一位符号位。问:
- (1). 带符号定点小数的最大表示范围是多少?
 - (2). 带符号定点整数的最大范围是多少?
 - (3). 浮点表示的最大正数是多少? 最大负数是多少? 浮点表示时, 最小的规格化正数是多少?

【解析】

- (1). 阶码占 8 位, 阶符占 1 位, 所以阶码的表示范围是: $-128 \sim 127$; 尾数占 24 位,

包含 1 位数符，所以尾数的表示范围是： $-1 \sim 1 - 2^{-23}$ 。

(2). 带符号的定点整数其符号位 1 位，其余的 31 位均用来表示数据，小数点位于数据位的后面，此时的表示范围 $-2^{31} \sim 2^{31} - 1$ 。

(3). 根据题目已知，机器字长为 32 位，阶码占 8 位，尾数占 24 位，且都包含 1 位符号位。可知，浮点表示的最大正数为： $1 - 2^{-23} \times 2^{127}$ ，最大负数为： $-2^{-23} \times 2^{-128}$ ，最小的规格化正数为： $2^{-1} \times 2^{-128}$ 。

考点 4 算数逻辑单元 ALU

温馨提示：本部分考查 ALU 的功能和结构，并考查串行加法和并行加法器，请同学们做一个简单的了解。有些自主命题高校，可能让同学们设计加法器，要求比较高，请同学们根据报考学校的要求来做好复习工作，我们根据统考要求，不对本部分作太深的分析。

一. 选择题部分

1. (原书 第 2 题) 有关运算器的功能描述，正确的是 ()。

- A. 完成加法运算
- B. 完成算术运算
- C. 既完成算术运算又完成逻辑运算
- D. 完成逻辑运算

【解析】本题考查运算器的功能。

运算器的主要功能是对数据进行各种运算。主要包括两个方面的运算：一方面，运算器可以进行常规的加、减、乘、除等基本的算术运算；另一个方面，运算器还能进行“逻辑判断”，如“与”、“或”、“非”这样的基本逻辑运算。

此外，运算器还可以进行数据的比较、移位等操作。

参考答案：C

2. (原书 第 3 题) 算术 / 逻辑运算单元 74181ALU 可完成 ()。

- A. 16 种算术运算功能
- B. 16 种逻辑运算功能
- C. 16 种算术运算功能和 16 种逻辑运算功能
- D. 4 位乘法运算和除法运算功能

【解析】74181ALU 芯片如图 2.5 所示。

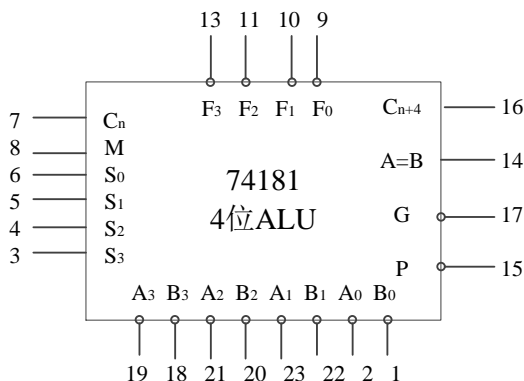


图 2.5 74181ALU 芯片结构

图 2.5 中，对于负逻辑而言，左边表示输入， C_n 表示进位， M 表示控制端为算术运算或者逻辑运算， $A=B$ 表示输出两个数相等， C_{n+4} 表示进位输出。对于正逻辑而言， S_3 表示工作方式选择输入， G 表示进位产生， P 表示进位传送， $F_3 \sim F_0$ 表示 4 位和。

74181ALU 两种工作方式。对正逻辑操作数来说，算术运算称高电平操作，逻辑运算称正逻辑操作(即高电平为“1”，低电平为“0”)。对于负逻辑操作数来说，正好相反。由于 $S_0 \sim S_3$ 有 16 种状态组合，因此对正逻辑输入与输出而言，有 16 种算术运算功能和 16 种逻辑运算功能。同样，对于负逻辑输入与输出而言，也有 16 种算术运算功能和 16 种逻辑运算功能。

参考答案：C

3. (原书 第 6 题) 四片 74181ALU 和一片 74182CLA 器件相配合，具有如下进位传递功能 ()。
- A. 形波进位
 - B. 组内先行进位，组间先行进位
 - C. 组内先行进位，组间行波进位
 - D. 组内形波进位，组间先行进位

【解析】74181 是 4 位的内部先行进位 ALU 芯片，74182 是 4 位的先行进位芯片。故而，4 片 74181 芯片和 1 片 74182 芯片可组成两级先行进位结构的 16 位 ALU。这 16 位 ALU 的进位传递功能是组内先行进位和组间先行进位。故而，本题选择 B 答案。

参考答案：B

4. (原书 第 9 题) 串行加法器采用先行进位的目的是 ()。
- A. 优化加法器的结构
 - B. 节省器材

- C. 加速传递进位信号
- D. 增强加法器结构

【解析】本题考查串行加法器先行进位的目的。

串行进位加法器需要一级一级的进位，进位延迟很大。先行进位加法器（也叫超前进位加法器）可以有效的减少进位延迟。

参考答案：C

本章到此就结束了，请问您有什么疑问吗？任何问题，欢迎您与我们作者进行交流！



梦享团队微信号



梦享团队官方微信公众号



梦享团队新浪微博

EPROM 的缺点是，不能作为随机存储器使用。

参考答案：B

3. (原书 第4题) 存储单元是指 ()。

- A. 存放一个二进制信息位的存储元
- B. 存放一个机器字的所有存储元集合
- C. 存放一个字节的存储元集合
- D. 存放两个字节的存储元集合

【解析】本题考查存储单元的概念。

一个二进制的码位为最小的存储单位，即存储位元。若干个存储位元组成一个存储单元。存储单元是可以存放一个机器字并具有特定存储地址的存储单位。

参考答案：B

4. (原书 第7题) 若磁盘的转速提高一倍，则 ()。

- A. 平均存取时间减半
- B. 平均找道时间减半
- C. 平均等待时间减半
- D. 存储密度可以提高一倍

【解析】本题考查磁盘的转速和转速提高对磁盘的影响。

讲到这里，我们顺便回顾一下磁盘的结构。在硬盘中信息分布呈以下层次：记录面、圆柱面、磁道和扇区(如图 3.1 所示)。

(1). 记录面

一台硬盘驱动器中有多个盘片，每个盘片有两个记录面，每个记录面对应一个磁头，所以记录面号就是磁头号。所有的磁头安装在一个公用的转动设备或支架上，磁头一致地沿盘面径向移动，单个磁头不能单独地移动。

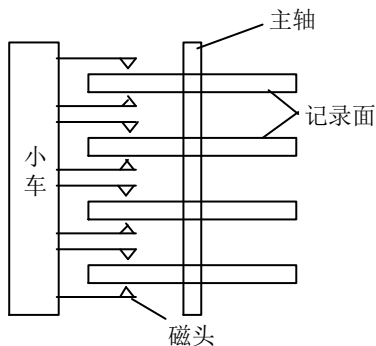


图 3.1 磁盘的记录面

(2). 磁盘柱面

在一个盘组中,各记录面上相同编号(位置)的诸磁道构成一个圆柱面(如图 3.2 所示)。引入圆柱面的概念是为了提高硬盘的存储速度。当主机要存入一个较长的文件时,若一条磁道存不完,就需要存放在同一圆柱面上的不同磁道上。由于各记录面的磁头已同时定位,换道的时间只是磁头选择电路的译码时间,相对于定位操作可以忽略不计。

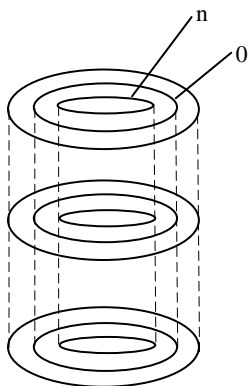


图 3.2 磁盘柱面

(3). 磁盘扇区

通常将一条磁道划分为若干个段,每个段称为一个扇区或扇段(如图 3.3 所示),每个扇区存放一个定长信息块(如 512 个字节)。一条磁道划分多少扇区,每个扇区可存放多少字节,一般由操作系统决定。磁道上的扇区编号从 1 开始,不像磁头或柱面编号从 0 开始。

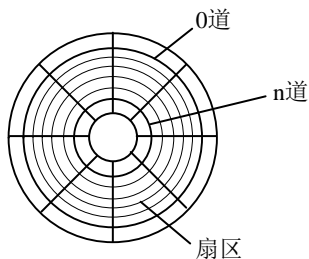


图 3.3 磁盘扇区

主机向磁盘控制器送出有关寻址信息,磁盘地址一般表示为:

(圆柱面(磁道)号,记录面(磁头)号,扇区号)

通过多磁盘物理结构的分析,我们可以得知,当磁盘的转速提高一倍,则平均等待时间减半,但是平均寻道时间和存储密度均与转速无关。因为平均存取时间=寻道时间+等待时间+读写操作时间,所以光是把等待时间减半显然不能使得平均存取时间减半。

参考答案: C

5. (原书 第 13 题) 下列关于存储器的描述,不正确的是 ()。

- A. SRAM 和 DRAM 都是易失性存储器
- B. ROM 存储器内容是预置的, 固定的, 无法改写
- C. 多模块交叉存储器主要是解决主存空间不够大的问题
- D. Cache 存储器是为了解决 CPU 和主存之间在速度上不匹配的问题

【解析】本题是一个综合性较强的选择题, 考查的内容涵盖了 RAM 和 ROM 的特点、多模块交叉存储器的作用和 Cache 存储器的作用等。

按掉电后是否还保存数据或程序分为易失与非易失存储器。

(1). 易失存储器, 通常称为 RAM (Random Access Memory 随机存取存储器), 主要包括 SRAM 与 DRAM。

(2). 非易失存储器, 通常称为 ROM (read-only memory 只读存储器), 主要包括 FLASH、EEPROM、EPROM 等。

可见, A 答案的说法正确。我们一般认为, ROM 的内容是无法改写的, RAM 的内容可读可写, 故而 B 答案正确。

目前, 主存的存取速度已成为计算机系统的瓶颈, 除去通过寻找高速元件来提高访问速度外, 也可以采用多个存储器并行工作, 并且用交叉访问技术来提高存储器的访问速度。多模块交叉访问存储器中有多个容量相同的存储模块 (存储体), 而且各存储模块具有各自独立的地址寄存器、读写电路和数据寄存器, 这就是多体系统。多体系统中, 各个存储体能并行工作, 又能交叉工作。所以, C 答案错误。

高速缓冲技术利用程序的局部性原理, 把程序中正在使用的部分存放在一个高速的容量较小的 Cache 中, 使 CPU 的访存操作大多数针对 Cache 进行, 从而使程序的执行速度大大提高, 从很大程度上解决了 CPU 与主存之间的速度不匹配问题。故而, D 答案正确。

参考答案: C

考点 2 存储器的层次化结构

温馨提示: 本考点考查存储器的层次化结构, 包括寄存器、Cache、主存和辅存几种层次, 请同学们注意存储器层次化的目的, 层次化的方法, 以及相邻各层之间的特点。

一. 选择题部分

1. (原书 第 1 题) 层次化存储器结构的设计是依据 () 原理。
 - A. 存储器周期性
 - B. 存储器强制性

C. 访存局部性

D. 容量失效性

【解析】本题考查层次化存储器结构设计原理。

为了解决存储容量、存取速度和价格之间的矛盾，通常把各种不同存储容量、不同存取速度的存储器，按一定的体系结构组织起来，形成一个统一整体的存储系统。层次化存储器结构设计的依据是访问局部性原理。

参考答案：C

2. (原书 第2题) 和外存储器相比，内存储器的特点是 ()。

- A. 容量大、速度快、成本低
- B. 容量大、速度慢、成本高
- C. 容量小、速度快、成本高
- D. 容量小、速度快、成本低

【解析】本题考查内部存储器和外部存储器的特点。

与外存相比，内存的容量一般比较小，如我们的笔记本电脑的内存一般 2GB，外存几百个 GB。内存的速度一般比外存快很多。对于相同容量的内存和外存，一般内存也比外存成本高很多。

参考答案：C

3. (原书 第3题) 存储器的最大存储容量取决于 ()。

- A. 指令中地址码位数
- B. 指令字长
- C. 寻址方式形成的存储器有效地址位数
- D. 造价成本

【解析】本题考查决定存储器最大容量的因子。

存储器的最大存储容量（显然，本题指的是内存），取决于机器的地址结构，即寻址方式形成的存储器有效地址位数。比如，对于 32 位的笔记本电脑，最大的内存只能是 4GB，若是给这个电脑装 16G 的内存，显然没有什么作用。

参考答案：C

4. (原书 第5题) 计算机的存储器系统是指 ()。

- A. RAM 存储器
- B. ROM 存储器
- C. 主存储器
- D. 主存储器和外存储器

【解析】本题考查计算机存储系统的概念。

计算机的存储系统，是指内存储器和外存储器。主存储器是整个存储系统的核心，它用来存放计算机运行期间所需要的程序和数据，CPU 可直接随机地对它进行访问。外部存储器可以用来存放需要比较长时间保存的程序和数据。

参考答案：D

二. 综合应用题部分

1. (原书 第 1 题) Cache—主存存储系统和主存—辅存存储系统有何不同？

【解析】Cache 存储系统（如图 3.4 所示）是为解决主存速度不足而提出来的。从 CPU 看，Cache 存储系统的速度接近 Cache 的速度，容量是主存的容量，每位价格接近于主存的价格。由于 Cache 存储系统全部用硬件来调度，因此它对系统程序员和应用程序员都是透明的。

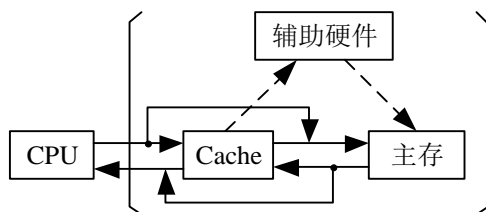


图 3.4 cache 存储系统

主存—辅存存储系统，即虚拟存储系统（如图 3.5 所示）。虚拟存储系统是为解决主存容量不足而提出来的。从 CPU 看，主存—辅存存储系统的速度接近主存的速度，容量是虚拟的地址空间，每位价格是接近于辅存的价格。由于虚拟存储系统需要通过操作系统来调度，因此对系统程序员是不透明的，但对应用程序员是透明的。

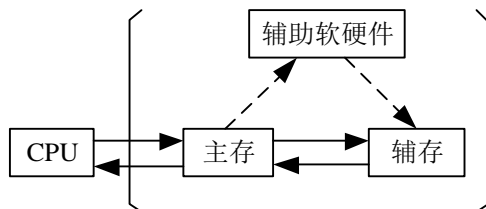


图 3.5 主存-辅存存储系统

考点 3 半导体随机存取存储器

温馨提示：半导体随机存取存储器包括 DRAM 和 SRAM，请同学们注意这两种存储器的工作原理和区别。因为在考研真题中，常将 RAM 和 ROM 结合在一起考查，所以我们也把 ROM 的内容融入本考点中。请同学们注意 RAM 和 ROM 的区别。

一. 选择题部分

1. (原书第1题) 有一个 $1\text{K} \times 1$ 的DRAM芯片，芯片内是 32×32 结构，采用分散刷新方式，如果刷新闻隔不超过 2ms ，刷新信号的周期是()。
- A. $62.5\mu\text{s}$ B. $6.25\mu\text{s}$
- C. $0.625\mu\text{s}$ D. 2ms

【解析】 本题考查存储器的分散刷新。

分散式刷新，将每个系统工作周期分为两部分，前半部分用于 DRAM 读/写/保持，后半部分用于刷新存储器的一行。故而，系统存取时间延长一倍，导致系统变慢。

DRAM 是 1K×1，芯片内部采用 32×32 的结构，采用分散刷新方式，在刷新间隔不超过 2ms 的情况下，刷新 32 行，每行平均可以在 $2\text{ms} \div 32 = 62.5\mu\text{s}$ 的时间内刷新一次。所以，刷新信号的周期为 62.5us。

参考答案：A

2. (原书第3题) 某计算机的字长 16 位，它的存储容量是 64KB，若按字编址，那么它的寻址范围是（ ）。
- A. 64K B. 32K
- C. 64KB D. 32KB

【解析】根据本题已知条件，机器的字长是 16 位，按字编址，则存储容量为 64KB 的存储空间，寻址范围为 $64\text{KB}/2\text{B}=32\text{K}$ ，选择 B 答案。

一般情况下，遇到类似的题目，都可以先求存储器的总容量，然后除去编址方式的位数（可能与机器字长有关），得到的结果就是寻址范围。

参考答案: B

3. (原书 第 4 题) 下述说法中 () 是正确的。
- A. 半导体 RAM 信息可读可写, 且断电后仍能保持记忆。
- B. 半导体 RAM 是易失性 RAM, 而静态 RAM 中的存储信息是不易失的。

本部分资料内容均从原书《2016 年考研核心考点命题思路解密—计算机组成原理》一书中精选的少部分题目, 供给考生免费使用, 任何考研机构用于盈利均为盗版, 必追究责任!

- C. 半导体 RAM 是易失性 RAM，而静态 RAM 只有在电源不掉时，所存信息是不易失的。

【解析】本题考查半导体 RAM 和静态 RAM 的特点。

DRAM 均属于易失性存储器，断电后信息丢失。即使没有断电，DRAM 也会因为电容的放电作用使信息丢失，所以要长期保存数据必须定期刷新存储单元。与 DRAM 不同的是，SRAM 只要不掉电可以永久保存信息，所以不需要老刷新。所以，C 选项正确。但是，不掉电信息不易失不代表 SRAM 是非易失性存储器。实际上，我们一般把掉电后信息不丢失的存储器当做非易失存储器。

【经典总结】

本题的答案十分经典，请同学们记住这个结论：SRAM 和 DRAM 均属于 RAM，都属于易失性存储器，断电后信息丢失。DRAM 在没有断电的情况下，因为电容的放电作用也会是信息丢失，所以要经常刷新。SRAM 只要不掉电就可以永久保存信息，所以不需要老刷新。

参考答案：C

4. (原书 第 7 题) 以下说法错误的是 ()。
- A. DRAM 和 SRAM 的存取时间都小于存取周期
 - B. DRAM 的存取时间小于存取周期，SRAM 的存取时间等于存取周期
 - C. DRAM 的存取时间等于存取周期，SRAM 的存取时间小于存取周期
 - D. DRAM 和 SRAM 的存取时间都等于存取周期

【解析】本题考查 DRAM 和 SRAM 的存取时间与存取周期的关系。

存取时间是指启动一次存储器操作到完成该操作所需要的时间，包括读出时间和写入时间。存取周期是指存储器进行两次连续存取操作的最小时间间隔。

对于 DRAM 来说，从存储器收到由 CPU 发来的地址开始，到数据线上产生有效输出的时间为读出时间。但由于 DRAM 的读取破坏了原信息，故在下次读操作前，还需要一次写入来恢复存储单元的值，故而存取时间一定小于存取周期。

对于 SRAM 来说，虽然其读出是非破坏性的，不需要重写，但是数据的读取仍将对存储单元的电平等状态造成一定的影响。故而，在一次读取前也需要一定的“恢复”时间，所以 SRAM 的存取时间也小于存取周期。

参考答案：A

5. (原书 第 10 题) 【2012 年 408 统考】下列关于闪存 (Flash Memory) 的叙述中，错

误的是 ()。

- A. 信息可读可写，并且读、写速度一样快
- B. 存储元由 MOS 管组成，是一种半导体存储器
- C. 掉电后信息不丢失，是一种非易失性存储器
- D. 采用随机访问方式，可替代计算机外部存储器

【解析】本题考查了闪存的特点。

闪存，我们常用的 U 盘就是其中一种，此外 SD 卡等也是常见的闪存卡。闪存的写操作必须在空白区域进行，如果目标区域已经有数据，必须先擦除后写入，而读操作不必如此，所以闪存的读速度比写速度快。其他三项均为闪存的特征，记住即可。

参考答案：A

二. 综合应用题部分

1. (原书 第 2 题) 静态存储器和动态存储器器件的特性有哪些主要区别？

【解析】本题考查静态存储器和动态存储器的区别。

静态存储器 SRAM 主要用于高速缓冲存储器 Cache，动态存储器主要用于主存储器。在表 3.1，我们比较一下半导体存储器的分类和特点。

表 3.1 常见的半导体存储器的分类和特点

分 类	名 称	特 点
随机存取存储器 (RAM)	静态RAM (SRAM)	速度快、工作稳定、不需要刷新电路
	动态RAM(DRAM)	集成度高，功耗低、需要刷新电路
	非易失性RAM(NVRAM)	在掉电或电源发生故障时，可自动保存信息
只读存储器 (ROM)	掩膜ROM	在生产过程中信息一次性直接写入，内容不能再改写
	可编成ROM	信息只能写入一次，不能擦除和改写
	可擦除可编程ROM(EPROM)	写入的信息，可以多次改写
	电可擦除可编程ROM(EEPROM)	具有RAM和ROM的双重功能特点，不需要专用编程工具，信息可以多次改写
	闪存存储器 (flash memory)	在线电擦写，低功耗，大容量，擦写速度快
新型存储器	同步突发SRAM	
	管道突发SRAM	
	快速页模式DRAM	
	扩充数据输出RAM	
	同步DRAM	

2. (原书 第4题)动态存储器为什么需要定时刷新? CPU 能否在刷新期间访问存储器? 如何进行控制?

【解析】DRAM 存储位元是基于电容器上的电荷量存储, 这个电荷随着时间和温度而减少, 因此需要定期的刷新, 以保持他们原来记忆的正确信息。

对于集中式刷新, CPU 不能在刷新期间访问存储器, 对于分散式刷新, 可以对存储器中没有正在刷新的行进行访问。

集中式刷新有固定的刷新周期, 当刷新时间到来时, 数据线输出被封锁, 刷新结束后再开始正常的读、写操作。

分散式刷新有固定的刷新周期, 但是每次刷新单位为行, 没有被刷新到的行可以让 CPU 访问, 刷新控制使用 DRAM 内部的刷新计数器, 当计数器加 1 则产生下一个需要刷新的地址。

3. 对具有 1024 个记忆单元 (32×32 的存储矩阵) 的存储芯片进行刷新, 假设存取周期为 500ns (0.5s), 试分析采用集中刷新、分散刷新和异步刷新三种方式的刷新情况。

【解析】在允许的最大刷新间隔 (如 2ms) 内, 按照存储芯片容量的大小集中安排若干个刷新周期, 刷新时停止读写操作。

刷新时间=存储矩阵行数 \times 刷新周期

这里刷新周期是指刷新一行所需要的时间, 由于刷新过程就是“假读”的过程, 所以刷新周期就等于存取周期。

(1). 集中刷新

对具有 1024 个记忆单元 (32×32 的存储矩阵) 的存储芯片进行刷新, 刷新是按行进行的, 且每刷新一行占用一个存取周期, 所以共需 32 个周期以完成全部记忆单元的刷新。存取周期为 500ns (0.5s), 从 0~3967 个周期内进行读写操作或保持, 而从 3968~3999 这最后 32 个周期集中安排刷新操作 (如图 3.6 所示)。

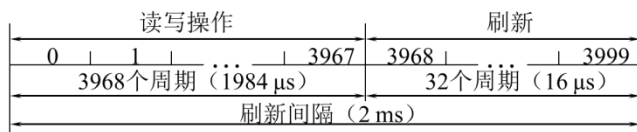


图 3.6 集中刷新方式

集中刷新方式的优点是读写操作时不受刷新工作的影响, 因此系统的存取速度比较高。主要缺点是在集中刷新期间必须停止读写, 这一段时间称为“死区”, 而且存储容量越大, 死

区就越长。

(2). 分散刷新

分散刷新是指把刷新操作分散到每个存取周期内进行, 此时系统的存取周期被分为两部分, 前一部分时间进行读写操作或保持, 后一部分时间进行刷新操作。在一个系统存取周期内刷新存储矩阵中的一行。

这种刷新方式增加了系统的存取周期, 如存储芯片的存取周期为 $0.5\mu\text{s}$, 则系统的存取周期应为 $1\mu\text{s}$ 。我们仍以前述的 32×32 矩阵为例, 整个存储芯片刷新一遍需要 $32\mu\text{s}$ (如图 3.7 所示)。

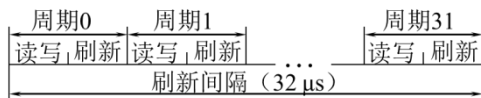


图 3.7 分散刷新

分散刷新方式的优点是没有死区。但是, 该刷新方式也有很明显的缺点, 第一是加长了系统的存取周期, 降低了整机的速度; 第二是刷新过于频繁 (本例中每 $32\mu\text{s}$ 就重复刷新一遍), 尤其是当存储容量比较小的情况下, 没有充分利用所允许的最大刷新间隔 (2ms)。

(3). 异步刷新

这种刷新方式是前两种方式的结合, 它充分利用了最大刷新间隔时间, 把刷新操作平均分配到整个最大刷新间隔时间内进行, 故有:

相邻两行的刷新间隔 = 最大刷新间隔时间 \div 行数

对于 32×32 矩阵, 在 2ms 内需要将 32 行刷新一遍, 所以相邻两行的刷新时间间隔 = $2\text{ms} \div 32 = 62.5\mu\text{s}$, 即每隔 $62.5\mu\text{s}$ 安排一个刷新周期。在刷新时封锁读写 (如图 3.8 所示)。

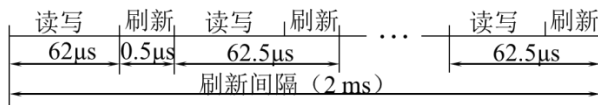


图 3.8 异步刷新方式

异步刷新方式虽然也有死区, 但比集中刷新方式的死区小得多, 仅为 $0.5\mu\text{s}$ 。这样可以避免使 CPU 连续等待过长的时间, 而且减少了刷新次数, 是比较实用的一种刷新方式。

考点 4 主存储器与 CPU 的连接

温馨提示: 主存储器和 CPU 的连接, 是历年计算机组成原理考研真题中常出现的内容。本考点主要以大题的形式出现, 而且分值很高, 请同学们务必掌握主存储器和 CPU 连接方法和连接图的画法。

一. 选择题部分

1. (原书 第1题)【2009年408统考】某计算机主存容量为64KB。其中ROM区为4KB，其余为RAM区，按字节编址；现要用 $2K \times 8$ 位的ROM芯片和 $4K \times 4$ 位的RAM芯片来设计该存储器，则需要上述规格的ROM芯片数和RAM芯片数分别是：()。
- A. 1、15 B. 2、15
C. 1、30 D. 2、30

【解析】根据题意，该计算机 ROM 区是 4KB，RAM 区是 60KB。由 $2K \times 8$ 位的 ROM 芯片来构建该计算机的 4KB 的 ROM 区，需要芯片数量为 $4KB / (2K \times 8 \text{ 位}) = 2$ 片。由 $4K \times 4$ 位的 RAM 芯片来构建该计算机的 60KB 的 RAM 区，需要该种芯片共 $60KB / (4K \times 4 \text{ 位}) = 30$ 片。所以，选择 D 答案。

参考答案: D

2. (原书第4题) 某 DRAM 芯片，其存储容量为 512K×8 位，该芯片的地址线和数据线数目为（ ）。
- A. 8、512 B. 512、8
- C. 18、8 D. 19、8

【解析】和第 3 题的计算方法类似，在存储器的位扩展、字扩展和位字同时扩展三种方式下计算数据线和地址线的根数这类题型都是比较简单的题型。对于该 DRAM 芯片，因为其存储容量为 $512\text{K} \times 8$ 位，所以地址线是 19 根，数据线是 8 根。

参考答案: D

3. (原书第6题) 将 RAM 芯片的数据线、地址线和读写控制线分别接在一起, 而将片选信号线单独连接, 其目的 ()。
- A. 增加存储器字长
- B. 增加存储单元数量
- C. 提高存储器速度
- D. 降低存储器的平均价格

【解析】本题描述的是存储器扩展。在存储器扩展中，将片选信号线单独连接，便于选中不同的芯片。

参考答案: B

4. (原书第8题) 某计算机字长是 32 位，它的存储容量是 256KB，按字编址，它的寻址范围是（ ）。
- A. 128K B. 64K
C. 64KB D. 128KB

【解析】计算机字长是 32 位，按照字编址时，32 位大小的存储单元作为一个单元进行访问。故而，当该机器的存储容量为 256KB 时，能访问到的以 32 位大小的存储单元的个数为 $256\text{KB}/4\text{B}=64\text{K}$ 。

参考答案: B

5. (原书第11题)某计算机主存容量为7KB。其中ROM区为4KB。其余为RAM区，按字节编址，则地址范围为()。
- A. 1000~2BFFH B. 0000~1FFFH
C. 0000~1CFFH D. 1000~2FFFH

【解析】依题意，该计算机主存容量为 7KB，按照字节编址，寻址范围是 0~7K，所以地址范围是 0000~1FFFH。

参考答案: B

二. 综合应用题部分

1. (原书第1题) 设 CPU 共有 16 根地址线, 8 根数据线, 并用 $\overline{\text{MREQ}}$ 作访存控制信号 (低电平有效), 用 $\overline{\text{WR}}$ 作读写控制信号 (高电平为读, 低电平为写)。现有下列芯片及各种门电路 (门电路自定), 如图 3.9 所示。画出 CPU 与存储器的连接图, 要求:
- (1). 存储芯片地址空间分配为: 最大 4K 地址空间为系统程序区, 相邻的 4K 地址空间为系统程序工作区, 最小 16K 地址空间为用户程序区;
 - (2). 指出选用的存储芯片类型及数量;
 - (3). 详细画出片选逻辑。

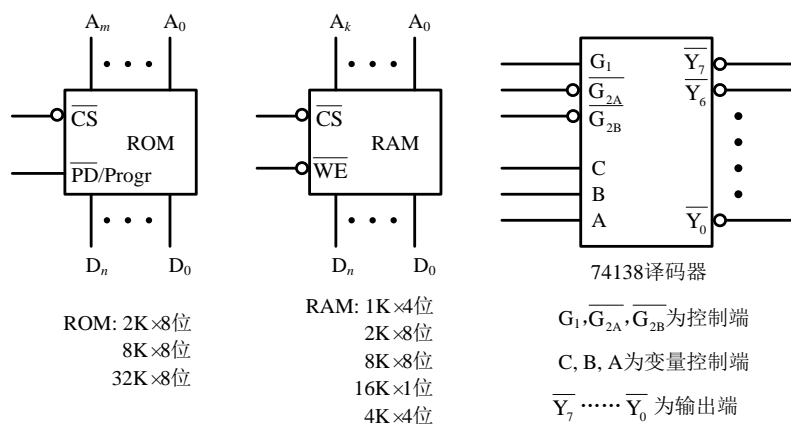


图 3.9 相关的芯片和门电路

【解析】

(1). 主存地址空间分配如下。

A_{15}	...	A_{11}	...	A_7	A_0	
1	1	1	1	1	1	1	1	1
1	1	1	1	1	0	0	0	0
1	1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0	0
								最大 4K 2K×8 位 ROM 2 片
1	1	1	0	1	1	1	1	1
1	1	1	0	0	0	0	0	0
								相邻 4K 4K×4 位 RAM 2 片
0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1	1
0	0	1	0	0	0	0	0	0
0	0	1	1	1	1	1	1	1
								最小 16K 8K×8 位 RAM 2 片

(2). 根据主存地址空间分配

最大 4K 地址空间为系统程序区, 选用 2 片 2K×8 位 ROM 芯片; 相邻的 4K 地址空间为系统程序工作区, 选用 2 片 4K×4 位 RAM 芯片; 最小 16K 地址空间为用户程序区, 选用 2 片 8K×8 位 RAM 芯片。

(3). 存储芯片的片选逻辑图如图 3.10 所示。

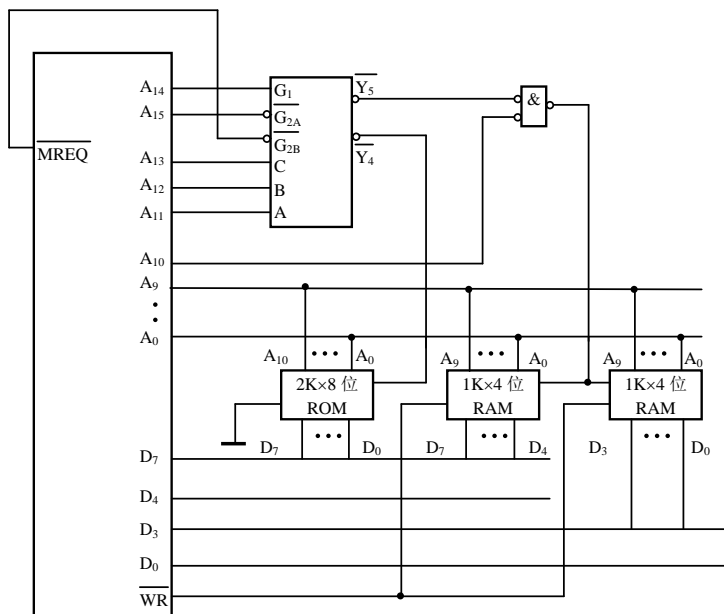


图 3.10 存储芯片的片选逻辑

【经典总结】3:8 译码器

3:8 译码器的结构如图 3.11 所示。该译码器用于地址译码的 3-8 译码器，输入 3 位地址信号，译码产生 8 个不同的选通输出。

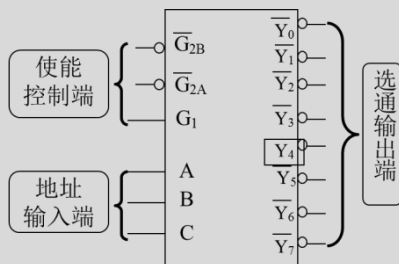


图 3.11 3:8 译码器管脚作用

从图 3.11 可以得知，各引脚作用如下：

- ① 输入信号 A、B、C。引入所要译码的三位地址信号。
- ② 输出信号 $\overline{Y}_0 \sim \overline{Y}_7$ 。对应每一个存储单元，低电平有效。
- ③ 使能信号 G_1 、 \overline{G}_{2A} 、 \overline{G}_{2B} 。当且仅当 $G_1=1$ 、 $\overline{G}_{2A}=0$ 、 $\overline{G}_{2B}=0$ 时，译码器正常工作。

2. (原书 第 3 题) 设 CPU 共有 16 根地址线，8 根数据线，并用 \overline{MREQ} 作访存控制信

号, 用 R/\overline{W} 作读写控制信号, 现有下列存储芯片:

RAM: 1K×8 位、2K×4 位、4K×8 位

ROM: 2K×8 位、4K×8 位

以及 74138 译码器和各种门电路 (自定), 画出 CPU 与存储器连接图。要求:

- (1). 最大 4K 地址空间为系统程序区, 与其相邻 2K 地址空间为用户程序区。
- (2). 合理选用上述存储芯片, 说明各选几片? 写出每片存储芯片的地址范围。
- (3). 详细画出存储芯片的片选逻辑。

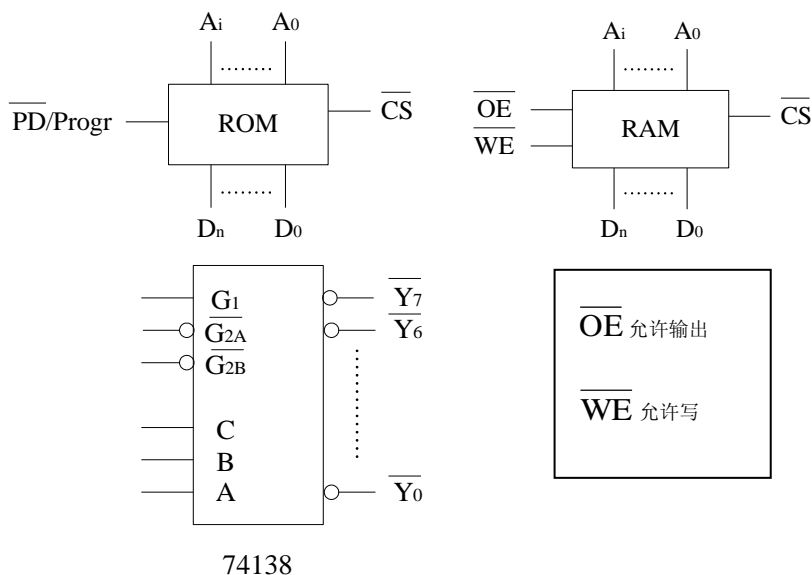


图 3.12 相关的芯片和信号

【解析】ROM 的大小为 4K, 设对应的空间范围是: 1111 0000 0000 0000 到 1111 1111 1111 1111。RAM 的大小为 2K, 对应的空间范围为: 1110 1000 0000 0000 到 11101111

1111 1111。根据题意, 我们可以选择 2 片 2K×8 位的 ROM 芯片, 2 片 2K×4 位的 RAM 芯片。

我们取 ROM 芯片 1 的地址范围为 1111 1000 0000 0000 到 1111 1111 1111 1111, ROM 芯片 2 的地址范围为 1111 0000 0000 0000 到 1111 0111 1111 1111。

RAM 芯片 1、2 用来实现位扩展, 对应的地址空间范围为 1110 1000 0000 0000 到 11101111 1111 1111。

CPU 与存储器连接图如图 3.13 所示。

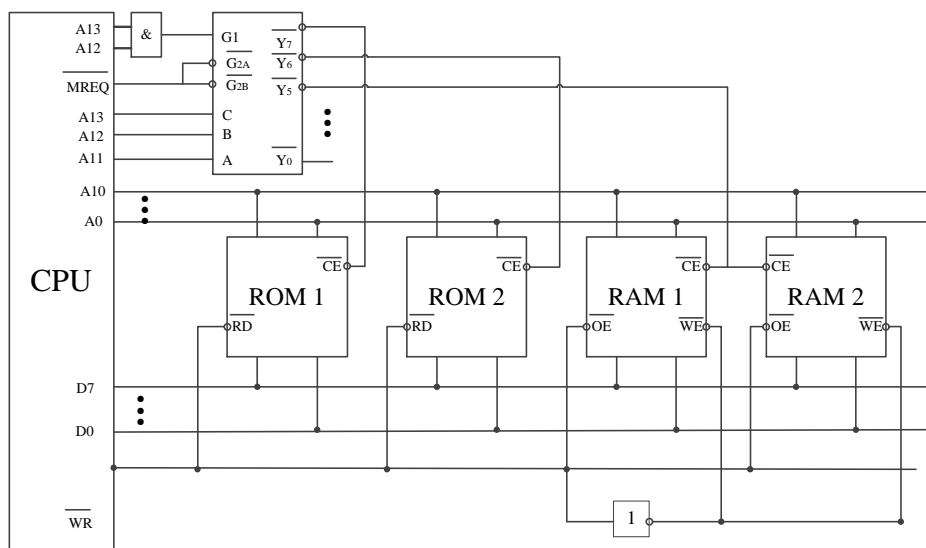


图 3.13 CPU 与存储器连接图

考点 5 双口 RAM 和多模块存储器

温馨提示：本考点考查双口 RAM 和多模块存储器，请同学们注意区别低位交叉存储器和高位交叉存储器。

一. 选择题部分

1. (原书 第 1 题) 交叉存储器实质上是一种 () 存储器，它能 () 执行 () 独立的读写操作。 ()
- A. 多体，并行，多个
B. 多体，串行，多个
C. 整体，并行，一个
D. 整体，串行，多个

【解析】本题考查交叉存储器。

目前，主存的存取速度已成为计算机系统的瓶颈，除去通过寻找高速元件来提高访问速度外，也可以采用多个存储器并行工作，并且用交叉访问技术来提高存储器的访问速度。

交叉访问存储器中有多个容量相同的存储模块（存储体），而且各存储模块具有各自独立的地址寄存器、读写电路和数据寄存器，这就是多体系统。各个存储体能并行工作，又能交叉工作。

参考答案：A

2. (原书 第 2 题) 交叉存储器实质上是一种多模块存储器, 它用 () 方式执行多个独立的读写操作。

- A. 流水
- B. 资源重复
- C. 顺序
- D. 资源共享

【解析】本题考查交叉存储器的工作特点。

多体交叉存储器有以下特点:

(1). 用地址码的低位字段经过译码选择不同的模块, 而高位字段指向相应模块内的存储字。

(2). 连续地址分布在相邻的不同模块内, 而同一个模块内的地址都是不连续的。

(3). 从定性分析, 对连续字的成块传送, 交叉方式的存储器可以实现多模块流水式并行存取, 大大提高存储器的带宽。

(4). 多模块交叉存储器的思想是基于: 能同时从主存取出 n 条指令, 以提高机器的运行速度。

参考答案：A

3. (原书 第 4 题) 双端口存储器所以能高速进行读写, 是因为采用 ()。

- A. 高速芯片
- B. 两套相互独立的读写电路
- C. 流水技术
- D. 新型器件

【解析】本题考查双端口存储器的特点。

双端口存储器是指同一个存储器具有两组相互独立的读写控制线路, 由于进行并行的独立操作, 是一种高速工作的存储器。

当两个端口的地址不相同, 在两个端口上进行读写操作, 一定不会发生冲突。当任一端口被选中驱动时, 就可对整个存储器进行存取, 每一个端口都有自己的片选控制和输出驱动控制。

当两个端口同时存取存储器同一存储单元时, 便发生读写冲突。为解决此问题, 特设置了 BUSY 标志。由片上的判断逻辑决定对哪个端口优先进行读写操作, 而暂时关闭另一个被延迟的端口。

参考答案：B

二. 综合应用题部分

1. **(原书 第3题)** 设存储器容量为 128M 字, 字长 64 位, 模块数 $m=8$, 分别用顺序方式和交叉方式进行组织。存储周期 $T=200\text{ns}$, 数据总线宽度为 64 位, 总线传送周期 $\tau=50\text{ns}$ 。问顺序存储器和交叉存储器的带宽各是多少?

【解析】 顺序存储器和交叉存储器连续读出 8 个字的信息总量都是:

$$q = 64 \text{ 位} \times 8 = 512 \text{ 位}$$

顺序存储器和交叉存储器连续读出 8 个字所需的时间分别是:

$$t_2 = MT = 8 \times 200\text{ns} = 1600\text{ns}$$

$$t_1 = T + (m - 1)\tau = 200 + 7 \times 50\text{ns} = 550\text{ns}$$

顺序存储器和交叉存储器的带宽分别为

$$w_2 = \frac{q}{t_2} = 32 \times 10^7 \text{ bit/s}$$

$$w_1 = \frac{q}{t_1} = 93 \times 10^7 \text{ bit/s}$$

2. **(原书 第4题)** 有一个具有 8 个存储体的低位交叉存储器中, 如果处理器的访问地址为以下八进制地址值, 求该存储器比单体存储器的平均访问速度提高多少? (忽略最初的启动时延)
- (1). $1001_8, 1002_8, 1003_8, \dots, 1100_8$;
 - (2). $1002_8, 1004_8, 1006_8, \dots, 1200_8$;
 - (3). $1003_8, 1006_8, 1011_8, \dots, 1300_8$ 。

【解析】 由题的三个序列可知, 每个访问序列总共访问 64 个存储单元。假设存储周期为 T , 那么在不使用低位交叉存储体的情况下访存耗时都为 $64T$ 。

- (1). 访问序列为 $1001_8, 1002_8, 1003_8, \dots, 1100_8$ 时, 该访问序列访存如表 3.2 所示。

表 3.2 访问序列对应的访存情况表

M0	M1	M2	M3	M4	M5	M6	M7
...
1000_8	1001_8	1002_8	1003_8	1004_8	1005_8	1006_8	1007_8
1010_8	1011_8	1012_8	1013_8	1014_8	1015_8	1016_8	1017_8
1020_8	1021_8	1022_8
...
1100_8							

可见,存储体每 $T/8$ 就能启动一个,每 $1/8$ 个存储周期 T 才能访问到下一个要访问的存储单元。显然,访问该段序列所用时间为 $T+63 \times T/8 = 71T/8$,速度提升了 7.2 倍。

(2). 当访问序列为 $1002_8, 1004_8, 1006_8, \dots, 1200_8$ 时,该访问序列访存如表 3.3 所示。

表 3.3 访问序列对应的访存情况表

M0	M1	M2	M3	M4	M5	M6	M7
...
...	...	1002 ₈	...	1004 ₈	...	1006 ₈	...
1010 ₈
1020 ₈
...
1200 ₈							

可见,虽然存储体每 $T/8$ 就能启动一个,但是每 $2/8$ 个存储周期 T 才能访问到下一个要访问的存储单元。则访问该段序列所用时间为 $3T + 63 \times 2T/8 = 75T/4$,速度提升了 3.4 倍。

(3). 当访问序列为 $1003_8, 1006_8, 1011_8, \dots, 1300_8$ 时,该访问序列访存如表 3.4 所示。

表 3.4 访问序列对应的访存情况表

M0	M1	M2	M3	M4	M5	M6	M7
...
...	1003 ₈	1006 ₈	...
...	1011 ₈	1014 ₈	1017 ₈
...	...	1022 ₈	1025 ₈
1030 ₈	1033 ₈	1036 ₈	...
...	1041 ₈	1044 ₈	1047 ₈
...
1300 ₈							

可见,虽然存储体每 $T/8$ 就能启动一个,但是每 $3/8$ 个存储周期 T 才能访问到下一个要访问的存储单元。则访问该段序列所用时间为 $4T + 63 \times 3T/8 = 221T/8$,速度提升了 2.3 倍。

考点 6 高速缓冲存储器 (Cache)

温馨提示：本考点考查：1、程序的局部性原理；2、Cache 的基本工作原理；3、Cache 与主存之间的映射方式；4、Cache 与主存块的替换算法；5、Cache 写策略。Cache 是一个很重要的考点，请同学们务必掌握。

一. 选择题部分

1. (原书 第 1 题)【2009 年 408 统考】某计算机的 Cache 共有 16 块，采用 2 路组相联映射方式，每个主存块大小为 32 字节，按字节编址。主存 129 号单元所在主存块应装入到 Cache 的组号是：()。
- A. 0 B. 2 C. 4 D. 6

【解析】本题考查组相联映射方式地址换算方法。

一个组含有几个块就称为几路组相联。2 路-组相联映射，说明一个组有两个块。该计算机的 Cache 是 16 块，所以分成 8 个组。每个主存块大小是 32 字节，按字节编址，每个主存块可有 4 个编址，所以主存的 129 号单元在第 33 个主存块。利用公式 $33\%8=4$ 可知所对应的 Cache 组号是 4。

参考答案：C

2. (原书 第 3 题)在 Cache 更新时，把数据同时写入 Cache 和主存的策略是()。
- A. 写直达 B. 写回法
C. 按写分配法 D. 不按写分配法

【解析】本题考查 Cache 的更新（写回）策略。

cache 命中时的写回策略有：

- (1). 写直达法（数据同时写入 Cache 和主存）；
- (2). 写回法（数据只写入 Cache，不写入主存。仅当需要替换时，才把修改过的 Cache 块写回主存，然后再调入新块）。

写 Cache 不命中时的策略有：

- (1). 不按写分配法（写直达法一般采用该方法）写 Cache 不命中时只写入主存，该单元所在块不调入 Cache；
- (2). 按写分配法（写回法一般采用该方法）写 Cache 不命中时除写入主存外，还将该地址所在块调入 Cache。

参考答案：A

3. (原书 第 5 题) 主存地址与 Cache 地址的三种映象方式为 ()。

- A. 直接映像、段页式映像、组相联映像
- B. 直接映像、全相联映像、组相联映像
- C. 直接映像、段式映像、组相联映像
- D. 直接映像、全相联映像、页式映像

【解析】本题考查主存地址与 Cache 地址的映射方式。我们来看看几种常见的 Cache 映射：

(1). 全相联映像

全相联映像就是让主存中任何一个块均可以映像装入到 Cache 中任何一个块的位置上 (如图 3.14 所示)。全相联映像方式比较灵活, Cache 的块冲突概率最低、空间利用率最高, 但是地址变换速度慢, 而且成本高, 实现起来比较困难。

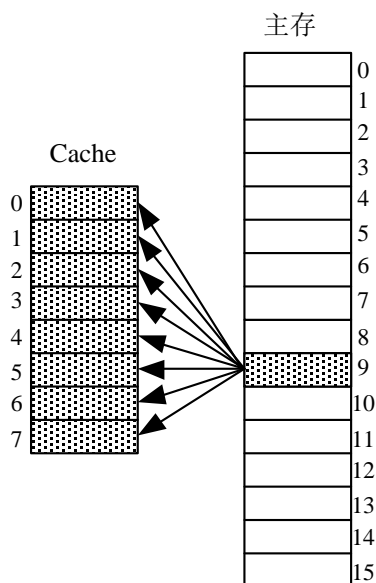


图 3.14 全相联映射

(2). 直接映像

直接映像是指主存中的每一个块只能被放置到 Cache 中惟一的一个指定位置, 若这个位置已有内容, 则产生块冲突, 原来的块将无条件地被替换出去 (如图 3.16 所示)。直接映像方式是最简单的地址映象方式, 成本低, 易实现, 地址变换速度快, 而且不涉及其它两种映像方式中的替换算法问题。但这种方式不够灵活, Cache 的块冲突概率最高、空间利用率最低。

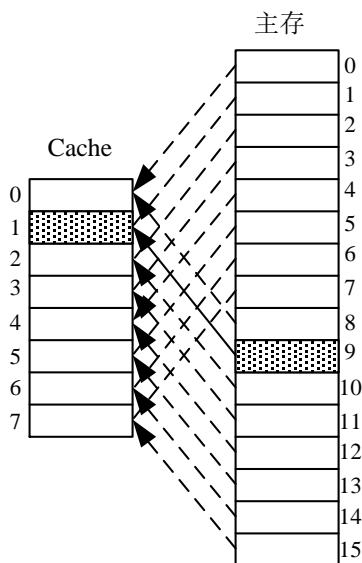


图 3.16 直接映射

(3). 组相联映像

组相联映像将主存空间按 Cache 大小等分成区后，再将 Cache 空间和主存空间中的每一区都等分成大小相同的组（如图 3.17 所示）。让主存各区中某组中的任何一块，均可直接映像装入 Cache 中对应组的任何一块位置上，即组间采取直接映像，而组内采取全相联映像。

组相联映像实际上是全相联映像和直接映像的折衷方案，所以其优点和缺点介于全相联和直接映像方式的优缺点之间。

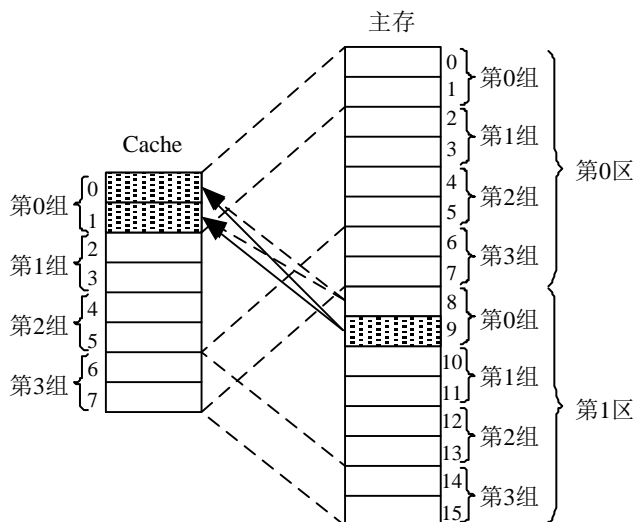


图 3.17 组相联映像

参考答案: B

4. (原书 第 11 题) 下列关于 cache 地址映射的描述, 不正确的是 ()。
- A. 全相联映射方式中, 主存的一个块可能存放到 cache 中任意一行
 - B. 直接映射方式中, 主存的一个块只能存放在 cache 的一个特定行
 - C. 全相联映射方式的 cache 利用率高, 直接映射方式的 cache 利用率低
 - D. 组相联映射方式是全相联映射和直接映射方式的折中方案, 即主存中的一个块放到 cache 的哪个组是灵活的, 而放到该组的哪个行是固定的

【解析】本题考查 Cache 的地址映射策略。

全相联映射方式中, 主存的一个块可以映射到 Cache 中的任意一行, 而直接映射方式中, 主存中的一个块只能映射到 Cache 中的一个特定行。所以, 全相联映射方式的 Cache 利用率一般比直接映射方式下 Cache 的利用率要高, A、B、C 答案正确。

组相联映射的确是全相联映射和直接映射方式的这种方案, 主存中的一个块放到 Cache 中的哪一个组是固定的, 但是放到该组中的哪一个行是灵活的。所以, D 答案错误。

参考答案: D

二. 综合应用题部分

1. (原书 第 3 题) CPU 执行一段程序时, cache 完成存取的次数为 1900 次, 主存完成存取的次数为 100 次, 已知 cache 存取周期为 50ns, 主存存取周期为 250ns, 求 cache/主存系统的效率和平均访问时间。

【解析】CPU 执行完一段程序时, Cache 完成的存取次数为 $N_c=1900$, 主存完成存取的次数为 $N_m=100$, 所以 Cache 的命中率

$$h = N_c / (N_c + N_m) = 1900 / (1900 + 100) = 0.95$$

Cache 存取周期为 $t_c=50\text{ns}$, 主存存取周期为 $t_m=250\text{ns}$, 那么平均访问时间 $t_a = h t_c + (1-h) t_m = 0.95 \times 50 + (1-0.95) \times 250 = 60\text{ns}$ 。故而, Cache/主存系统的效率为

$$e = t_c / t_a = 50 / 60 = 83.3\%$$

2. (原书 第 5 题) 一个组相联映像 Cache 有 64 个存储块构成, 每组包含 4 个存储块, 主存包含 4096 个存储块, 每块由 128 个字组成, 以字作为访存地址单位。
- (1). 计算主存地址有多少位? Cache 地址有多少位? 并画出地址结构图。
 - (2). 计算主存地址格式中区号、组号、块号和块内地址字段的位数。

【解析】

(1). 主存包含 $4096=2^{12}$ 个块, 所以主存块地址是 12 位; Cache 中包含 $64=2^6$ 个块, 所以 Cache 的块地址是 6 位 (包括组号和块号); 每个块包含 $128=2^7$ 个字, 所以块内地址是 7 位。所以主存地址为 $12+7=19$ 位, Cache 地址为 $6+7=13$ 位。如图 3.18 所示。

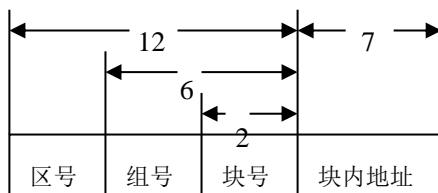


图 3.18 Cache 地址结构

(2). 由上图可以计算出: 区号为 $19-13=6$ 位, 组号为 $6-2=4$ 位, 块号为 2 位, 块内地址为 7 位。

3. (原书 第 13 题) 机字长 8 位, CPU 地址总线 16 位, 数据总线 8 位, 存储器按字节编址, CPU 的控制信号线有: MREQ# (存储器访问请求, 低电平有效), R/W# (读写控制, 低电平为写信号, 高电平为读信号)。试问:

(1). 若该机主存采用 $16K \times 1$ 位的 DRAM 芯片 (内部为 128×128 阵列) 构成最大主存空间, 则共需多少个芯片? 若采用异步刷新方式, 单元刷新周期为 2ms, 则刷新信号的周期为多少毫秒?

(2). 若为该机配备 $2K \times 8$ 位的 Cache, 每字块 4 字节, 采用 4 路组相联映象, 则主存地址中字段块内地址、字段 Cache 组地址、字段高位标记各多少位? 若主存地址为 1234H, 则该地址映象到的 Cache 的第几组?

(3). 若 CPU 执行一段时间时, Cache 完成存取的次数为 2400 次, 主存完成的存取次数为 100 次, 已知 cache 的存储周期为 20ns, 主存的存储周期为 100ns。则 Cache/主存系统的平均访问时间为多少 ns? Cache/主存系统的效率又是多少?

【解析】

(1). 根据题意, 该机器字长 8 位, CPU 地址线 16 位, 其主存的最大存储空间为 $2^{16} \times 8$ 位。若采用 $16K \times 1$ 位的芯片, 需要芯片的数量为 $(2^{16} \times 8) / (16K \times 1) = 32$ 片。16K 的 DRAM 芯片内采用 128×128 阵列, 在 2ms 内刷新 128 行, 所以刷新信号的周期为 $2ms/128 = 1/64$ ms。

(2). 该机器配备 $2K \times 8$ 位的 Cache, 每字块 4 个字节, 按照字节编址方式, 字段块内地址是 2 位。

每个组 4 个存储块，就是 4 路组相联，即将四个存储块合并，在组内采用全相联映射，在组间采用直接映射方式。Cache 的空间大小可以用 2K 表示，所以字段 Cache 组地址 7 位。因为主存的地址空间是 16 位，所以必须要 7 位作为字段高位标记。

(3). Cache 完成存取次数为 2400 次, 主存完成的存取次数为 100 次, 总共完成了 2500 次存取, 所以 Cache 的命中率为 $2400/2500=96\%$ 。Cache/主存系统的平均访问时间为 $0.04 \times 100\text{ns} + 0.96 \times 20\text{ns} = 23.2\text{ns}$ 。

考点 7 虚拟存储器

温馨提示：本部分主要考查虚拟存储器的基本概念，页式、段式、段页式三种虚拟存储器，以及快表 TLB。在 408 统考中，本考点的内容与《计算机操作系统》的存储管理部分的虚拟存储器，是一致的。本考点容易考查大题，请同学们务必掌握。

一. 选择题部分

1. (原书第1题) 一个分段存储管理系统中，地址长度为32位，其中段号8位，则最大段长是()。
- A. 2^8 字节 B. 2^{18} 字节
C. 2^{24} 字节 D. 2^{32} 字节

A. 2^8 字节

B. 2^{18} 字节

C. 2^{24} 字节

D. 2^{32} 字节

【解析】 本题考查段地址结构和最大段长。

段式虚拟存储器中的段是按照程序的逻辑结构划分的，各个段的长度因程序而异。为了把程序虚地址变换成主存实地址，需要一个段表(如图 3.19 所示)。**段表中每一行记录了某个段对应的若干信息，包括段号、装入位、段起点和段长等。**由于段的大小可变，所以在段表中要给出各段的起始地址与段的长度。段表实际上是程序的逻辑结构段与其在主存中所存放的位置之间的关系对照表。

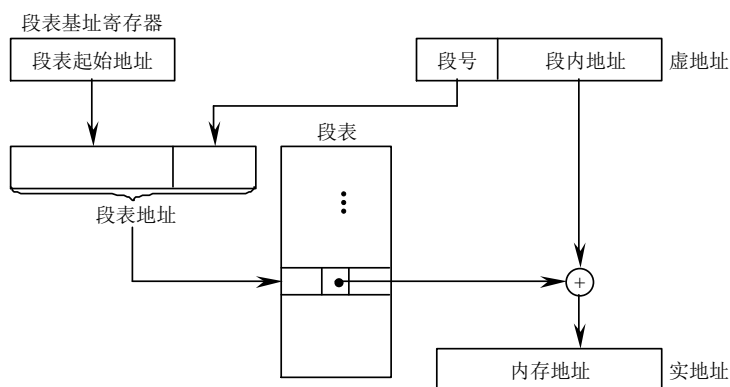


图 3.19 段式虚拟系统的虚拟-物理地址转换

在分段式存储管理系统中，地址的长度是 32 位，段号为 8 位，最段内地址是 24 位，最大段长是 2^{24} 字节。

参考答案：C

2. (原书 第 3 题) 采用虚拟存储器的目的是 ()。

- A. 提高主存储器的存取速度
- B. 扩大主存储器的存储空间，并能进行自动管理和调度
- C. 提高外存储器的存取速度
- D. 扩大外存储器的存储空间

【解析】本题考查采用虚拟存储器的目的。

虚拟存储系统是为解决主存容量不足而提出来的。从 CPU 看，速度接近主存的速度，容量是虚拟的地址空间，每位价格接近于辅存的价格。由于虚拟存储系统需要通过操作系统来调度，因此对系统程序员是不透明的，但对应用程序员是透明的。

参考答案：B

3. (原书 第 4 题) 常用的虚拟存储系统由 () 两级存储器组成，其中后者是大容量磁表面存储器。

- A. 主存和辅存
- B. Cache 和主存
- C. Cache 和辅存
- D. 寄存器和主存

【解析】本题考查虚拟存储器系统的组成。

虚拟存储器由主存储器和联机工作的辅助存储器（通常为磁盘存储器）共同组成，这两个存储器在硬件和系统软件的共同管理下工作，对于应用程序员，可以把它们看作是一个单一的存储器。

参考答案：A

4. (原书 第6题) 虚拟存储器中段页式存储管理方案的特性为 ()。

- A. 空间浪费大, 存储共享不易, 存储保护容易, 不能动态连接
- B. 空间浪费小, 存储共享容易, 存储保护不易, 不能动态连接
- C. 空间浪费大, 存储共享不易, 存储保护容易, 能动态连接
- D. 空间浪费小, 存储共享容易, 存储保护容易, 能动态连接

【解析】本题考查虚拟存储器中段页式存储管理方式的特点。

【经典总结】

这里的段页式存储器, 和《计算机操作系统》一书中所提到的段页式存储器是一回事。从其他科目学到的基础知识, 要学会在《计算机组成原理》中融会贯通。

页式存储管理的优点是页表对程序员来说是透明的, 地址变换快, 调入操作简单; 缺点是各页不是程序的独立模块, 不便于实现程序和数据保护。

段式存储管理的优点是消除了内存零头, 易于实现存储保护, 便于程序动态装配; 缺点是调入操作复杂, 地址变换速度慢于页式存储管理。

段页式存储管理, 优点是便于动态申请内存、管理和使用归一化、便于共享和保护、便于动态链接, 缺点是有碎片问题。因为段页式存储, 页划分比较小, 所以空间浪费也比较小。

参考答案：D

二. 综合应用题部分

1. (原书 第2题) 下图 5 表示使用页表的虚实地址转换条件, 页表存放在相联存储器中, 其容量为 8 个存储单元, 求:

页号	该页在主存中的起始地址
33	42000
25	38000
7	96000
6	60000
4	40000
15	80000
5	50000
30	70000

虚拟地址	页号	页内地址
1	15	0324
2	7	0128
	48	0516

- (1). 当 CPU 按虚拟地址 1 去访问主存时, 主存的实地码是多少?
- (2). 当 CPU 按虚拟地址 2 去访问主存时, 主存的实地码是多少?
- (3). 当 CPU 按虚拟地址 3 去访问主存时, 主存的实地码是多少?

【解析】

(1). 用虚拟地址为 1 的页号 15 作为页表检索项, 查得页号为 15 的页在主存中的起始地址为 80000, 故将 80000 与虚拟地址中的页内地址码 0324 相加, 求得主存实地地址码为 80324。

(2). 同理, 主存实地地址码 = $96000 + 0128 = 96128$ 。

(3). 虚拟地址为 3 的页号为 48, 查页表时, 发现此页面没在页表中, 此时操作系统暂停用户作业程序的执行, 转去查页表程序。如该页面在主存中, 则将该页号及该页在主存中的起始地址写入主存; 如该页面不在主存中, 则操作系统要将该页面从外存调入主存, 然后将页号及其主存中的起始地址写入页表。

2. (原书 第 3 题) 请求分页管理系统中, 假设某进程的页表内容如表 3.5 所示。

表 3.5 某进程的页表内容

页号	页框号	有效位 (存在位)
0	101H	1
1	--	0
2	254H	1

页面大小为 4KB, 一次内存的访问时间是 100ns, 一次快表 (TLB) 的访问时间是 10ns, 处理一次缺页的平均时间为 10^8ns (已含更新 TLB 和页表的时间), 进程的驻留集大小固定为 2, 采用最近最少使用置换算法 (LRU) 和局部淘汰策略。假设

- (1). TLB 初始为空;
- (2). 地址转换时先访问 TLB, 若 TLB 未命中, 再访问页表 (忽略访问页表之后的 TLB 更新时间);
- (3). 有效位为 0 表示页面不在内存, 产生缺页中断, 缺页中断处理后, 返回到产生缺页中断的指令处重新执行。

设有虚地址访问序列 2362H、1565H、25A5H, 请问:

- (1). 依次访问上述三个虚地址, 各需多少时间? 给出计算过程。
- (2). 基于上述访问序列, 虚地址 1565H 的物理地址是多少? 请说明理由。

【解析】

(1). 根据页式管理的工作原理, 应先考虑页面大小, 以便将页号和页内位移分解出来。页面大小为 4KB, 即 2^{12} , 则得到页内位移占虚地址的低 12 位, 页号占剩余高位。可得 3 个虚地址的页号 P 如下 (16 进制的一位数字转换成 4 位二进制。因此, 十六进制的低 3 位正好为页内位移, 最高位为页号):

① 2362H: P=2, 访问快表 10ns, 因初始为空, 访问页表 100ns 得到页框号, 合成物理地址后访问主存 100ns, 共计 $10\text{ns}+100\text{ns}+100\text{ns}=210\text{ns}$ 。

② 1565H: P=1, 访问快表 10ns, 落空, 访问页表 100ns 落空, 进行缺页中断处理 10^8ns , 访问快表 10ns, 合成物理地址后访问主存 100ns, 共计 $10\text{ns}+100\text{ns}+10^8\text{ns}+10\text{ns}+100\text{ns}\approx 10^8\text{ns}$ 。

③ 25A5H: P=2, 访问快表, 因第一次访问已将该页号放入快表, 因此花费 10ns 便可合成物理地址, 访问主存 100ns, 共计 $10\text{ns}+100\text{ns}=110\text{ns}$ 。

(2). 当访问虚地址 1565H 时, 产生缺页中断, 合法驻留集为 2, 必须从页表中淘汰一个页面, 根据题目的置换算法, 应淘汰 0 号页面, 因此 1565H 的对应页框号为 101H。由此可得 1565H 的物理地址为 101565H。

本章到此就结束了, 请问您有什么疑问吗? 任何问题, 欢迎您与我们作者进行交流!



梦享团队微信号



梦享团队官方微信公众号



梦享团队新浪微博

第四章 指令系统

考点 1 指令格式

温馨提示：本部分考查指令的基本格式、定长操作码指令格式和扩展操作码指令。在历年考研中，有些自主命题高校喜欢考查扩展操作码，请同学们根据自己所考学校的要求作复习。本部分不是 408 统考命题位置，我们不布置过多习题。更多的题目，请参考梦享团队编写的《2016 年考研核心考点命题思路解密——计算机组成原理》。

一. 选择题部分

1. (原书 第 1 题) 程序控制类指令的功能是 ()。
- A. 进行算术运算和逻辑运算
 - B. 进行主存与 CPU 之间的数据传送
 - C. 进行 CPU 和 I/O 设备之间的数据传送
 - D. 改变程序执行顺序

【解析】本题考查程序控制类指令的功能。

一般情况下，CPU 执行程序是按照指令的顺序逐条执行的，但实际上程序不可能总是顺序执行，而经常需要改变程序的执行流程，转移到所要求的目标地址去执行，这就必须安排一条程序转移类指令（程序控制类指令的一种）。

【经典总结】

程序控制类指令就是专用来控制程序执行顺序的，包括无条件转移、条件转移、循环控制及中断控制四种类型。

参考答案：D

2. (原书 第 2 题) 扩展操作码技术是 ()。
- A. 操作码字段以外的辅助操作字段的代码
 - B. 指令格式中不同字段设置的操作码
 - C. 一种指令优化技术，加快指令的译码和执行

- D. 一种指令优化技术，即让操作码的长度随地址数的减少而增加，不同地址数的指令可以具有不同的操作码长度

【解析】 本题考查扩展操作码技术的概念和特点。

通常一条指令由操作码和地址码组成，操作码指出该指令应该执行什么性质的操作和具有何种功能，地址码指出操作数所在的寄存器地址或存储器的地址。

根据指令中操作数地址码的数目的不同，可以将指令分为零地址指令、一地址指令、二地址指令、三地址指令和多地址指令等多种指令格式。

【经典总结】

通常在指令字中用一个固定长度的字段来表示基本操作码。而对于一部分不需要某个地址码的指令，可将它们的操作码扩充到该地址字段，这样既可以充分利用指令字的各个字段，又能在不增加指令长度的情况下扩展操作码的长度，使它能表示更多的指令。

可见，D 答案的描述更好。

参考答案: D

3. (原书第5题) 零地址运算指令在指令格式中不给出操作数地址，它的操作数来自()。
- A. 立即数和栈顶 B. 暂存器
- C. 栈顶和次栈顶 D. 累加器

- A. 立即数和栈顶 B. 暂存器
C. 栈顶和次栈顶 D. 累加器

- C. 栈顶和次栈顶 D. 累加器

【解析】 本题考查零地址指令的操作数来源。

零地址指令只给出了操作码，没有显地址码部分。这种指令通常有两种可能：

- (1). 不需要操作数,主要用来完成某种控制功能,如暂停指令(HLT)、空操作指令(NOP)、开中断指令(STI)等。

- (2). 零地址的算术逻辑类指令是用于堆栈计算机中的，堆栈计算机没有一般计算机中必备的通用寄存器，因此堆栈成为提供操作数和保存运算结果的唯一场所。通常，参加算术逻辑运算的两个操作数隐含地从堆栈顶部弹出，送到运算器中进行运算，运算的结果再隐含地压入堆栈。

参考答案: C

4. (原书 第 11 题) 存放欲执行指令的寄存器是 ()。
- A. MAR B. PC C. MDR D. IR

- A. MAR B. PC C. MDR D. IR

【解析】 本题考查指令寄存器的作用。

我们对 PC、IR、MDR 和 MAR 四个寄存器的作用，做一个简单的总结：

- (1). 程序计数器 PC, 存放下一条要执行的指令地址;
- (2). 指令寄存器 IR, 存放当前正在执行的指令 (非指令地址);
- (3). 存储器数据寄存器 MDR, 存放写入或读出的数据/指令;
- (4). 存储器地址寄存器 MAR, 存放写入或读出的数据/指令的地址。

参考答案: D

5. (原书 第 15 题) 在二地址指令中 () 是正确的。
- A. 指令的地址码字段存放的一定是操作数
 - B. 指令的地址码字段存放的一定是操作数地址
 - C. 运算结果通常存放在其中一个地址码所提供的地址中
 - D. 指令的地址码字段存放的一定是操作码

【解析】本题考查 2 地址指令格式的特点。

通常算术逻辑运算指令需要两个操作。二地址指令给出了源操作数和目的操作数的地址, 并规定将操作结果放回目的操作数的地址位置。通常也将二地址指令叫做双操作数指令。

参考答案: C

二. 综合应用题部分

1. (原书 第 3 题) 某机字长 16 位, 存储字长等于指令字长, 若存储器直接寻址空间为 128 字, 变址时的位移量为 -64~+63, 16 个通用寄存器可作为变址寄存器。设计一套指令格式, 满足下列寻址类型的要求。
- (1). 直接寻址的二地址指令 3 条;
 - (2). 变址寻址的一地址指令 6 条;
 - (3). 寄存器寻址的二地址指令 9 条;
 - (4). 直接寻址的一地址指令 13 条。

【解析】

(1). 因为存储字长等于指令字长, 而且机器字长为 16 位, 可取指令字长为 16 位。对于直接寻址的二地址指令, 因为指令的条数是 3 条, 需要操作码 3 个, 用两位 2 进制数表示。而指令的直接寻址范围是 128 字, 需要 7 位来表示。那么, 直接寻址的 2 地址指令格式如图 4.1 所示。

OP(2)	Address_1(7)	Address_2(7)
-------	--------------	--------------

图 4.1 直接寻址的 2 地址指令格式

其中，OP 字段可取 00、01 和 10 来表示三条指令的操作码字段。

(2). 变址寻址的位移量为-64~+63，需要 7 位 2 进制数来表示这个范围。对于 1 地址指令，若是寄存器变址寻址，16 个寄存器需要 4 位 2 进制数来表示。故而，指令格式可设计如图 4.2 所示。

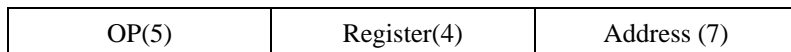


图 4.2 变址寻址指令格式

其中，OP 字段可取 11000~11101 来表示该 6 条变址寻址指令的操作码字段。

(3). 因为共有 16 个寄存器，寄存器寻址的二地址指令有两个寄存器地址字段，每个字段用 4 位 2 进制数表示。该寄存器寻址的二地址指令格式如图 4.3 所示。

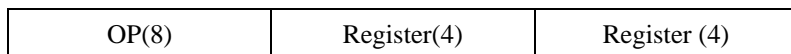


图 4.3 寄存器寻址的二地址指令格式

其中，OP 字段可取 11110000~11111000 来表示该 9 条寄存器寻址指令。

(4). 直接寻址的一地址指令，需要 7 位来表示地址，故而可以采用如图 4.4 所示的指令格式。

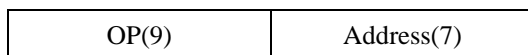


图 4.4 直接寻址的一地址指令格式

其中，可采用 1 1111 0001~1 1111 1101 来表示该 13 条直接寻址指令的 OP 字段。

2. **（原书 第 13 题）**某机字长 16 位，存储器直接寻址空间为 128 字，变址时的位移量为-64~+63，16 个通用寄存器均可作为变址寄存器。设计一套指令系统格式，满足下列寻址类型的要求：

- (1). 直接寻址的二地址指令 3 条；
- (2). 变址寻址的一地址指令 6 条；
- (3). 寄存器寻址的二地址指令 8 条；
- (4). 直接寻址的一地址指令 12 条；
- (5). 零地址指令 32 条。

【解析】

(1). 根据题意，直接寻址的每一个地址都应该能访问到 128 字的空间，即 7 位。对于直接寻址的二地址指令，需要 7 位的地址字段两个字段。因为机器字长是 16 位，剩下的 2

位操作码可以用 00、01、10 来表示直接寻址的 3 条指令，11 用来作为下一种格式指令的操作码扩展使用。如图 4.5 所示。

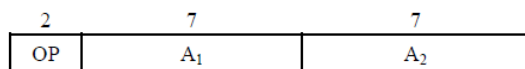


图 4.5 直接寻址的二地址指令格式

(2). 指令可以直接寻址 128 个字，故而变址寻址指令的地址段位移量范围是 $-64 \sim +63$ ，形式地址字段 A 取 7 位。另外，16 个寄存器也应该能访问到，这 16 个通用寄存器可作为变址寄存器，取 4 位用来表示变址寄存器 R_x 的编号。那么，只有 5 位可用作操作码了。

因为需要变址寻址的指令总共 6 条，可以将 11 000~11 101 这 6 个编码作为 6 条变址寻址指令的操作码，剩下的两个编码 11110 和 11111 作为扩展用。这 6 条变址寻址的一地址指令的格式如图 4.6 所示。

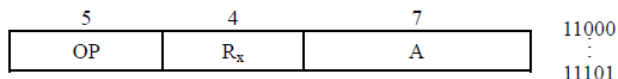


图 4.6 变址寻址的指令格式

(3). 寄存器寻址的 2 地址指令，分为需要两个地址字段，每个字段 4 位，用来表示 16 个寄存器的编号。剩下的 8 位可用作操作码。

显然，我们可以分配 11110 000~11110 111 给这 8 条指令，作为它们的操作码。剩下的 11111 000~11111 111 共 8 个编码作为扩展用。如图 4.7 所示。

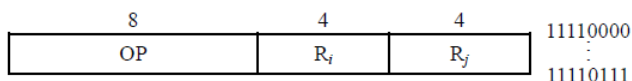


图 4.7 寄存器寻址的 2 地址指令格式

(4). 直接寻址的指令，其地址码字段每个字段应该为 7 位，显然当直接寻址的指令是一地址指令时，地址码字段是 7 位，操作码可以为 9 位。

显然，可以采用 11111 0000~11111 1011 作为 12 条 1 地址直接寻址指令的操作码，剩下的 11111100~11111111 作为扩展用。如图 4.8 所示。

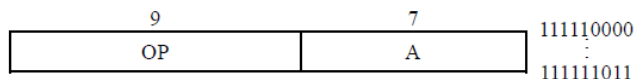


图 4.8 直接寻址的 1 地址指令格式

(5). 在零地址指令中，指令的 16 位都作为操作码。显然可以采用 1111 1110 0000 0000~1111 1110 0001 1111 来构成 32 条零地址指令的操作码，剩下的其他操作码暂时不用。

- A. 直接、立即、间接 B. 间接、直接、立即
C. 立即、直接、间接 D. 直接、间接、立即

【解析】本题考查直接、间接、立即寻址三种寻址方式的速度快慢比较。

立即寻址方式中,指令的操作数直接在指令的地址字段,无需访问内存,所以执行速度最快。直接寻址,指令中地址码字段给出的地址 A 就是操作数的有效地址,需要访存一次去取操作数,速度显然没有立即寻址快。间接寻址(如图 4.11 与图 4.12 所示),指令中给出的地址 A 不是操作数的地址,而是存放操作数地址的地址,所以,对于一次间址,需要访存两次,第一次访存是为了取操作数的有效地址,第二次访存是为了取操作数。显然,间址寻址的速度又不及直接寻址。

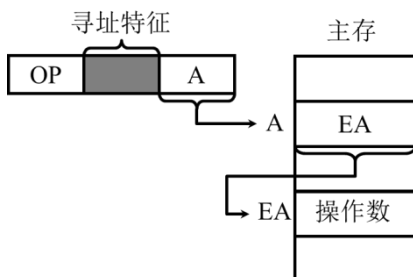


图 4.11 一次间址的地址转换过程

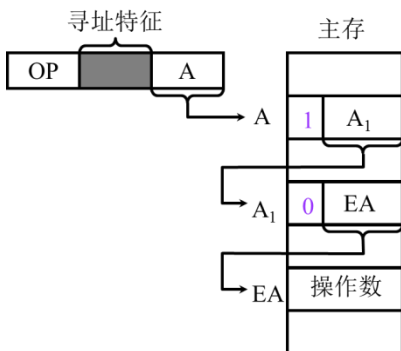


图 4.12 多次间址的地址转换过程

综上，立即寻址、直接寻址、间接寻址三者的速度依次变慢，本题选择 C 答案。

参考答案：C

3. (原书第6题) 采用变址寻址可扩大寻址范围, 且()。
- A. 变址寄存器内容由用户确定, 在程序执行过程中不可变
- B. 变址寄存器内容由操作系统确定, 在程序执行过程中不可变
- C. 变址寄存器内容由操作系统确定, 在程序执行过程中可变

D. 变址寄存器内容由用户确定，在程序执行过程中可变

【解析】本题考查变址寻址方式的特点。

基址寻址和变址寻址在形成有效地址时所用的算法是相同的。但这两种寻址方式应用的场合不同：

(1). 变址寻址是**面向用户**的，用于访问字符串、向量和数组等成批数据。变址寄存器是面向用户的，在指令的执行过程中，变址寄存器的内容可由用户改变。但是，形式地址不变；

(2). 基址寻址**面向系统**，主要用于逻辑地址和物理地址的变换，用以解决程序在主存中的再定位和扩大寻址空间等问题。基址寄存器是面向操作系统的，其内容只能由操作系统或管理程序来管理，用户指令无权操作和修改。在指令的执行过程中，基址寄存器内容不变。但是，形式地址可变。

参考答案：D

4. (原书第10题) 堆栈寻址方式中，设A为累加寄存器，SP为堆栈指示器，M_{sp}为SP指示器的栈顶单元，如果操作的动作是： $(A) \rightarrow M_{sp}$ ， $(SP) - 1 \rightarrow SP$ ，那么出栈操作的动作为：()。
- A. $(M_{sp}) \rightarrow A$ ， $(SP) + 1 \rightarrow SP$
- B. $(SP) + 1 \rightarrow SP$ ， $(M_{sp}) \rightarrow A$
- C. $(SP) - 1 \rightarrow SP$ ， $(M_{sp}) \rightarrow A$
- D. $(M_{sp}) \rightarrow A$ ， $(SP) - 1 \rightarrow SP$

【解析】本题考查堆栈寻址的操作方法。

关于堆栈寻址，我们接触得并不多。其实，这里的“堆栈”，操作和我们在《数据结构》中接触到的堆栈的操作是类似的。 $(A) \rightarrow M_{sp}$ ，即将累加器A中的内容压入SP的栈顶M_{sp}，然后栈顶指针减1。

【经典问题】

那么，为什么是减1呢？

从主存中划出一段区域来作堆栈(软堆栈)。堆栈的大小可变，栈底固定，栈顶浮动，故需要一个专门的硬件寄存器作为堆栈栈顶指针SP，简称栈指针。栈指针所指定的主存单元，就是堆栈的栈顶。

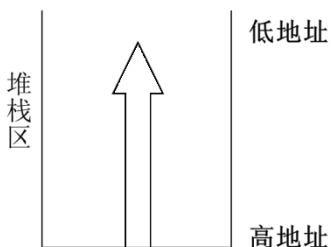


图 4.13 计算机内的堆栈结构

如图 4.13 所示，一般计算机系统中堆栈的栈底地址大于栈顶地址，通常栈指针始终指向栈顶的单元。进栈时，SP 的内容需要先自动减 1，然后再将数据压入堆栈（亦可反过来，看系统的情况）。本题中，采用的是先将数据压入栈顶，然后栈顶指针减 1，说明栈顶指针指向当前栈顶的下一个位置。所以，数据出栈时，也应该先将栈顶指针 SP 自加 1，使该指针指向栈顶元素，再让栈顶元素（M_{sp}）出栈。

故而，选择 B 答案。

【思考】请同学们思考一个问题，若本题的语句 (A) → M_{sp} 与 (SP) - 1 → SP 的位置对换，那么是不是仍然选择 B 答案呢？

参考答案：B

5. (原书 第 16 题) 基址寻址方式中，操作数的有效地址是 ()。

- A. 基址寄存器内容加上形式地址（位移量）
- B. 程序计数器内容加上形式地址
- C. 变址寄存器内容加上形式地址
- D. 寄存器内容加上形式地址

【解析】本题考查基址寻址方式下操作数有效地址的计算方法。

基址寻址，将基址寄存器 R_b 的内容与位移量 D 相加，形成操作数有效地址：

$$EA = (R_b) + D$$

基址寄存器的内容称为基址值，指令的地址码字段是一个位移量，位移量可正可负。如图 4.14 所示。

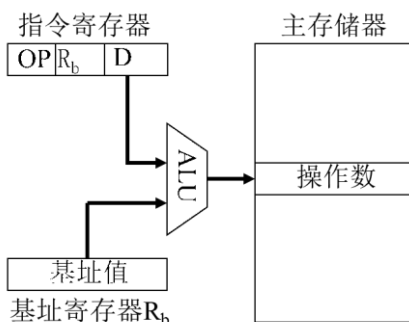


图 4.14 基址寻址的地址转换过程

参考答案：A

6. (原书 第 26 题) 【2013 年 408 统考】假设变址寄存器 R 的内容 1000H，指令中的形式地址为 2000H，地址 1000H 中的内容为 2000H，地址 2000H 中的内容为 3000H，地址 3000H 中的内容为 4000H，则变址寻址方式下访问到的操作数是 ()。
- A. 1000H B. 2000H C. 3000H D. 4000H

【解析】变址寻址方式下，偏移量在指令的地址码字段 A，基准的寄存器是变址寄存器。变址寄存器的内容与形式地址的内容相加之后，得到操作数的实际地址，根据实际地址访问内存。

根据本题的含义，变址寄存器 R 的内容为 1000H，指令的形式地址是 2000H，所以操作数而当有效地址是 3000H。地址 3000H 中的内容是 4000H（如图 4.15 所示）。

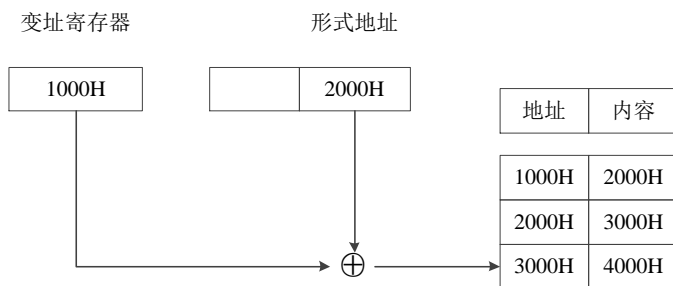


图 4.15 变址寻址的有效地址计算过程

参考答案：D

二. 综合应用题部分

1. (原书 第 1 题) 基址寄存器的内容为 2000H，变址寄存器的内容为 03A0H,指令地址

码部分是 3FH，当前正在执行的指令所在地址为 2B00H，请求出：

- (1). 变址编址（考虑基址）访存有效地址？
- (2). 相对编址访存有效地址？

【解析】本题考查基址寻址和变址寻址的有效地址计算方法。

(1). 变址寻址方式的有效地址为变址寄存器的内容加上指令中的形式地址字段的值，故而，变址编址（考虑基址）访存有效地址=2000H+03A0H+3FH=23DFH。

(2). 相对寻址的有效地址为当前 PC 的内容与形式地址的和。本题中，相对编址访存有效地址=2B00H+3FH=2B3FH。

2. (原书 第 5 题) 基址寄存器的内容为 3000H，变址寄存器的内容为 02B0H，指令的地址码为 002BH，程序计数器（存放当前正在执行的指令的地址）的内容为 4500H，且存储器内存放的内容如表 4.1 所示。

表 4.1 存储器的地址和相应的存放内容

地址	002BH	302BH	32B0H	32DBH	3500H	452BH
内容	3500H	3500H	5600H	2800H	2600H	2500H

- (1). 若采用基址寻址方式，则取出的操作数是什么？
- (2). 若采用变址寻址（考虑基址）方式，取出的操作数是什么？
- (3). 若采用立即寻址方式，取出的操作数是什么？
- (4). 若采用存储器间接寻址（不考虑基址）方式，取出的操作数是什么？
- (5). 若相对寻址用于转移指令，则转移地址是多少？

【解析】本题考查常见的寻址方式的有效地址计算方法。

(1). 采用基址寻址方式，有效地址 EA 等于基址寄存器的内容加上形式地址。本题中，基址寄存器的内容为 3000H，指令的地址码为 002BH，所以操作数的有效地址为 302BH。查表可知，地址 302BH 的内容是 3500H，即操作数是 3500H。

(2). 变址寻址考虑基址的情况下，如同综合题第 1 题（1）。计算有效地址 EA=基址寄存器的内容+变址寄存器的内容+偏移量=3000H+02B0H+002BH=32DBH。查表可知，该存储单元的内容（即操作数）是 2800H。

(3). 采用立即寻址方式，指令的形式地址字段的值，就是操作数本身，所以操作数为 002BH。

(4). 采用寄存器间接寻址方式, 指令的地址码字段给出的是操作数的有效地址, 而不是操作数本身。查表可知, 操作数的地址为 3500H, 存放在该物理位置的操作数为 2600H。采用相对寻址的方式, 指令的有效地址为 PC 的值加上形式地址, 故而 $EA=4500H+002BH=452BH$ 。

3. **(原书 第9题)【2010 年 408 统考】**某计算机字长为 16 位, 主存地址空间大小为 128KB, 按字编址, 采用字长指令格式, 指令各字段定义如表 4.2 所思。

表 4.2 指令各字段定义

15	12	11	6	5	0
OP	Ms	Rs	Md	Rd	
源操作数			目的操作数		

转移指令采用相对寻址方式, 相对偏移是用补码表示, 寻址方式定义如表 4.3 所示。

表 4.3 寻址方式定义

Ms/Md	寻址方式	助记符	含义
000B	寄存器直接	Rn	操作数= (Rn)
001B	寄存器间接	(Rn)	操作数= ((Rn))
010B	寄存器间接、自增	(Rn) +	操作数= ((Rn)), (Rn) + 1 → Rn
011B	相对	D (Rn)	转移目标地址= (PC) + (Rn)

注:

(X) 表示有储蓄地址 X 或寄存器 X 的内容, 请回答下列问题:

- (1). 该指令系统最多可有多少条指令? 该计算机最多有多少个通用寄存器? 存储器地址寄存器 (MDR) 和存储器数据寄存器 (MDR) 至少各需多少位?
- (2). 转移指令的目标地址范围是多少?
- (3). 若操作码 0010B 表示加法操作 (助记符为 add), 寄存器 R4 和 R5 的编号分别为 100B 和 101B, R4 的内容为 1234 H, R5 的内容为 5678 H, 地址 1234 H 中的内容为 5678 H, 地址 5678H 中的内容为 1234 H, 则汇编语言为 add (R4), (R5) + (逗号前原操作数, 逗号后为目的操作数) 对应的机器码是什么 (用十六进制表示)? 该指令执行后, 哪些寄存器和存储单元的内容会改变? 改变后的内容是什么?

【解析】

- (1). 操作码占 4 位, 则该指令系统最多可有 $2^4=16$ 条指令; 操作数占 6 位, 寻址方式

占 3 位, 于是寄存器编号占 3 位, 则该机最多有 $2^3=8$ 个通用寄存器; 主存容量 128KB, 按字编址, 计算机字长为 16 位, 划分为 $128\text{KB}/2\text{B}=2^{16}$ 个存储单元, 故 MDR 和 MAR 至少各需 16 位。

(2). PC 和 Rn 可表示的地址范围均为 $0\sim 2^{16}-1$, 而主存地址空间为 2^{16} , 故转移指令的目标地址范围是 $0000\text{H}\sim\text{FFFFH}$ ($0\sim 2^{16}-1$)。

(3). $\text{add}(\text{R4}), (\text{R5})$ 对应的机器码为 0010 001 100 010 101。执行后 R5 和 5678H 的内容会改变, 从 5678H 变为 5679H。存储器 5678H 的内容 $=5678\text{H}+1234\text{H}=68\text{ACH}$ 。

考点 3 CISC 和 RISC 的基本概念

温馨提示: 本考点考查 CISC 和 RISC 的基本概念, 请同学们注意区别这两种指令集系统。

一. 选择题部分

1. (原书 第 1 题)【2009 年 408 统考】下列关于 RISC 的叙述中, 错误的是 ()。

- A. RISC 普遍采用微程序控制器
- B. RISC 大多数指令在一个时钟周期内完
- C. RISC 的内部通用寄存器数量相对 CISC 多
- D. RISC 的指令数、寻址方式和指令格式种类相对 CISC 少

【解析】RISC 的基本特征: (1) 精简指令数量; (2) 简化指令格式; (3) 采用单周期指令; (4) 采用寄存器操作; (5) 硬线控制逻辑; (6) 优化编译程序。

显然, A 答案错误。

参考答案: A

2. (原书 第 4 题) 下列关于 RISC 的描述中, 不正确的是 ()。

- A. 指令条数比 CISC 少
- B. 指令长度固定, 指令格式种类少, 寻址方式种类少
- C. 在程序中出现频率占 80% 的指令占指令总数的 20%
- D. 只有取数/存数指令访问存储器

【解析】较之与 CISC，RISC 的指令系统比较简单，指令条数也比较少。而且，RISC 的指令长度固定，指令格式种类少，寻址方式种类也少。当然，RISC 机器只有 Load 和 Store 指令能访问内存。

CISC 指令系统中，指令丰富、功能强。这些计算机被称为复杂指令系统计算机，简称 CISC。

大量测试表明，CISC 最常使用的是一些比较简单的指令，这类指令仅占指令总数的 20%，但在各种程序中出现的频度却占 80%，其余大多数指令是功能复杂的指令，这类指令占指令总数的 80%，但使用频度很低，仅占 20%。因此，人们把这种情况称为“20%-80% 定律”。

但是，这个定律不是针对 RISC 的，而是针对 CISC 的。

参考答案：C

3. (原书 第 5 题) 下面关于 RISC 技术的描述中，正确的是 ()。
- A. 采用 RISC 技术后，计算机的体系结构又恢复到早期的比较简单的情况
 - B. 为了实现兼容性设计的 RISC 是从原来的 CISC 系统的指令系统中挑选一部分实现的
 - C. RISC 的主要目标是减少指令数
 - D. RISC 设有乘法指令和浮点运算指令

【解析】采用 RISC 技术之后，计算机的指令变简单了，但是这是为了并行处理，提高系统指令执行速度和系统效率，并没有使计算机的体系结构恢复到早期的比较简单的情况。故而，A 错误。

RISC 的指令系统并不是都来源于 CISC，也不是从 CISC 中抽取比较简单的指令。二者没有必然的联系。

RISC 中没有复杂的指令，乘法指令和浮点数运算指令属于复杂指令，运算速度慢，所以 RISC 一般不引入该类指令。

参考答案：C

二. 综合应用题部分

1. (原书 第 3 题) 列表比较 CISC 处理机和 RISC 处理机的特点。

【解析】表 4.4 是 CISC 处理机和 RISC 处理机的特点比较表。

表 4.4 CISC 和 RISC 的特点比较

比较内容	CISC	RISC
指令系统	复杂、庞大	简单、精简
指令数目	一般大于 200	一般小于 100
指令格式	一般大于 4	一般小于 4
寻址方式	一般大于 4	一般小于 4
指令字长	不固定	等长
可访存指令	不加限定	只有 LOAD/STORE 指令
各种指令使用频率	相差很大	相差不大
各种指令执行时间	相差很大	绝大多数在一个周期内完成
优化编译实现	很难	较容易
程序源代码长度	较短	较长
控制器实现方式	绝大多数为微程序控制	绝大部分为硬布线控制
软件系统开发时间	较短	较长

本章到此就结束了，请问您有什么疑问吗？任何问题，欢迎您与我们作者进行交流！



梦享团队微信号



梦享团队官方微信公众号



梦享团队新浪微博

第五章 中央处理器（CPU）

考点 1 CPU 的功能和基本结构

温馨提示：本考点主要考查 CPU 的结构和功能，请同学们掌握 CPU 内部结构和相关部件的功能。

一. 选择题部分

1. (原书第2题) 在 CPU 的寄存器中, () 对用户是完全透明的。
- A. 程序计数器 B. 状态寄存器
- C. 指令寄存器 D. 通用寄存器

【解析】 本题考查对用户透明的寄存器。

作为高速存储单元，微处理器内部有多种寄存器，用于暂时存放程序执行过程中的代码和指令。有些寄存器对应用人员来说是不可见的，不能直接控制。例如，保存指令代码的指令寄存器。所以，它们被称为透明寄存器。这里的“透明”，表示实际存在，但从某个角度看好像没有。

参考答案: C

2. (原书第6题) 指令寄存器的位数取决于()。
- A. 存储器的容量 B. 指令字长
- C. 机器字长 D. 存储字长

【解析】本题考查决定指令寄存器的位数的因素。

指令寄存器用来存放从存储器中取出的指令。当指令从主存取出暂存于指令寄存器之后，在执行指令的过程中，指令寄存器的内容不允许发生变化，以保证实现指令的全部功能。

指令字长是指机器指令中二进制代码的总位数。指令字长取决于操作码的长度、操作数地址的长度和操作数地址的个数。不同的指令的字长是不同的。指令寄存器的位数，应该与指令字长相等，这样从内存中读到的 2 进制数是一条完整的指令。

参考答案: B

3. (原书第9题) 程序计数器 PC 用来存放指令地址，每当执行完一条指令后，通常由程序计数器提供后继指令地址，其位数和()位数相同。
- A. 指令寄存器 IR B. 主存数据寄存器 MDR
- C. 程序状态寄存器 PSR D. 主存地址寄存器 MAR

【解析】 本题考查程序计数器功能和特点。

程序计数器又称**指令计数器**，用来存放即将要执行的指令地址。在取指时，将 PC 中存放的指令地址送到存储器地址寄存器 MAR，从内存中取出 MAR 对应的地址单元的内容送到指令寄存器 IR。所以，PC 的位数应该与存储器地址寄存器 MAR 的位数相同。

【经典错误】

注意，指令计数器不是指令寄存器，这二者不要混淆了！

参考答案: D

二. 综合应用题部分

1. (原书 第 2 题) CPU 有哪些基本功能?

【解析】CPU 对整个计算机系统的运行是及其重要的，它具有如下 4 个方面的基本功能：

(1). 指令控制

程序的顺序控制称为指令控制。由于程序是一个指令序列，这些指令的相互顺序不能任意颠倒，必须严格按照程序规定的顺序执行。

(2). 操作控制

一条指令的功能往往是由若干个操作信号的组合来实现的，因此 CPU 管理并产生由内存取出的每条指令的操作信号，把各种操作信号送往相应的部件，从而控制这些部件按指令的要求进行动作。

(3). 时间控制

对各种操作实施时间上的定时称为时间控制。在计算机中，各种指令的操作信号以及一条指令的整个执行过程都受到时间的严格定时。

(4). 数据加工

对数据进行算术运算和逻辑运算处理。

考点 2 指令执行过程

温馨提示：本考点考查指令的执行过程，请同学注意取指令、分析指令和执行指令等基本过程。

一. 选择题部分

1. (原书 第 1 题) 指令周期是指 ()。

- A. CPU 从主存取出一条指令的时间
- B. CPU 执行一条指令的时间
- C. CPU 周期时间
- D. CPU 取出指令和执行指令的时间

【解析】本题考查指令周期的基本概念。

指令周期是指从取指令、分析指令到执行完该指令所需的全部时间。由于各种指令的操作功能不同，有的简单，有的复杂，因此各种指令的指令周期不尽相同。

参考答案：D

2. (原书 第 4 题) 设机器字长 16 位，存储器按字节编址，CPU 读取一条单字长指令后，PC 值自动加 ()。

- A. 1
- B. 2
- C. 3
- D. 4

【解析】我们先来一个经典的小总结。

【经典总结】

我们一般都说，当一条指令的取指周期执行完毕之后，PC 的值会自动加 1。那么，这里的“加 1”是什么意思呢？其实，加 1 简单的理解，它表示的是当前执行的本条指令的长度。这样一来，PC 的值加 1，就是下一条指令的地址。

本题中，机器字长是 16 位，一般机器字长和指令字长相等，所以我们取指令字长为 16 位。按照字节编址，指令字长是两个字节。所以，CPU 取一条单字长指令之后，PC 的值应该自动加 2，那才是下一条指令的起始地址。

参考答案：B

3. (原书 第 7 题) 在间址周期中，()。

- A. 所有指令的间址操作都是相同的

- B. 凡是存储器间接寻址的指令，它们的操作都是相同的
- C. 对于存储器间接寻址或寄存器间接寻址的指令，它们的操作是不同的
- D. 以上都不对

【解析】本题考查间址周期。

间址周期是为了完成取操作数有效地址的任务，具体操作如下：

- (1). 将指令的地址码部分（形式地址）送至存储器地址寄存器，记作(MDR)→MAR；
- (2). 向主存发读命令，启动主存读操作，记作 Read；
- (3). 将 MAR 所指的主存单元中的内容（有效地址）经数据总线读至 MDR，记作 M(MAR)→MDR；
- (4). 将有效地址送至存储器地址寄存器 MAR，记作(MDR)→MAR。此操作在有些机器中可省略。

不是所有指令的间址操作都是相同的，比如对存储器间址寻址和寄存器间址寻址的指令，它们的操作就不相同。也不是凡是存储器间址寻址的指令的间址操作都是相同的，比如对存储器的一次间址和多次间址寻址，其间址操作就不同。所以，C 答案正确。

参考答案：C

4. (原书 第 10 题) 以下叙述中 () 是错误的。
- A. 取指令操作是控制器固有的功能，不需要在操作码控制下完成
 - B. 所有指令的取指令操作都是相同的
 - C. 在指令长度相同的情况下，所有指令的取指操作都是相同的
 - D. 一条指令包含取指、分析、执行三个阶段

【解析】不是所有指令的取指操作都相同的，只有指令长度相同的情况下，取指操作才相同。如，一条单字长指令和一条双字长指令的取指操作就不同。

参考答案：B

二. 综合应用题部分

1. (原书 第 2 题) 什么是指令周期？什么是机器周期？什么是时钟周期？三者之间的关系如何？

【解析】指令周期是完成一条指令所需的时间。包括取指令、分析指令和执行指令所需的全部时间。机器周期也称为 CPU 周期，是指被确定为指令执行过程中的归一化基准时

间,通常等于取指时间(或访存时间)。时钟周期是时钟频率的倒数,也可称为节拍脉冲或 T 周期,是处理操作的最基本单位。一个指令周期由若干个机器周期组成,每个机器周期又由若干个时钟周期组成。

2. (原书 第 4 题) 设机器 A 的主频为 8MHz, 机器周期含 4 个时钟周期, 且该机的平均指令执行速度是 0.4MIPS, 试求该机的平均指令周期和机器周期。每个指令周期包含几个机器周期? 如果机器 B 的主频为 12MHz, 且机器周期也含 4 个时钟周期, 试问 B 机的平均指令执行速度为多少 MIPS?

【解析】根据机器 A 的主频为 8MHz, 得时钟周期为 $1/8\text{MHz}=0.125\mu\text{s}$

- (1). 机器周期= $0.125\mu\text{s}\times 4 = 0.5\mu\text{s}$
- (2). 平均指令执行时间为 $1/0.4=2.5\mu\text{s}$
- (3). 每个指令周期含 $2.5/0.5=5$ 个机器周期
- (4). 在机器周期所含时间周期数相同的前提下, 两机平均指令执行速度与他们的主频有关, 即

$$\frac{A\text{机的平均指令速度}}{B\text{机的平均指令速度}} = \frac{A\text{机主频}}{B\text{机主频}}$$

$$\text{则 B 机的平均指令执行速度} = \frac{0.4\text{MIPS}\times 12\text{MHz}}{8\text{MHz}} = 0.6\text{MIPS}$$

考点 3 数据通路的功能和基本结构

温馨提示: 数据通路的功能和工作原理部分, 在历年的《计算机组成原理》考研中, 常以大题方式跟大家见面, 属于核心知识点, 也属于难点。命题的方式, 都是给大家一个数据通路图, 让同学们分析指令执行过程中的数据通路、控制信号和使用部件。本考点很难掌握, 其实出题的方式如出一辙, 请同学们根据我们给出的例题, 掌握解该类题型的精髓。

一. 选择题部分

1. (原书 第 1 题) CPU 中的译码器主要用于 ()。
- | | |
|----------------|---------|
| A. 地址译码 | B. 指令译码 |
| C. 选择多路数据至 ALU | D. 数据译码 |

【解析】译码器是一种具有“翻译”功能的逻辑电路，这种电路能将输入二进制代码的各种状态，按照其原意翻译成对应的输出信号。

指令由操作码和操作数组成。操作码表示要执行的操作性质，即执行什么操作，或做什么；操作数是操作码执行时的操作对象，即对什么数进行操作。计算机执行一条指定的指令时，必须首先分析这条指令的操作码是什么，以决定操作的性质和方法，然后才能控制计算机其他各部件协同完成指令表达的功能。这个分析工作由译码器来完成。

参考答案：B

二. 综合应用题部分

1. (原书 第 1 题) 某计算机有如图 5.1 所示的功能部件，其中 M 为主存，指令和数据均存放在其中，MDR 为主存数据寄存器，MAR 为主存地址寄存器， $R_0 \sim R_3$ 为通用寄存器，IR 为指令寄存器，PC 为程序计数器（具有自动加 1 功能），C、D 为暂存寄存器，ALU 为算术逻辑单元，移位器可左移、右移、直通传送。

(1). 将所有功能部件连接起来，组成完整的数据通路，并用单向或双向箭头表示信息传送方向。

(2). 画出“ADD R_1 , (R_2)”指令周期流程图。该指令的含义是将 R_1 中的数与 (R_2) 指示的主存单元中的数相加，相加的结果直通传送到 R_1 中。

(3). 画出“ADD R_1 , R_2 ”指令周期流程图。该指令的含义是将 R_1 中的数与 R_2 中的数相加，相加的结果直通传送到 R_1 中。

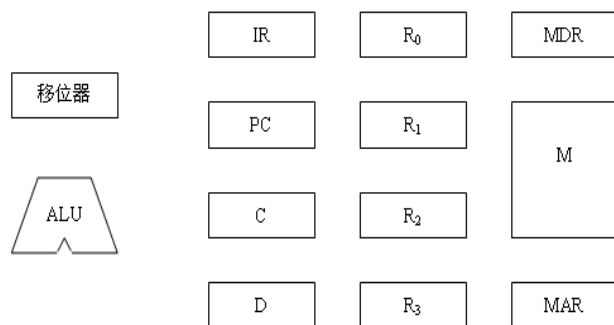


图 5.1 某机器的功能部件

【解析】

(1). 依题意，M 为主存，数据经过 MDR 寄存器可从主存中读出，亦可写入，所以该

位置的数据可以双向流动。寄存器 R0~R3 的数据也可以双向流动。

IR 寄存器在指令阶段结束后，获得当前执行的指令内容，但是也要取出来译码，所以数据是双向的。PC 寄存器的内容能被写入，这个我们都知道，但是 PC 的内容也常被读出，这个很多同学可能转不过弯来。比如我们常写的“PC→MAR”就是一个读 PC 寄存器内容的操作了。

MAR 寄存器的数据应该是单向流动的，没有数据经过这个部件送往 CPU。C、D 这两个寄存器的数据应该也是双向的，但是在这两个寄存器的数据送往 ALU 以及 ALU 进行移位运算、运算结果经过数据线送往内存的过程，数据都是单向流动的。

综上，数据通路如图 5.2 所示。

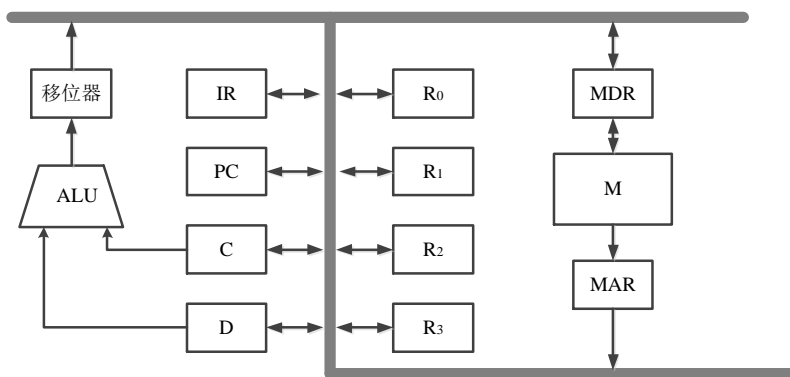


图 5.2 某机器的数据通路连接图

(2). 此指令为 RS 型指令，一个操作数在 R1 中，另一个操作数在 R2 为地址的内存单元中，相加结果放在 R1 中。

图 5.3 中的左图，即是该指令的周期流程图。其中，①~③为取指令，④~⑥为取源操作数，⑦为取目标操作数，⑧为 C 和 D 两个操作数相加，⑨将运算结果存回寄存器 R1。

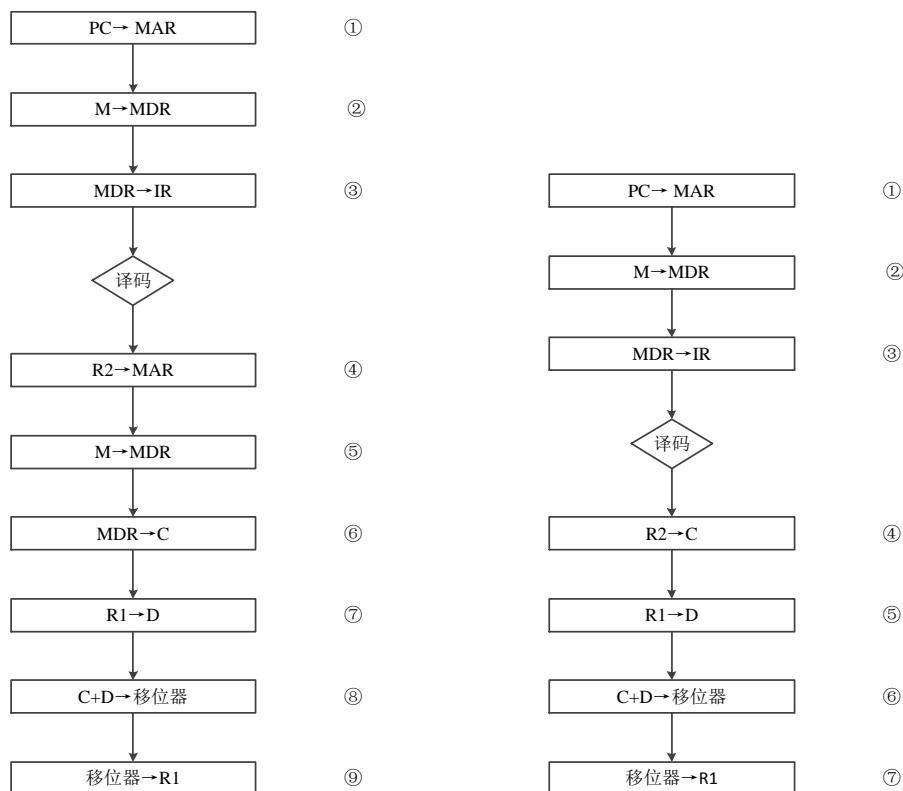


图 5.3 该指令的周期流程图

(3). 此指令为 RR 型指令，两个操作数都在寄存器中，两个寄存器相加的结果放在寄存器 R1 中。

图 5.3 中的右图，即是此指令的周期流程图。其中，①~③为取指令，④为取源操作数，⑤为取目的的操作数，⑥为 C 和 D 两个操作数相加，⑦将运算结果存回寄存器 R1。

2. **(原书 第 5 题)**【2009 年 408 统考】某计算机字长 16 位，采用 16 位定长指令字结构，部分数据通路结构如图 5.4 所示，图中所有控制信号为 1 时表示有效、为 0 时表示无效，例如控制信号 MDRinE 为 1 表示允许数据从 DB 打入 MDR，MDRin 为 1 表示允许数据从内总线打入 MDR。假设 MAR 的输出一直处于使能状态。加法指令“ADD (R1), R0”的功能为 $(R0) + ((R1)) \rightarrow (R1)$ ，即将 R0 中的数据与 R1 的内容所指主存单元的数据相加，并将结果送入 R1 的内容所指主存单元中保存。

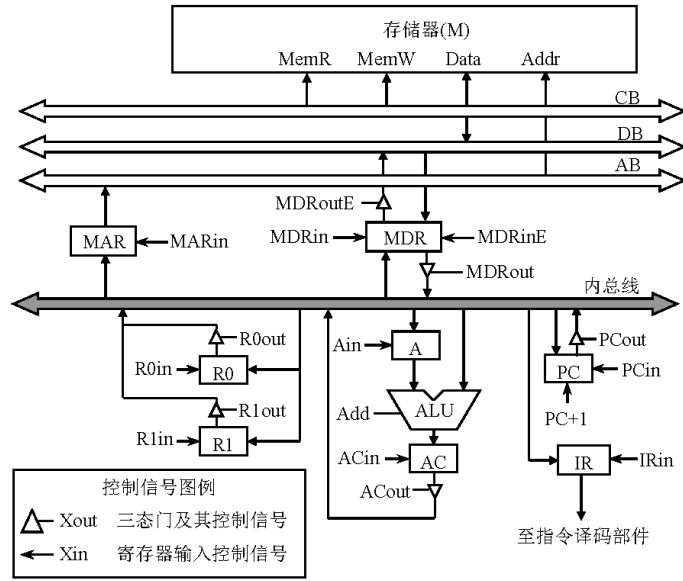


图 5.4 机器的部分数据通路结构

表 5.1 给出了上述指令取指和译码阶段每个节拍（时钟周期）的功能和有效控制信号，请按表中描述方式用表格列出指令执行阶段每个节拍的功能和有效控制信号。

表 5.1

时钟	功能	有效控制信号
C1	$MAR \leftarrow (PC)$	PCout, MARin
C2	$MDR \leftarrow M(MAR)$ $PC \leftarrow (PC) + 1$	MemR, MDRinE PC+1
C3	$IR \leftarrow (MDR)$	MDRout, IRin
C4	指令译码	无

【解析】本题是 2009 年全国 408 统考真题，综合性比较强。下面，我们来详细解析此题。

一条指令的执行过程通常由取指、译码和执行 3 个步骤完成，本题中给出的表格表明了取指阶段和译码阶段的执行过程，该过程在取指阶段安排了 3 个节拍、译码用 1 个节拍。执行加法运算并把结果写入主存如何完成呢？我们需要划分执行步骤、确定完成的功能、要提供的控制信号，这是本题的要测试的内容。为回答这个问题，首先需要看清图中给出的部件组成情况和信息传送的路径。

本题图比较复杂，我们把复杂的问题简单分析。

【经典总结】

怎么简单分析呢？

先不考虑需要提供的控制信号，仅安排执行步骤和每一个步骤具体的操作，然后再安排每一个具体操作应该有的控制信号。这样一来，本题又回到了简单的数据通路题型的情况。

下面，我们来根据一般数据通路来安排“ADD (R1), R0”指令执行步骤。

(1). $MAR \leftarrow (R1)$

将寄存器 R1 的内容送往 MAR。(相应地，有效信号为 R1 寄存器的 R1out 和 MAR 的 MARin)

(2). $MDR \leftarrow M(MAR)$

存储器地址寄存器 MAR 中的地址对应的内存单元中的内容被送往存储器数据寄存器 MDR。(相应地，有效控制信号 MemR 和 MDRinE)

(3). $A \leftarrow (MDR)$

将存储器数据寄存器 MDR 的内容送往寄存器 A。因为运算器部件 ALU 的两个输入端只有一个与寄存器连接，所以必须有一个数先存放到寄存器 A 中。显然，另一个操作数已经放在寄存器 R0 中，可以直接读进来进行加法运算。因此寄存器 A 可以存放 MDR 送过来的数据。(相应地，有效信号为 MDRout, Ain)

(4). $AC \leftarrow (A) + (R0)$

两个操作数一个被送到寄存器 A 中，一个被送到寄存器 R0 中。可将这两个寄存器的内容送入运算器 ALU 进行运算，运算结果送往 AC 寄存器中。(相应地，有效控制信号为 R0out、Add 和 ACin)

(5). $MDR \leftarrow (AC)$

AC 寄存器存放的运算结果需要送到寄存器 R1 的内容所指定的内存单元中，所以先将 AC 寄存器的内容送往存储器数据寄存器 MDR 中。(相应地，有效控制信号为 ACout, MDRin)

(6). $M(MAR) \leftarrow MDR$

实际上，MAR 中仍然存放着寄存器 R1 的内容，所以 R1 所指定的内存单元也即 MAR 内容所对应的内存单元。CPU 发出写命令，将存储器数据寄存器 MDR 的内容送往存储器地址寄存器 MAR 的内容对应的内存单元中。(相应地，有效控制信号为 MDRoutE, MemW)

控制信号的安排，主要看数据的流向。我们把控制信号表述在以上 6 个步骤的括号内部分，方便同学们阅读。这 6 个步骤的时钟、功能和有效信号分别如表 5.2 所示。

表 5.2 6 个步骤的时钟、功能和有效信号示意图

时钟	功能	有效控制信号
C5	$MAR \leftarrow (R1)$	R1out, MARin
C6	$MDR \leftarrow M(MAR)$	MemR, MDRinE
C7	$A \leftarrow (MDR)$	MDRout, Ain
C8	$AC \leftarrow (A) + (R0)$	R0out, Add, ACin
C9	$MDR \leftarrow (AC)$	ACout, MDRin
C10	$M(MAR) \leftarrow (MDR)$	MDRoutE, MemW

考点 4 控制器的功能和工作原理

温馨提示：控制器主要有微程序控制器和硬布线控制器两种，请同学们了解这两种控制器的特点，并加以区别。另外，在微程序控制器部分，我们还需要掌握微程序、微指令和微命令的概念、特点和区别，并了解微指令的编码方式和微地址的形成方式。

一. 选择题部分

1. (原书 第 1 题) 控制器的功能是 ()。
- A. 产生时序信号
 - B. 从主存取出一条指令
 - C. 完成指令操作码译码
 - D. 从主存取出指令，完成指令操作码译码，并产生有关的操作控制信号，以解释执行该指令

【解析】本题考查控制器的功能。

控制器的主要功能有：

- (1). 从主存中取出一条指令，并指出下一条指令在主存中的位置。
- (2). 对指令进行译码或测试，产生相应的操作控制信号，以便启动规定的动作。
- (3). 指挥并控制 CPU、主存和输入输出设备之间的数据流动方向。

运算器的主要功能有：

- (1). 执行所有的算术运算；
- (2). 执行所有的逻辑运算，并进行逻辑测试。

【经典总结】

一句话概括：控制器的功能是对数据通路实施控制，以便信息能够正确传输，从而完

成指令的执行。

参考答案：D

2. (原书 第3题) 在采用增量方式的微指令中, 下一条微指令的地址 ()。

- A. 在微指令计数器中
- B. 在微指令寄存器中
- C. 在程序计数器中
- D. 在本条微指令的顺序控制字段中

【解析】本题考查微指令后继指令地址的形成。

后继微指令地址的形成, 有两种方式:

(1). 增量方式

顺序执行时后继微地址就是现行微地址加上一个增量(通常为“1”); 转移或转子时, 由微指令的顺序控制字段产生转移微地址。因此, 在微程序控制器中应当有一个微程序计数器(uPC)。

增量方式的优点是简单, 易于掌握, 编制微程序容易, 每条机器指令所对应的一段微程序一般安排在 CM 的连续单元中; 其缺点是这种方式不能实现两路以上的并行微程序转移, 因而不利于提高微程序的执行速度。

(2). 断定方式

断定方式的后继微地址可由微程序设计者指定, 或者根据微指令所规定的测试结果直接决定后继微地址的全部或部分值。

参考答案：A

3. (原书 第6题) 微程序控制器中, 机器指令与微指令的关系是 ()。

- A. 每一条机器指令由一组微指令编成的微程序来解释执行
- B. 每一条机器指令由一条微指令来执行
- C. 一段机器指令组成的程序可由一条微指令来执行
- D. 一条微指令由若干条机器指令组成

【解析】本题考查微程序控制器中, 机器指令与微指令的关系。

采用微程序控制方式的控制器称为微程序控制器。所谓微程序控制方式是指微命令不是由组合逻辑电路产生的, 而是由微指令译码产生。一条机器指令往往分成几步执行, 将每一步操作所需的若干微命令以代码形式编写在一条微指令中, 若干条微指令组成一段微程序, 对应一条机器指令。

参考答案：A

4. (原书 第 8 题) 在组合逻辑控制器的组成结构中, 不包括 ()。

- A. 数据缓冲寄存器
- B. 指令操作码译码器
- C. 控制信号产生器、时序信号产生器
- D. 程序计数器 PC

【解析】本题考查组合逻辑控制器的组成结构。

组合逻辑控制器的组成结构如图 5.5 所示。指令寄存器、程序计数器、指令操作码译码器都在组合逻辑控制器中。

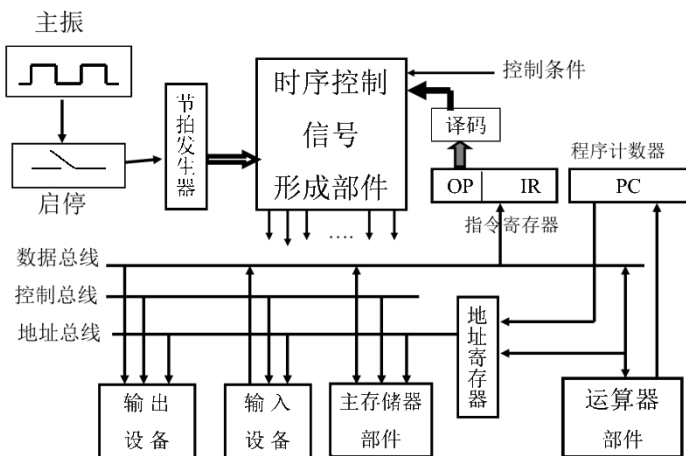


图 5.5 组合逻辑控制器的组成结构

数据缓冲寄存器用来暂时存放由内存储器读出的一条指令或一个数据字；反之，当向内存存入一条指令或一个数据字时，也暂时将它们存放在数据缓冲寄存器中。显然，缓冲寄存器并没有在控制器中。

缓冲寄存器的作用：

- (1). 作为 CPU 和内存、外部设备之间信息传送的中转站；
- (2). 补偿 CPU 和内存、外围设备之间在操作速度上的差别；
- (3). 在单累加器结构的运算器中，数据缓冲寄存器还可兼作为操作数寄存器。

参考答案：A

5. (原书 第 17 题) 水平型微指令的特点是 ()。

- A. 一次可以完成多个操作
- B. 微指令的操作控制字段不进行编码

- C. 微指令的格式简短
- D. 由水平型微指令解释指令得到的微程序比较长

【解析】本题考查水平型微指令的特点。

一次能定义并执行多个并行操作微命令的微指令,叫做水平型微指令。显然, A 答案正确。由水平型微指令解释指令的微程序,有微指令字较长而微程序短的特点。垂直型微指令则相反。所以, D 答案错误。

水平型微指令一指多用, **格式虽长**但并行性高,速度快,够灵活。垂直型微指令单指单用,长度虽短但效率低,并行性差,指令数多。

水平型微指令的操作控制字段也参与编码。按照控制字段的编码方法不同,水平型微指令又分为三种:全水平型(不译法)微指令,字段译码法水平型微指令,以及直接和译码相混合的水平型微指令。

参考答案: A

6. (原书 第 26 题) 水平型微指令和垂直型微指令相比, ()。
- A. 前者一次只能完成一个操作
 - B. 后者一次只能完成一个操作
 - C. 两者都一次只能完成一个操作
 - D. 两者都是一次完成多个操作

【解析】本题考查了水平型微指令和垂直型微指令一次能完成的操作数,可以归纳为这两种微指令的特点。

水平型微指令与垂直型微指令的比较如下:

(1). 水平型微指令并行操作能力强,效率高,灵活性强,垂直型微指令则较差。

在一条水平型微指令中,设置有控制信息传送通路(门)以及进行所有操作的微命令,因此在进行微程序设计时,可以同时定义比较多的并行操作的微命令,来控制尽可能多的并行信息传送,从而使水平型微指令具有效率高及灵活性强的优点。

在一条垂直型微指令中,一般只能完成一个操作,控制整个信息传送通路,因此微指令的并行操作能力低,效率低。

(2). 水平型微指令执行一条指令的时间短,垂直型微指令执行时间长。

因为水平型微指令的并行操作能力强,因此与垂直型微指令相比,可以用较少的微指令数来实现一条指令的功能,从而缩短了指令的执行时间。而且当执行一条微指令时,水平型微指令一般直接控制对象,而垂直型微指令要经过译码,会影响速度。

由水平型微指令解释指令的微程序,具有微指令字比较长,但微程序短的特点。垂直型微指令则相反,微指令字比较短而微程序长。

二. 综合应用题部分

1. (原书 第 2 题) 微程序控制器主要由哪几部分组成? 各部分的作用是什么?

【解析】主要由: 控制存储器, 微指令寄存器, 地址转移逻辑三部分组成。

(1). 控制存储器: 存放实现全部指令系统的微程序;

(2). 微指令寄存器: 存放由控制存储器读出的一条微指令信息;

地址转移逻辑: 承担自动修改微地址的任务。

2. (原书 第 7 题) 某 32 位机共有微操作控制信号 52 个, 构成 5 个相斥类的微命令组, 各组分别包含 4 个, 5 个, 8 个, 15 个, 20 个微命令, 已知可判定的外部条件有 CY 和 ZF 两个, 微指令字长 29 位。

(1). 给出采用断定方式的水平型微指令格式。

(2). 控制存储器器的容量应为多少位?

【解析】

(1). 水平型微指令格式如表 5.3 所示。

表 5.3 水平型微指令格式

D28~D26	D25~D23	D22~D19	D18~D15	D14~D10	D9 D8	D7~D0
4 个 微命令	5 个 微命令	8 个 微命令	15 个 微命令	20 个 微命令	条件测试字 段	下一地址字段
3 位	3 位	4 位	4 位	5 位	2 位	8 位

(2). 因为每个微指令的长度为 29 位, 所以控制存储器器的容量为: $2^8 \times 29 = 256 \times 29$ 位。

3. (原书 第 9 题) 已知单总线计算机结构如图 5.6 所示, 其中 M 为主存, XR 为变址寄存器, EAR 为有效地址寄存器, LATCH 为暂存器。假设指令地址已存于 PC 中, ADD X, D 指令为单字长指令, 其中: X 为变址寄存器 XR, D 为形式地址 (在流程图用 Ad(IR)表示)。寄存器的输入和输出均受控制信号控制, 如 PC_i 表示 PC 的输入控制信号, MDR_o 表示 MDR 的输出控制信号。凡是需要经过总线实现寄存器之间的传送, 需在流程图中注明, 如 $PC \rightarrow B \rightarrow MAR$ 相应的控制信号为 PC_o 和 MAR_i , 画出 ADD X, D 指令周期信息流程图, 并列出相应的微操作控制信号序列。

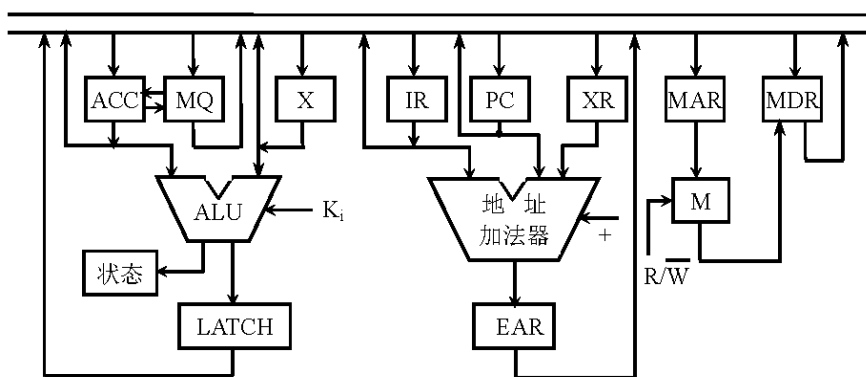


图 5.6 某单总线计算机结构

【解析】完成 ADD X, D 指令取指周期和执行周期的信息流程及相应的控制信号如图 5.7 所示。

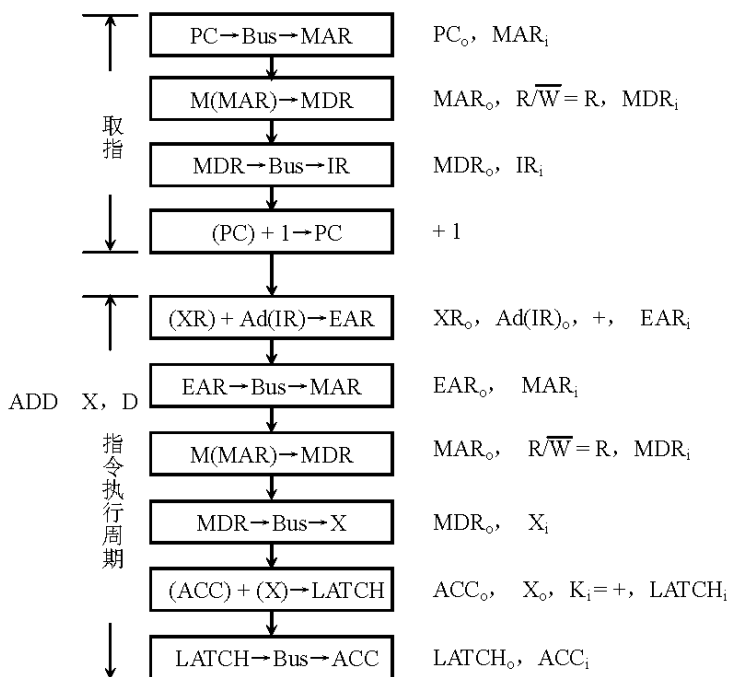


图 5.7 完成 ADD X, D 指令取指周期和执行周期的信息流程及相应的控制信号

考点 5 指令流水线

温馨提示：本考点主要考查指令流水线、超标量流水线和动态流水线的基本概念，请参加 408 统考的同学们掌握。部分自主命题高校可能在流水线部分出大题，请参考学校的考研真题，来决定本考点的复习层次。计算机组成原理部分，很多考点的某一个考查内容出题方式和解析方法很相像，同学们掌握了一个题目的解法，认真总结，可做到举一反三。

一. 选择题部分

1. (原书 第 1 题)【2009 年 408 统考】某计算机的指令流水线由四个功能段组成。指令流经各功能段的时间(忽略各功能段之间的缓存时间)分别是 90ns、80ns、70ns 和 60ns。则该计算机的 CPU 时钟周期至少是：()。

A. 90ns B. 80ns C. 70ns D. 60ns

【解析】为了能够完全执行完指令的每一个流水段的功能，通常令时钟周期等于执行时间最长的流水段的时间。故而，在指令流经各功能段的时间分别为 90ns、80ns、70ns 和 60ns 的情况下，该计算机的 CPU 时钟周期至少为 90ns。

参考答案：A

2. (原书 第 3 题) 流水线中造成控制相关的原因是执行 () 指令而引起。
- A. 条件转移 B. 访存
C. 算逻 D. 无条件转移

【解析】本题考查造成流水线控制相关的原因。

流水线中的相关是指相邻或相近的两条指令因存在某种关联，后一条指令不能在原指定的时钟周期开始执行。可以将流水线的相关分为以下三类：

(1). 结构相关

当硬件资源满足不了同时重叠执行的指令的要求，而发生资源冲突时，就发生了结构相关。

(2). 数据相关

当指令在流水线中重叠执行时，流水线有可能改变指令读/写操作数的顺序，使之不同于它们在非流水实现时的顺序，这将导致数据相关。

(3). 控制相关

当流水线遇到分支指令（这里指的是条件转移指令）和其他能够改变 PC 值的指令时，就会发生控制相关。

参考答案：A

3. (原书 第 8 题)【2011 年 408 统考】下列给出的指令系统特点中,有利于实现指令流水线的是 ()。

I. 指令格式规整且长度一致

II. 指令和数据按边界对齐存放

III. 只有 Load/Store 指令才能对操作数进行存储访问

A. 仅 I、II

B. 仅 II、III

C. 仅 I、III

D. I、II、III

【解析】以上三个 RISC 特征都能够有效提高取指令或执行指令的效率,从而有利于流水线操作。

参考答案：D

4. (原书 第 9 题)【2013 年 408 统考】某 CPU 主频为 1.03GHz,采用 4 级指令流水线,每个流水段的执行需要 1 个时钟周期,假设 CPU 执行了 100 条指令,在其执行过程中,没有发生任何流水线阻塞,此时流水线的吞吐率为 ()。

A. 0.25×10^9 条指令/秒

B. 0.97×10^9 条指令/秒

C. 1.0×10^9 条指令/秒

D. 1.03×10^9 条指令/秒

【解析】第一条指令从开始到执行完毕,需要 4 个时钟周期的时间,此后每一个时钟周期都会有一条指令完成。故而,100 条指令从第一条开始执行到最后一条执行完毕共需要 103 个时钟周期。因为 CPU 的主频为 1.03GHz,即每秒有 1.03G 个时钟周期。所以执行完这 100 条指令需要 10^{-7} 秒。故而,流水线的吞吐率为 $100 \text{ 条} \div 10^{-7} \text{ 秒} = 1.0 \times 10^9 \text{ 条指令每秒}$ 。

参考答案：C

二. 综合应用题部分

1. (原书 第 3 题)指出下面程序中是否有数据相关?如果有请指出是何种数据相关,并作简要说明。

(1). $I_1 \text{ SUB } R_1, R_2, R_3; (R_2) - (R_3) \rightarrow R_1$

$I_2 \text{ ADD } R_5, R_4, R_1; (R_4) + (R_1) \rightarrow R_5$

(2). $I_3 \text{ MUL } R_3, R_1, R_2; (R_1) \times (R_2) \rightarrow R_3$

$I_4 \text{ ADD } R_3, R_1, R_2; (R_1) + (R_2) \rightarrow R_3$

(3). $I_5 \text{ STO } A, R_1; R_1 \rightarrow M(A), M(A) \text{ 是存储器单元}$

$I_6 \text{ ADD } R_4, R_3, R_2; (R_2) + (R_3) \rightarrow R_4$

(4). $I_7 \text{ LAD } R_1, B; M(B) \rightarrow R_1, M(B) \text{ 是存储器单元}$

$I_8 \text{ MUL } R_1, R_2, R_3; (R_2) \times (R_3) \rightarrow R_1$

【解析】

第(1)组指令中, I_1 指令运算结果应先写入 R_1 , 然后在 I_2 指令中读出 R_1 内容。由于 I_2 指令进入流水线, 变成 I_2 指令在 I_1 指令写入 R_1 前就读出 R_1 内容, 发生写后读(RAW)相关。

第(2)组指令中, 如果 I_4 指令的加法运算完成时间早于 I_3 指令的乘法运算时间, 变成指令 I_4 在指令 I_3 写入前就写入 R_3 , 导致 R_3 的内容错误, 发生写后写(WAW)相关。

第(3)组指令中, I_7 指令和 I_8 指令没有涉及到相同的寄存器, 所以没有发生数据相关。

第(4)组指令中, I_7 指令和 I_8 指令都要写入 R_1 , 发生了写后写(WAW)相关。只要 I_8 的完成在 I_7 之后, 就不会出错。

【经典错误】

我们通过 3 个问题, 来描述数据相关的 3 个经典错误。

(1) WAW 怎么读呢?

很多比较聪明的同学, 看到 WAW, 不管三七二十一直接读成“写后写”。的确, 即使同学们没有区分清楚, 老师看到了同学们这样的答案, 也不知道同学们有没有区分清楚。

(2) 但是, 对于 RAW 和 WAR 怎么读呢?

很显然, 这就要注意顺序了。因为同学们要是区别错误了, 把读后写叫成写后读, 或者把写后读叫成读后写, 错误就明显了。RAW (Read after Write), 显然是写了之后再读, 即写后读。而 WAR (Write after Read) 也是读了之后再写, 即读后写。

其实, WAR、RAW、WAW 都是从后往前读的, 请同学们要牢记。

(3) 那么, 有没有 RAR 相关呢?

这点也需要同学们牢记。其实读后读读到的数据都是一样的, 所以不存在数据相关。其他的三种(WAR、RAW、WAW)都可能因为操作不当而导致读取到数据不是想要读取的数据, 故而存在数据相关。

2. (原书 第 6 题) 设指令流水线分取指令(IF), 指令译码/读寄存器(ID), 执行/有效地址计算(EX), 存储器访问(MEM), 结果寄存器写回(WB)五个过程段。现有下列指令序列进入该流水线。

- ① $\text{ADD } R1, R2, R3 \quad ; (R2) + (R3) \rightarrow R1$
 ② $\text{SUB } R4, R1, R5 \quad ; (R1) - (R5) \rightarrow R4$
 ③ $\text{AND } R6, R1, R7 \quad ; (R1) \text{ AND } (R7) \rightarrow R6$
 ④ $\text{OR } R8, R1, R9 \quad ; (R1) \text{ OR } (R9) \rightarrow R8$
 ⑤ $\text{XOR } R10, R1, R11 \quad ; (R1) \text{ XOR } (R11) \rightarrow R10$

试问：

- (1). 如果处理器不对指令之间的数据相关进行特殊处理，而允许这些指令进入流水线，试问上述指令中哪些指令将从未准备好数据的 R1 寄存器中取到错误的操作数？
 (2). 假如采用将相关指令延迟到所需操作数被写回到寄存器后再执行的方式，以解决数据相关的问题，那么处理器执行该指令序列需占多少个时钟周期？

【解析】

(1). 由上述指令序列可见，ADD 指令后的所有指令都用到 ADD 指令的计算结果。表 5.4 列出了未采用特殊处理的流水线示意，表中 ADD 指令在 WB 段才将计算结果写入寄存器 R1 中，但 SUB 指令在其 ID 段就要从寄存器 R1 中读取该计算结果。同样，AND 指令和 OR 指令也将受到这种相关关系的影响。ADD 指令只有到第五个时钟周期末尾才能结束对寄存器 R1 的写操作，使 XOR 指令可以正常操作，因为它在第六个时钟周期才读寄存器 R1 的内容。

表 5.4 未对数据相关进行特殊处理的流水线

时钟周期	1	2	3	4	5	6	7	8	9
ADD	IF	ID	EX	MEM	WB				
SUB		IF	ID	EX	MEM	WB			
AND			IF	ID	EX	MEM	WB		
OR				IF	ID	EX	MEM	WB	
XOR					IF	ID	EX	MEM	WB

(2). 表 5.5 列出了对这些指令之间数据相关进行特殊处理的流水示意。由此表可见，从第一条指令进入流水线到最后一条指令出结果，共需 12 个时钟周期。

表 5.5 对数据相关进行特殊处理的流水线

时钟周期	1	2	3	4	5	6	7	8	9	10	11	12
ADD	IF	ID	EX	MEM	WB							
SUB		IF				ID	EX	MEM	WB			
AND			IF				ID	EX	MEM	WB		
OR				IF				ID	EX	MEM	WB	
XOR					IF				ID	EX	MEM	WB

本章到此就结束了，请问您有什么疑问吗？任何问题，欢迎您与我们作者进行交流！



梦享团队微信号



梦享团队官方微信公众号



梦享团队新浪微博

第六章 总线

考点 1 总线概述

温馨提示：本部分主要考查：1、总线的基本概念；2、总线的分类；3、总线的组成和性能指标。其中，总线的基本概念和总线的分类部分通常考查概念，总线的性能指标部分，涉及到简单的计算，请同学们多留意。

一. 选择题部分

1. (原书 第 1 题)计算机使用总线结构的主要优点是便于实现积木化,同时()。
- A. 减少了信息传输量
 - B. 提高了信息传输的速度
 - C. 减少了信息传输线的条数
 - D. 三者均正确

【解析】本题考查总线结构的优点。

总线构成计算机系统的互联机构,是系统内各功能部件之间进行信息传送的公共通路。

采用总线结构有以下优点:

- (1). 便于采用模块化结构设计方法,简化系统设计;
- (2). 减少信息传输线的条数;
- (3). 标准总线得到各厂商的支持,便于开发相互兼容的硬件和软件;
- (4). 模块结构便于系统的扩充和升级。

参考答案: C

2. (原书 第 2 题)【2009 年 408 统考】假设某系统总线在一个总线周期中并行传输 4 字节信息,一个总线周期占用 2 个时钟周期,总线时钟频率为 10MHZ,则总线带宽是()。
- A. 10MB/s
 - B. 20MB/s
 - C. 10 MB/s
 - D. 80 MB/s

【解析】本题考查总线带宽的计算方法。

依题意,该系统总线在一个总线周期中并行传输 4 字节信息,一个总线周期占用 2 个

时钟周期。当总线时钟频率为 10MHz 时, 时钟周期为 10^{-7} s, 总线周期为 2×10^{-7} s, 则总线带宽为 $4\text{B} \div (2 \times 10^{-7}\text{s}) = 2 \times 10^7 \text{B/s}$, 即 20MB/s。

参考答案: B

3. (原书 第 5 题) 所谓三总线结构的计算机是指 ()。

- A. 地址线、数据线和控制线三组传输线
- B. I/O 总线、主存总线和 DMA 总线三组传输线
- C. I/O 总线、主存总线和系统总线三组传输线
- D. 设备总线、主存总线和控制总线三组传输线

【解析】本题考查 3 总线结构的基本概念。

【经典总结】借助这个题, 我们来回顾一下单总线、双总线和三总线。

(1). 单总线结构

使用一条单一的系统总线来连接 CPU、主存和 I/O 设备, 叫做单总线结构, 如图 6.1 所示。

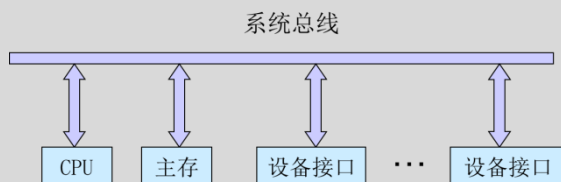


图 6.1 单总线结构

单总线具有结构简单便于扩充等优点, 但由于所有数据的传送都通过这一共享的总线, 因此总线可能成为系统的瓶颈。所以单总线结构多在对速度要求不高的微型机和小型机中。

(2). 双总线结构

双总线结构 (如图 6.2 所示) 在单总线结构的基础上, 在 CPU 和主存之间专门设置了一组高速的存储总线, 使 CPU 可通过专用总线与存储器交换信息, 并减轻了系统总线的负担。

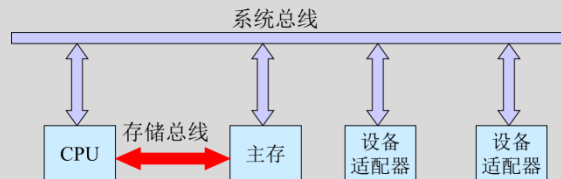


图 6.2 双总线结构

双总线结构有以下特点:

(1). CPU 可通过专用总线与存储器交换信息, 减轻了系统总线的负担;

(2). 高速外设与主存之间仍可通过系统总线实现 DMA 操作;

(3). CPU 通过系统总线与中低速外部设备交换信息。

优缺点: 双总线结构保持了单总线系统简单、易于扩充的优点, 又提高了信息传送的吞吐量。但这是以增加硬件为代价的。

(3). 三总线结构

三总线结构 (如图 6.3 所示) 是在双总线结构的基础上增加 I/O 总线形成的。三总线结构支持通道功能。

通道是一台具有特殊功能的处理器, 又称为 IOP(I/O 处理器), 它分担了一部分 CPU 的功能, 以实现对外设的统一管理及外设与主存之间的数据传送。显然, 由于增加了 IOP, 使整个系统的效率大大提高。

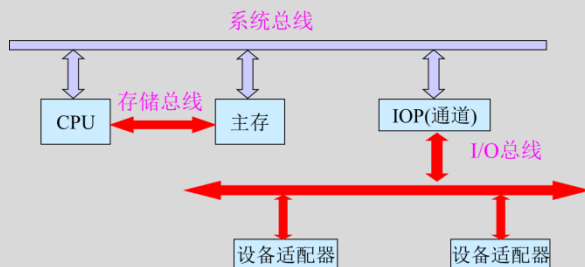


图 6.3 三总线结构

参考答案: B

4. (原书 第 9 题) 系统总线是指 ()。
- 运算器、控制器和寄存器之间的信息传送线
 - 运算器、寄存器和主存之间的信息传送线
 - 运算器、寄存器和外围设备之间的信息传送线
 - CPU、主存和外围设备之间的信息传送线

【解析】本题考查系统总线的基本概念。

系统总线是指连接 CPU、存储器和各种 I/O 模块等主要部件的总线。

参考答案: D

5. (原书 第 18 题) 系统总线上的信号有 ()。
- 地址信号
 - 数据信号、控制信号
 - 控制信号
 - 数据信号、控制信号、地址信号

【解析】本题考查系统总线上传输的信号类别。

系统总线按照传输的数据类型，可以分为控制总线、数据总线和地址总线。控制总线用于传输控制信号和命令，数据总线用于传输数据信号，而地址总线用于传输地址信号。所以，总线上的信号有数据信号、控制信号和地址信号。

参考答案：D

6. (原书 第 21 题) 下列说法中正确的是 ()。

- A. 半双工总线只能在一个方向上传输信息，全双工总线可以在两个方向上轮流传输信息
- B. 半双工总线只能在一个方向上传输信息，全双工总线可以在两个方向上同时传输信息
- C. 半双工总线可以在两个方向上轮流传输信息，全双工总线可以在两个方向上同时传输信息
- D. 半双工总线可以在两个方向上同时传输信息，全双工总线可以在两个方向上轮流传输信息

【解析】根据总线上信号的传递方向，总线可分为单向传输（单工）总线和双向传输（双工）总线，而双工总线又可分为半双工总线和全双工总线。其中，单工总线只能向一个方向传递信号，半双工总线可以在两个方向上轮流传递信号，全双工总线可以在两个方向上同时传递信号。

【故事助记】

关于单工、半双工、全双工通信，网络上有 3 个很好的故事来帮助大家理解和记忆。

(1). 单工

单工就是指 A 只能发信号，而 B 只能接收信号，通信是单向的，就象灯塔之于航船——灯塔发出光信号而航船只能接收信号以确保自己行驶在正确的航线上。

(2). 半双工

半双工是指 A 能发信号给 B，B 也能发信号给 A，但这两个过程不能同时进行。最典型的例子就象我们在影视作品中看到的对讲机一样：

007：呼叫总部，请求支援，OVER。

总部：收到，增援人员将在 5 分钟内赶到，OVER。

007：要 5 分钟这么久？！要快呀！OVER。

总部：……

在这里，通信双方说完一句话后都要说个 OVER，然后切换到接收状态，同时也告之

对方——你可以发言了。如果通信双方同时处于接收状态，或者同时处于发送状态，便不能正常通信了。

(3). 全双工

全双工比半双工更进了一步。在 A 给 B 发信号的同时，B 也可以给 A 发信号。典型的例子就是打电话。

A: 我跟你说呀...

B: 你先听我说，情况是这样的...

A 和 B 在说的同时也能听到对方说的内容，这就是全双工。

参考答案: C

二. 综合应用题部分

1. (原书 第 4 题) (1) 某总线在一个总线周期中并行传送 32 位数据，假设一个总线周期等于一个总线时钟周期，总线时钟频率为 50MHz，总线带宽是多少? (2) 如果一个总线周期中并行传送 64 位数据，总线时钟频率升为 100MHz，总线带宽是多少?

【解析】本题和选择题第 2 题，以及大题的第 2 题类似，解法如下：

- (1). 一个总线周期等于一个总线时钟周期，当时钟频率为 50MHz 时，1 秒可以有时钟周期 50M 个。每个时钟周期可以并行传输 32 位数据信息，即 4 字节信息。易知，总线带宽

$$D = 4B \times 50 \times 10^6 / s = 200MB/s$$

- (2). 当时钟频率为 100MHz 时，1 秒可以有时钟周期 100M 个。每个时钟周期可以并行传输 64 位信息，即 8 个字节，易求得总线带宽

$$Dr = 8B \times 100 \times 10^6 / s = 800MB/s$$

2. (原书 第 6 题) 题目描述如下：

- (1). 某总线在一个总线周期中并行传送 4 个字节的数据，若一个总线周期等于一个时钟周期，总线频率为 33MHz，问总线带宽是多少?
- (2). 若一个总线周期中并行传送 64 位数据，总线时钟提高为 66MHz，问总线带宽是多少?
- (3). 分析影响带宽的有哪些因素?

【解析】本题考查总线带宽的概念、影响总线带宽的主要因素以及计算方法。

- (1). 本题中，一个总线周期等于一个时钟周期。当总线频率为 33MHz 时，一秒钟可以有总线周期 33M 个。因为一个总线周期可以并行传送 4 个字节的数据，所以总线带宽为

$$D = 4B \times 33 \times 10^6 / s = 132MB/s$$

(2). 当时钟频率为 66MHz 时, 一秒可以有时钟周期 66M 个。每个时钟周期可以并行传输 64 位信息, 即 8 个字节, 易求得总线带宽

$$D = 8B \times 66 \times 10^6 / s = 528MB/s$$

(3). 总线带宽是总线能提供的数据传送速率, 通常用每秒传送信息的字节数(或位数)来表示。影响总线带宽的主要因素有: 总线宽度、传送距离、总线发送和接收电路工作频率限制及数据传送形式等。

3. (原书 第 7 题) 考虑以下两种总线:

- (1). 总线 1 是 64 位数据和地址复用的总线。能在一个时钟周期中传输一个 64 位的数据或地址。任何一个读写操作总是先用一个时钟周期传送地址, 然后有 2 个时钟周期的延迟, 从第四时钟周期开始, 存储器系统以每个时钟 2 个字的速度传送, 最多传送 8 个字。
- (2). 总线 2 是分离的 32 位地址和 32 位数据的总线。读操作包括: 一个时钟周期传送地址, 2 个时钟周期延迟, 从第 4 个时钟周期开始, 存储器系统以每时钟 1 个字的速度传输最多 8 个字。对于写操作, 在第 1 个时钟周期内第 1 个数据字与地址一起传输, 经过 2 个时钟周期的延迟后, 以每个时钟 1 个字的速度最多传输 7 个余下的数据字。假定进行 60% 的读操作和 40% 的写操作。

在以下两种情况下, 求这两种总线和存储器能提供的带宽。

- (1). 只进行单数据字的传输。
- (2). 所有的传输都是 8 个字的数据块。

【解析】设时钟周期为 T , 一个字为 32 位, 64 位则为 2 个字。总线采用地址/数据复用技术。所以, 读和写操作所花时间都一样。总线 2 的地址线 and 数据线分离。所以, 读和写操作所花时间不一样。

(1). 单数据字传送的情况

总线 1: 虽然每个时钟周期可传 2 个字, 但只需传一个字。一个读写操作的第一个时钟周期用于传送地址, 接下来的两个时钟周期为延迟。因为一个时钟周期能传输一个 64 位的数据或者地址, 当从第 4 个时钟周期开始, 存储器利用一个时钟周期的时间来传送一个字。综上, 总共需要花费时间为 $4T$ 。因为在单数据字传送的情况下, 每个时钟周期只传送一个字。因此, 带宽为 $4B/4T = 1B/T$ 。

总线 2: 该总线在读操作下第 1 个时钟周期传送的地址, 2 个时钟周期的延迟, 第 4 个时钟周期开始以每个时钟周期传送 1 个字的速度传输数据, 所以读一字时间为: $3+1=4T$;

对于写操作, 在第 1 个时钟周期内第 1 个数据字和地址一起传送, 经过两个时钟周期的延迟, 以每个时钟 1 个字的速度传输余下的数据字。所以, 写 1 个字所需时间为 $3T$ 。因此, 带宽为

$$D=(4B/4T) \times 60\% + (4B/3T) \times 40\% = 1.1 \text{ B/T}。$$

显然, 在只进行单数据字传输的情况下, 总线 2 比总线 1 的带宽要大。

(2). 8 个字的数据块传送情况

总线 1: 在传送 8 个字的情况下, 总线 1 在读写操作的第一个时钟周期用于传送地址, 经过两个时钟周期的延迟, 从第 4 个时钟周期开始, 每个时钟周期传输 2 个字, 最多传送 8 个字, 需要 4 个时钟周期才能传送完这 8 个字 (即第 4、5、6、7 个时钟周期用于传送这 8 个字)。所以, 传送这 8 个字的数据块, 所花费时间总共为 $4T+3T$ 。也即读或写 8 个字所花时间都为 $7T$ 。因此, 带宽为

$$D=8 \times 4B/7T=(32/7) \text{ B/T}。$$

总线 2: 该总线采用地址线 and 数据线分离的方法, 读操作时, 第一个时钟周期传送地址, 经过 2 个时钟周期的延迟, 从第 4 个时钟周期开始以每个时钟周期传输一个字节的速度传送数据。故而, 读 8 个字时间为: $3+8=11T$; 在写操作下, 第一个时钟周期传送了第一个数据字和地址, 经过 2 个时钟周期的延迟, 从第 4 个时钟周期开始, 以每个时钟周期传送 1 个字的速度传送余下的 7 个字。故而, 写 8 个字所花费的总时间为: $3+7=10T$ 。因此, 带宽为

$$D=8 \times 4B/11T \times 60\% + 8 \times 4B/10T \times 40\% = (32/10.6) \text{ B/T}。$$

显然, 总线 1 的带宽比总线 2 的带宽要大。

考点 2 总线仲裁

温馨提示: 总线仲裁包括集中仲裁方式和分布仲裁方式。其中, 集中仲裁方式分为计数器定时查询、链式查询和独立请求三种方式, 也是本章一个重要的命题内容, 请大家掌握这三种集中式仲裁方式的特点和区别。

一. 选择题部分

1. (原书 第 1 题) 三种常见的总线集中控制优先权的仲裁方式中, () 方式响应速度最快, 但增加了控制线数。

A. 链式查询

B. 计数器定时查询

C. 独立请求

D. 自动排队

【解析】本题考查三种常见的总线集中控制方式的特点。

由于总线上连接着多个部件，何时由哪个部件发送信息、如何定时、如何防止信息丢失、如何避免多个设备同时发送信息、如何规定接收部件等一系列问题都需要总线控制器来统一管理，主要包括总线的判优控制（仲裁逻辑）和通信控制。

总线仲裁逻辑（即总线判优）可分为集中式和分布式两种，前者将控制逻辑集中在一处（如在 CPU 中），后者将控制逻辑分散在总线的各个部件之上。

集中式仲裁中，每个功能模块有两条线连到中央仲裁器：一条是送往仲裁器的总线请求信号线 BR，一条是总线仲裁器送出的总线授权信号线 BG。集中仲裁方式分为链式查询、计数器定时查询和独立请求查询三种方式。

独立请求方式（如图 6.4 所示）下，每个连接到总线的设备都有一组单独的总线请求信号 BR_i 与总线授权信号 BG_i。每个设备请求使用总线时，它们各自发出自己的总线请求信号。中央仲裁器中设置了一个专门的排队电路，由它根据一定的优先次序决定优先响应哪个设备的请求，然后给该设备总线授权信号 BG_i。

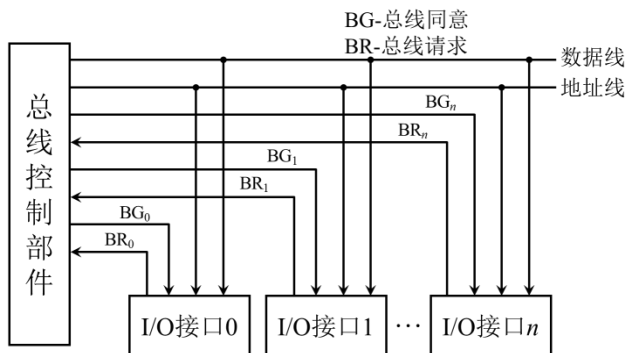


图 6.4 独立请求方式

独立请求方式的优点是响应时间快，即确定优先响应的设备所花费的时间少，用不着一个设备接一个设备地查询。其次，对优先次序的控制相当灵活。它可以预先设定，例如 BR₀ 优先级最高，BR₁ 次之，……，BR_n 最低；也可以通过程序来改变优先次序；还可以用屏蔽（禁止）某个请求的办法，不响应来自该设备的请求。

但是，独立请求方式多出来很多控制线，每一个接口都有仲裁信号线 BR 和 BG 各一根。

参考答案：C

2. （原书第2题）三种集中式总线控制中，（ ）方式对电路故障最敏感。

A. 链式查询

B. 计数器定时查询

C. 独立请求

D. 以上都不对

【解析】本题考查总线控制方式中的链式查询方式的特点。

链式查询方式（如图 6.5 所示）采用菊花链的方式连接所有具有总线使用能力的部件，各设备共用一根总线请求信号线 BR、总线授权信号线 BG、总线忙信号线 BS 与中央仲裁器连接。

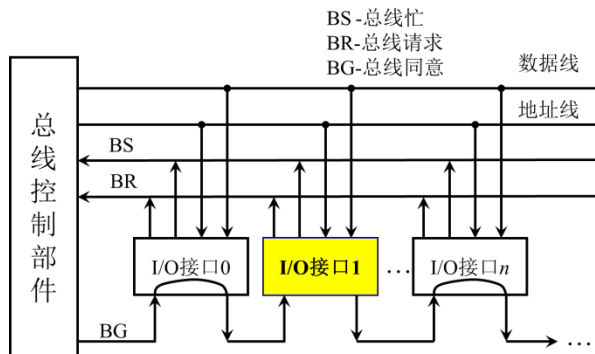


图 6.5 链式查询方式

【经典总结】

链式查询方式的主要特点是，总线授权信号 BG 串行地从一个 I/O 接口传送到下一个 I/O 接口。假如 BG 到达的接口无总线请求，则继续往下查询；假如 BG 到达的接口有总线请求，BG 信号便不再往下查询。这意味着该 I/O 接口就获得了总线控制权。

链式查询方式的优点是，只用很少几根线就能按一定优先次序实现总线仲裁，并且这种链式结构很容易扩充设备。

链式查询方式的缺点是对询问链的电路故障很敏感，如果第 i 个设备的接口中有关链的电路有故障，那么第 i 个以后的设备都不能进行工作。另外查询链的优先级是固定的，如果优先级高的设备出现频繁的请求时，那么优先级较低的设备可能长期不能使用总线。

参考答案：A

3. （原书 第 4 题）在计数器定时查询方式下，若计数从 0 开始，则（ ）。
- 设备号小的优先级高
 - 每个设备使用总线的机会相等
 - 设备号大的优先级高

【解析】本题考查从 0 开始计数时计数器查询方式的特点。

【经典总结】

计数器查询方式（如图 6.6 所示）下，总线上的设备通过总线请求信号 BR，发出请求，总线仲裁器接收到请求信号后，在总线忙信号 BS 为“0”的情况下，让计数器开始计数，计

数值通过一组地址线发往各设备（不仅仅是发出请求的设备）。每个设备有一个地址判别电路，如果地址线上的计数值与总线请求设备地址一致，则该设备对 BS 线置“1”，表示该设备获得了总线使用权，同时中止计数查询。

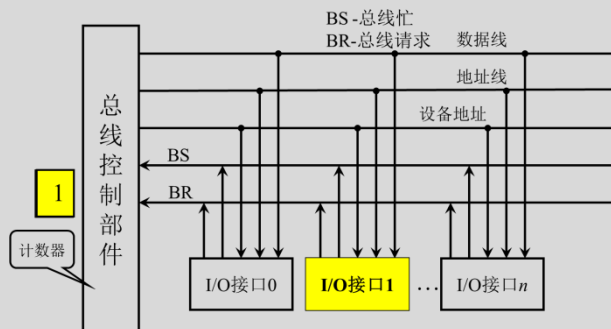


图 6.6 计数器查询方式

计数器定时查询方式下，三种计数方式的优先级分别如下：

- (1). 若每次计数从“0”开始，设备号小的优先级高，设备号大的优先级低；
- (2). 若每次计数都从设备号最大的编号开始计数，则设备号大的优先级高，设备号小的优先级低；
- (3). 若从上一次终止计数的位置的下一个位置开始计数，则各设备使用总线的优先级相等；
- (4). 若计数器的初值用程序来设定，就可以方便地改变优先级次序。

参考答案：A

4. (原书 第 5 题) 在菊花链方式(即链式查询方式)下, 越靠近控制器的设备()。
- A. 得到总线使用权的机会越多, 优先级越高
 - B. 得到总线使用权的机会越少, 优先级越低
 - C. 得到总线使用权的机会越多, 优先级越低
 - D. 得到总线使用权的机会越少, 优先级越高

【解析】本题考查链式查询方式的特点。

从第 2 题的图中可以发现，在查询链中离中央仲裁器最近的设备具有最高优先级，离中央仲裁器越远的设备，优先级越低。而且，链式查询方式是通过接口的优先级排队电路来实现的。

参考答案：A

二. 综合应用题部分

1. (原书 第 2 题) 总线的集中式仲裁有哪几种方式? 各有什么优缺点?

【解析】总线判优控制(或称仲裁逻辑)解决多个部件同时申请总线时的使用权分配问题,分为集中式和分布式两种,前者将控制逻辑集中在一处(如在 CPU 中),后者将控制逻辑分散在与总线连接的各个部件或设备上。

集中查询方式有以下三种方式:

(1). 链式查询方式

优点: 只用很少几根线就能按一定的优先次序实现总线仲裁, 容易扩充设备。

缺点: 对查询链的电路故障很敏感, 如果第 i 个设备的接口中有关键的电路有故障, 那么该设备以后的所有设备都不能进行工作了; 而且链式查询方式的优先级比较固定, 如果优先级高的设备出现频繁请求 那么低优先级的设备可能长期不能使用总线。

(2). 计数器定时查询方式

优点: 优先级可以变动, 比较灵活。

缺点: 增加了线数, 所以增加了成本。

(3). 独立请求方式

优点: 相应时间快, 确定优先响应次序所花费的时间少, 控制优先次序也比较灵活。

缺点: 比较复杂, 实现比较困难。

【故事助记】

班级开讨论会, 每一位成员都有发言的机会, 有以下的几种争取发言权的机会:

(1). 根据离讲台的远近(假设所有人只能排成一排)来判断发言权的大小, 离讲台近的同学比离讲台远的同学有更高的发言权。班长说, 现在可以发言了, 从前往后, 大家想发言的可以依次发言(BUSY 置 0)。那么, 当有两个或者多个同学想发言的时候, 找一个离讲台最近的同学, 让他先发言, 等到他发言结束之后, 找到剩下的要发言的同学里面离讲台最近的同学, 把发言权交给他……如此持续下去, 直到离讲台最远的那位要发言的同学发言完毕为止。

(2). 比较随意一些, 哪位同学想发言都可以, 班长在讲台上制定了一个优先级策略, 来决定让哪一位同学先发言。

(3). 采用计数的方式, 从某一个数开始计数。比如, 计数为 1 的时候, 班长就会说, “现在 1 号同学可以发言了, 请问你要发言吗?”若是第 1 位同学要发言, 那么等待第一位同学发言完毕, 若该同学没有发言, 则跳过。班长继续问第二位同学“现在 2 号同学可以发言了, 请问你要发言吗?”……如此询问下去, 直到最后一位同学。

故事中, 班长相当于仲裁器, 方案 1 采用了链式查询的方式, 方案 2 采用独立请求方式, 方案 3 采用了计数器查询方式。

考点 3 总线操作和定时

温馨提示：总线操作和定时，主要有同步定时方式和异步定时方式，请同学们注意区别这两种定时方式，并注意异步通信方式下的不互锁、半互锁和全互锁三种方式。

一. 选择题部分

1. (原书 第 1 题) 总线的异步通信方式 ()。

- A. 采用时钟信号，不采用握手信号
- B. 不采用时钟信号，只采用握手信号
- C. 既采用时钟信号，又采用握手信号
- D. 既不采用时钟信号，又不采用握手信号

【解析】本题考查总线的异步通信方式特点。

异步通信控制方式下，采用非时钟定时，没有一个公共的时钟标准。因此，能够连接带宽范围很大的各种设备。总线能够加长而不用担心时钟偏移问题。

采用握手协议（应答方式）。由一系列步骤组成，只有当双方都同意时，发送者或接收者才会进入到下一步，协议通过一对附加的“握手”信号（Ready、Ack）来实现。

参考答案：B

2. (原书 第 2 题) 同步控制是 ()。

- A. 只适用于 CPU 控制的方式
- B. 只适用于外围设备控制的方式
- C. 由统一时序信号控制的方式
- D. 所有指令执行时间都相同的方式

【解析】本题考查同步控制的定义。

同步定时方式要求所有的模块由统一的时序信号（即时钟脉冲）进行操作控制，各模块的所有动作均在时钟周期的开始产生，并且多数动作在一个时钟周期内完成。

参考答案：C

3. (原书 第 4 题) 在各种异步通信方式中，() 速度最快。

- A. 全互锁
- B. 半互锁
- C. 不互锁

【解析】本题考查对三种常见的异步通信握手方式的速度进行比较。

异步通信有非互锁、半互锁和全互锁三种方式，如图 6.7 所示。

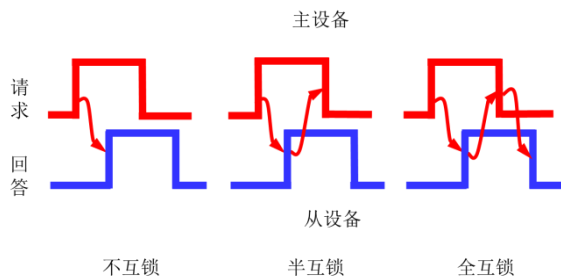


图 6.7 异步通信的三种互锁方式

三种“握手”方式中，只有不互锁方式的请求信号和回答信号没有相互的制约关系，主设备在发出请求信号后，不必等待回答信号的到来，便自己撤销了请求信号，所以速度最快。

【经典例子】

一男生给女朋友发短信，“我想你了，下午两点在公园门口等你，咱们一起游玩去！”。有三种方式。第一种，直接发过去，不管女朋友有没有收到，都当作人家知道了就好了。可是，男生还是忍不住担心，她有没有收到我的短信呢？万一没有收到，我去了公园，人家不知道，人没来怎么办？

第二种，发过去，并提示女朋友一定要给一个确认才行。于是，有了一个通信的过程：

男：我想你了，下午两点在公园门口等你，咱们一起游玩去！

女：收到你的短信了，我也想你，我收拾一下，就出发，下午两点见。

这时候，男生收到女朋友这条短信，心里肯定乐开花了。可是，女朋友回了男朋友短信，她也得想，我回复他了，万一他没有收到我的回复，以为我没有回他，认为我下午没空，于是下午他没去怎么办？让他收到我的回复之后，在告诉我他收到了我的回复了，就好了。于是，诞生了第三种短信通信方式。

男：我想你了，下午两点在公园门口等你，咱们一起游玩去！

女：收到你的短信了，我也想你，我收拾一下，就出发，下午两点见。

男：嗯，好！下午两点见！

女生收到这条消息，就认为男生收到了她的回复了。而男生也知道女生收到了他的约会信息。这样，一个全互锁的通信就完成了。

参考答案：C

4. (原书 第5题) 若一个 8 bit 组成的字符至少需 10 bit 来传送，这是 () 传送方式。

A. 同步

B. 异步

C. 并联

D. 混合

【解析】异步传送方式并不要求发送方和接收方的时钟完全一样，字符与字符间的传输是异步的。异步传输存在一个潜在的问题，即接收方并不知道发送方的数据会在什么时候到达。在它检测到数据并做出响应之前，第一个比特已经过去了。

【经典问题】怎么理解？

这就像有人出乎意料地从后面走上来跟你说话，而你没来得及反应过来，漏掉了最前面的几个词。

因此，每次异步传输的信息都以一个起始位开头，它通知接收方数据已经到达了，这就给了接收方响应、接收和缓存数据比特的时间；在传输结束时，一个停止位表示该次传输信息的终止。这两个位都是用来“同步”的。

同步传送方式的同步传输过程中，发送方和接收方的时钟是统一的、字符与字符间的传输是同步无间隔的。所以，不需要这样的“同步”信号。

参考答案：B

考点 4 总线标准

温馨提示：总线标准在自主命题高校中，出现比较少。但是，在 408 统考的选择题中，总线标准却是一个常考的内容，请参加 408 统考的同学多留心。

一. 选择题部分

1. (原书 第 2 题) 描述 PCI 总线中基本概念不正确的句子是 ()。
- A. PCI 总线是一个与处理器无关的高速外围设备
 - B. PCI 总线的基本传输机制是猝发式传送
 - C. PCI 设备一定是主设备
 - D. 系统中只允许有一条 PCI 总线

【解析】本题考查 PCI 总线的基本概念。

PCI 是一个与处理器无关的高速外围总线，其基本机制是猝发式传送。PCI 设备可以是主设备，也可以是从设备，或兼而有之。

在 PCI 设备中不存在 DMA 的概念，这是因为 PCI 总线支持无限的猝发式传送。这样，传统总线上用 DMA 方式工作的设备移植到 PCI 总线上时，采用主设备工作方式即可。

系统中允许有多条 PCI 总线，它们可以使用 HOST 桥与 HOST 总线相连，也可使用

PCI/PCI 桥与已和 HOST 总线相连的 PCI 总线相连,从而得以扩充整个系统的 PCI 总线负载能力。

参考答案: CD

2. (原书 第 4 题)【2010 年 408 统考】下列选项中的英文缩写均为总线标准的是 ()。
- A. PCI、CRT、USB、EISA
 - B. ISA、CPI、VESA、EISA
 - C. ISA、SCSI、RAM、MIPS
 - D. ISA、EISA、PCI、PCI-Express

【解析】本题考查常见的总线标准。

CRT (Cathode Ray Tube) 是一种使用阴极射线管的显示器,其结构不需要大家掌握。CPI 为执行一条指令所需要的平均时钟周期。RAM 为随机访问存储器。MIPS 为每秒百万条指令数。显然,这些都与总线标准无关。

参考答案: D

3. (原书 第 5 题)【2012 年 408 统考】下列关于 USB 总线特性的描述中,错误的是 ()。
- A. 可实现外设的即插即用和热插拔
 - B. 可通过级联方式连接多台外设
 - C. 是一种通信总线,可连接不同外设
 - D. 同时可传输 2 位数据,数据传输率高

【解析】USB 总线是串行总线,数据被 1 位接着 1 位的传输,不可能同时传输 2 位数据。所以, D 答案正确。

参考答案: D

本章到此就结束了,请问您有什么疑问吗?任何问题,欢迎您与我们作者进行交流!



梦享团队微信号



梦享团队官方微信公众号



梦享团队新浪微博

第七章 输入输出 (I/O) 系统

考点 1 I/O 系统基本概念

温馨提示：本部分主要考查 I/O 系统的基本概念，考查的都是基础知识，请同学们作一个简单的了解。

一. 选择题部分

1. (原书 第 1 题) CPU 与 I/O 设备之间传送信息时，串行传送和并行传送两种方式相比，()。
- A. 并行传送信息传送速度快，但要求数据线较多
 - B. 并行传送信息传送速度快，而且要求数据线少
 - C. 并行传送与串行传送速度相差不大，但串行传送要求数据线少
 - D. 串行传送信息传送速度快，且要求数据线少

【解析】本题考查了 CPU 与 I/O 设备之间使用串行方式和并行方式传送数据的特点。

串行传输方式是 n 位字长信息通过一条通信信号线一位一位传送。串行传输方式又分为单工、半双工及全双工，其中全双工需要两条通信信号线。串行传输的传输速度慢，但传输距离远。

并行传输方式是 n 位字长信息由 n 位通信信号线同时传送。并行传输的传输速度快，但传输距离近。

参考答案：A

考点 2 外部设备

温馨提示：本考点主要考查：1、输入设备，主要包括键盘、鼠标；2、输出设备，主要包括显示器、打印机；3、外存储器，主要包括硬盘存储器、磁盘阵列和光盘存储器。

一. 选择题部分

1. (原书 第1题) 用户与计算机通信的界面是 ()。

- A. CPU
- B. 外围设备
- C. 应用程序
- D. 系统程序

【解析】本题考查了 CPU 与 I/O 设备之间使用串行方式和并行方式传送数据的特点。

串行传输方式是 n 位字长信息通过一条通信信号线一位一位传送。串行传输方式又分为单工、半双工及全双工，其中全双工需要两条通信信号线。串行传输的传输速度慢，但传输距离远。

并行传输方式是 n 位字长信息由 n 位通信信号线同时传送。并行传输的传输速度快，但传输距离近。

参考答案：B

2. (原书 第8题) 下列外存中，属于顺序存取存储器的是 ()。

- A. 软盘
- B. 硬盘
- C. 磁带
- D. 光盘

【解析】本题考查顺序存取存储器。

在计算机中，常用的数据存取方式主要有顺序存取、直接存取、随机存取和相联存取四种。如果将存储器按照其存取方式来分类，可以分为如下五类。

(1). 随机存取存储器

随机存取是指 CPU 可以对存储器中的内容随机地存取，且其存取时间相同，与信息所处位置无关。RAM 具有读/写方便，使用灵活等优点，但断电后无法保存信息，因此只能用于暂存数据，可用于主存和高速缓冲存储器。

RAM 又可分为 DRAM 和 SRAM 两种。其中 DRAM 的信息会随时间逐渐消失，因此需要定时对其进行刷新来维持信息不丢失；SRAM 在不断电的情况下信息能够一直保持而不会丢失。

(2). 只读存储器

ROM 采用的存取方式也是随机存取，且其存储时间也相同。但 ROM 的信息已固化在存储器中。ROM 出厂时其内容由厂家采用掩膜技术 (mask) 写好，其信息只可读，无法改写。一般用于存放系统程序 BIOS (Basic Input Output System, 基本输入输出系统) 和用于微程序控制。

(3). 顺序存取存储器 (Sequential Access Memory, SAM)

SAM 的存取方式与前两种完全不同。它只能按某种顺序存取，存取时间的长短与信息在存储体上的物理位置有关，所以 SAM 只能用平均存取时间作为衡量存取速度的指标。磁

带机就是这样一类存储器。

(4). 直接存取存储器 (Direct Access Memory, DAM)

DAM 采用直接存取方式进行信息存取, 即当要存取所需的信息时, 第一步是直接指向整个存储器中的某个小区域 (如磁盘上的磁道); 第二步在小区域内顺序检索或等待, 直至找到目的地后再进行读/写操作。DAM 的存取时间也是与信息所在的物理位置有关的, 但比 SAM 的存取时间要短。磁盘机就属于这类存储器。

由于 SAM 和 DAM 的存取时间都与存储体的物理位置有关, 所以又可以把它们统称为串行访问存储器。

(5). 相联存储器 (Content Addressable Memory, CAM)

CAM 是一种特殊的存储器, 是一种基于数据内容进行访问的存储设备。当对其写入数据时, CAM 能够自动选择一个未用的空单元进行存储; 当要读出数据时, 不是给出其存储单元的地址, 而是直接给出该数据或者该数据的一部分内容。CAM 对所有的存储单元中的数据同时进行比较并标记符合条件的所有数据以供读取。由于比较是同时、并行进行的, 所以这种基于数据内容进行读写的机制, 其速度比基于地址进行读写的方式要快许多。

参考答案: C

3. (原书 第 14 题) 磁盘存储器的等待时间通常是指 ()。

- A. 磁盘旋转一周所需的时间
- B. 磁盘旋转半周所需的时间
- C. 磁盘旋转 $\frac{1}{3}$ 周所需的时间
- D. 磁盘旋转 $\frac{2}{3}$ 周所需的时间

【解析】本题考查磁盘存储器的等待时间。磁盘存储器的等待时间通常指的是磁盘旋转半周所需要的时间。

参考答案: B

4. (原书 第 17 题) 在磁盘和磁带两种磁表面存储器中, 存取时间与存储单元的物理位置有关, 按存储方式分, ()。

- A. 二者都是串行存取
- B. 磁盘是部分串行存取, 磁带是串行存取
- C. 磁带是部分串行存取, 磁盘是串行存取
- D. 二者都是并行存取

【解析】DAM 采用直接存取方式进行信息存取，即当要存取所需的信息时，第一步是直接指向整个存储器中的某个小区域（如磁盘上的磁道）；第二步在小区域内顺序检索或等待，直至找到目的地后再进行读/写操作。DAM 的存取时间也是与信息所在的物理位置有关的，但比 SAM 的存取时间要短。磁盘机就属于这类存储器。

但是，磁带机不能随机存取，只能顺序存取。故而，选择 B 答案。

参考答案: B

5. (原书第18题) 【2010年408统考】假定一台计算机的显示存储器用 DRAM 芯片实现，若要求显示分辨率为 1600×1200 ，颜色深度为 24 位，帧频为 85 Hz，显存总带宽的 50% 用来刷新屏幕，则需要的显存总带宽至少约为
- A. 245 Mbps B. 979 Mbps
C. 1958 Mbps D. 7834 Mbps

【解析】显示分辨率为 1600×1200 ，颜色深度为 24 位，帧频为 85Hz，显存每秒需要提供屏幕刷新的信息量为： $1.6 \times 10^3 \times 1.2 \times 10^3 \times 24 \times 85 = 3916.8 \times 10^6$ 。显存总带宽的 50% 用来刷屏，需总线带宽为 $2 \times 3916.8 \times 10^6 \text{bps} \approx 7834 \text{Mbps}$ 。

参考答案: D

二. 综合应用题部分

1. (原书第1题)某磁盘组有6片磁盘,每片可有2个记录面,存储区域内径为22cm,外径为33cm,道密度40道/cm,位密度400b/cm,转速2400 r/min。试问:
- (1). 共有多少个存储面可用?
 - (2). 共有多少个圆柱面?
 - (3). 整个磁盘组的总存储总量有多少?
 - (4). 数据传送率是多少?
 - (5). 如果某文件长度超过一个磁盘的容量,应将它记录在同一存储面上还是记录在同一圆柱面上?为什么?
 - (6). 如果采用定长信息块记录格式,直接寻址的最小单位是什么?寻址命令中如何表示磁盘地址?

【解析】

- (1). 磁盘组共有 6 片磁盘，每片磁盘两个记录面，共 $6 \times 2 = 12$ (面)，即共有 12 个存储面可用。
- (2). 存储区域内径是 22cm，外径是 33cm，道密度是 40 道/cm， $40 \times (33 - 22) / 2 = 220$

【解析】现代计算机系统中外部设备的种类繁多，各类外部设备不仅结构和工作原理不同，而且与主机的连接方式也是复杂多变的。

主机和外设的连接方式有辐射型连接、总线型连接等。输入输出接口（I/O 接口）是主机和外设之间的交接界面，通过接口可以实现主机和外设之间的信息交换。

主机和外设各自具有自己的工作特点，它们在信息形式和工作速度上具有很大的差异，接口正是为了解决这些差异而设置的。

接口主要有以下功能（如图 7.1 所示）：

- (1). 实现主机和外设的通信联络控制；
- (2). 进行地址译码和设备选择；
- (3). 实现数据缓冲；
- (4). 数据格式的变换；
- (5). 传递控制命令和状态信息。

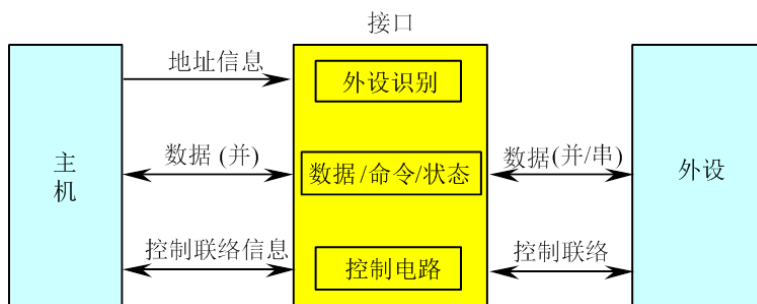


图 7.1 接口的结构和功能

接口中要分别传送数据信息、控制信息和状态信息。

数据信息、控制信息和状态信息都通过数据总线来传送。大多数计算机都把外部设备的状态信息视为输入数据，而把控制信息看成输出数据，并在接口中分设各自相应的寄存器，赋以不同的端口地址，各种信息分时地使用数据总线传送到各自的寄存器中去。

参考答案：C

3. （原书 第 6 题）下列关于 I/O 端口和接口的说法中，正确的是（ ）。
- A. 按照不同的数据传送格式，可以将接口分为同步传送接口和异步传送接口
 - B. 在统一编址方式下，存储单元与 I/O 设备是靠不同的地址线来区分的
 - C. 在独立编址方式下，存储单元与 I/O 设备是靠不同的地址线来区分的
 - D. 在独立编址方式下，CPU 需要设置专门的输入/输出指令访问端口

【解析】按照不同的数据传送格式，可以将接口分为并行接口和串行接口，前者可以

一次传送多位数据，后者一次只能传送一位数据，故而 A 答案错误。

在统一编址方式下，存储单元与 I/O 设备采用了相同的地址线，所以靠相同的地址线来区分 I/O 设备与存储单元是不可行的。但是，统一编址方式下，地址线上传送不同的地址码时，可以命中不同的存储单元或者 I/O 设备，这可以用来区分存储单元和 I/O 设备。

在独立编址方式下,存储单元与 I/O 设备是靠不同的指令来区分的。所以,C 答案错误。

在统一编址方式下, 存储单元和 I/O 设备可以用相同的指令来访问, 但是在独立编址方式下却是靠不同的指令来区分的。所以, D 答案正确。

参考答案: D

二. 综合应用题部分

1. (原书 第1题)说明 I/O 地址与存储器地址的统一编址和独立编址各自的含义和特点。

【解析】

(1). 独立编址：将 I/O 端口单独编排地址，独立于存储器地址。优点：I/O 端口的地址空间独立，控制和地址译码电路相对简单，专门的 I/O 指令使程序清晰易读。缺点：I/O 指令没有存储器指令丰富。

(2). 统一编址：将 I/O 端口与存储器地址统一编排，共享一个地址空间。优点：不需要专门的 I/O 指令，I/O 数据存取灵活。缺点：占去部分存储器空间，程序不易阅读（在汇编语言级不容易区别访问存储器和访问外设）。

考点 4 I/O 方式

温馨提示：I/O 方式有 4 种，即程序查询方式、程序中断方式、DMA 方式和通道方式。请同学们主要这几种 I/O 方式的特点和区别，并重点掌握中断响应过程、中断处理过程、多重中断和中断屏蔽方法，该部分常以综合题的方式考查，分值很高。

一. 选择题部分

1. (原书 第 6 题) 主机与外围设备之间传送数据时, CPU 工作效率最高的方式是 ()。

- A. 程序查询方式 B. 中断方式
C. DMA 方式 D. 暂定访问方式

【解析】 本题考查几种常见的 I/O 方式的效率。

无论程序查询还是程序中断方式，主要的工作都是由 CPU 执行程序完成的，这需要花费时间，因此不能实现高速外设与主机的信息交换。

DMA 方式是在外设和主存之间开辟一条“直接数据通道”，在不需要 **CPU** 干预也不需要软件介入的情况下在两者之间进行的高速数据传送方式。在 **DMA** 传送方式中，对数据传送过程进行控制的硬件称为 **DMA** 控制器。当外设需要进行数据传送时，通过 **DMA** 控制器向 **CPU** 提出 **DMA** 传送请求，**CPU** 响应之后将让出系统总线，由 **DMA** 控制器接管总线进行数据传送。

参考答案：C

2. (**原书第14题**)DMA接口访问主存时让CPU处于等待状态，等一批数据访问结束后，CPU再恢复工作，这种情况称为()。
- A. 暂停CPU访问主存 B. 周期挪用
C. DMA与CPU交替访问 D. DMA

【解析】本题考查 DMA 方式的暂停 CPU 访问主存法。

DMA 的传送方法主要有以下三类:

(1). CPU 停止访问主存法

这种方法是用 DMA 请求信号迫使 CPU 让出总线控制权。CPU 在现行机器周期执行完成之后，使其数据、地址总线处于三态，并输出总线批准信号。每次 DMA 请求获得批准，DMA 控制器获得总线控制权以后，连续占用若干个存取周期（总线周期）进行成组连续的数据传送，直至批量传送结束，DMA 控制器才把总线控制权交回 CPU。如图 7.2 所示。

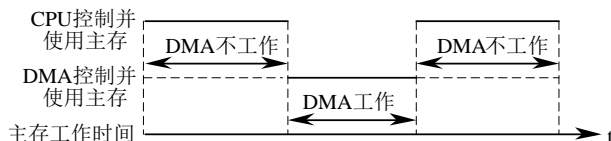


图 7.2 CPU 停止访问主存法

(2). DMA 与 CPU 交替访问

把原来的一个存取周期分成两个时间片，一片分给 CPU，一片分给 DMA，使 CPU 和 DMA 交替地访问主存。这种方法无须申请和归还总线，使总线控制权的转移几乎不需要什么时间，所以对 DMA 传送来讲效率是很高的，而且 CPU 既不停止现程序的运行，也不进入保持状态，在 CPU 不知不觉中便进行了 DMA 传送。但这种方法需要主存在原来的存取周期内为两个部件服务，如果要维持 CPU 的访存速度不变，就要求主存的工作速度提高

一倍。如图 7.3 所示。

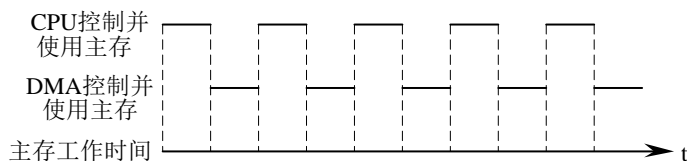


图 7.3 DMA 与 CPU 交替访问法

(3). 周期挪用法

周期挪用法是前两种方法的折衷。当外设有 DMA 请求时，CPU 按程序要求访问主存。一旦外设有 DMA 请求并获得 CPU 批准后，CPU 让出一个周期的总线控制权，由 DMA 控制器控制系统总线，挪用一個存取周期进行一次数据传送，传送一个字节或一个字。然后，DMA 控制器将总线控制权交回 CPU，CPU 继续进行自己的操作，等待下一个 DMA 请求的到来。重复上述过程，直至数据块传送完毕。如图 7.4 所示。

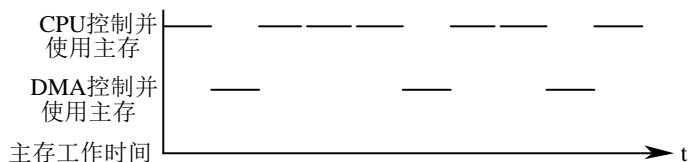


图 7.4 周期挪用法

参考答案：A

3. (原书第 17 题) 中断工作方式的五个步骤：中断响应、中断排队、中断请求、中断返回、中断服务，其工作顺序是 ()。
- 中断响应、中断排队、中断请求、中断返回、中断服务
 - 中断响应、中断请求、中断排队、中断服务、中断返回
 - 中断请求、中断响应、中断排队、中断服务、中断返回
 - 中断请求、中断排队、中断响应、中断服务、中断返回

【解析】本题考查中断工作方式的主要步骤。

这里所说的中断全过程指的是从中断源发出中断请求开始，CPU 响应这个请求，现行程序被中断，转至中断服务程序，直到中断服务程序执行完毕，CPU 再返回原来的程序继续执行的整个过程。

中断全过程分为 5 个阶段：①中断请求，②中断判优（排队），③中断响应，④中断处理（服务），⑤中断返回。

其中，中断处理就是执行中断服务程序，这是中断系统的核心。

参考答案: D

4. (原书 第 23 题)采用 DMA 方式传送数据时,每传送一个数据就要用一个()。

- A. 指令周期 B. 数据周期
C. 存储周期 D. 总线周期

【解析】在第 14 题的周期挪用部分，我们提到：当外设没有 DMA 请求时，CPU 按程序要求访问主存；一旦外设 DMA 请求并获得 CPU 批准后，CPU 让出一个周期的总线控制权，由 DMA 控制器控制系统总线，**挪用一个存取周期进行一次数据传送，传送一个字节或一个字**；然后，DMA 控制器将总线控制权交回 CPU，CPU 继续进行自己的操作，等待下一个 DMA 请求的到来。重复上述过程，直至数据块传送完毕。

借助此题，我们看看 DMA 控制器与 CPU 的连接情况，如图 7.5 所示。

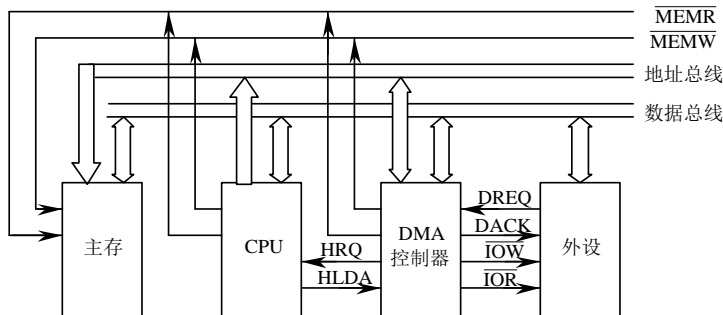


图 7.5 DMA 控制器的连接

参考答案: C

5. (原书 第 32 题) 单级中断与多级中断的区别是 ()。

- A. 单级中断只能实现单中断，而多级中断可以实现多重中断
- B. 单级中断的硬件结构是一维中断，而多级中断的硬件结构是二维中断
- C. 单级中断处理机只通过一根外部中断请求线接到它的外部设备系统；而多级中断，每一个 I/O 设备都有一根专用的外部中断请求线

【解析】 本题考查单级中断和多级中断的区别。

根据计算机系统中断处理策略的不同，可分为单级中断系统和多级中断系统。

单级中断系统是中断结构中最基本的形式。在单级中断系统中，所有的中断源都属于同一级，所有中断源触发排成一行，其优先级次序是离 CPU 近的优先级高，离 CPU 远的优先级低。当响应某一个中断请求时，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子程序。也就是说，即使有更高优先权的中断请求，也只能等当前中断服务程序执行完毕之后才能够被响应。

多级中断系统是指计算机系统中有相当多的中断源，根据各中断条件的轻重缓急程度分成若干级别，每个中断级别分配一个优先权。一般来说，优先权高的中断级别可以打断优先权低的中断级别的中断服务程序。

图 7.6 是单重中断和多重中断的服务程序流程。

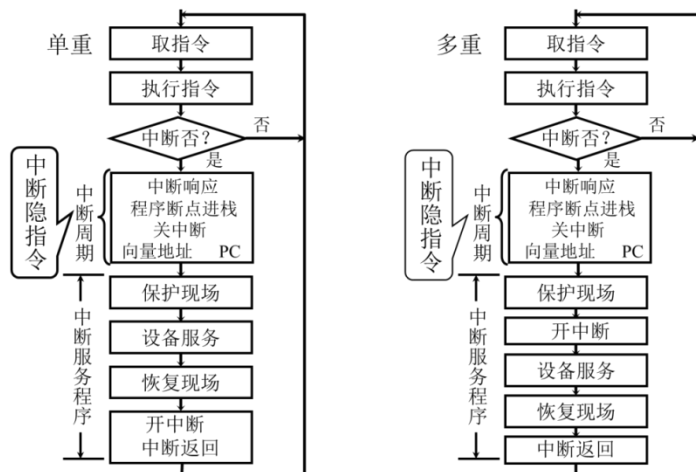


图 7.6 单重中断和多重中断的服务程序流程

参考答案：A

6. (原书第 34 题) 通道是重要的 I/O 方式，其中适合连接大量终端及打印机的通道是 ()。

- A. 数组多路通道
- B. 选择通道
- C. 字节多路通道

【解析】通道有以下几种类型：

(1). 字节多路通道

字节多路通道(如图 7.7 所示)是一种简单的共享通道,用于连接与管理多台低速设备,以字节交叉方式传送信息。字节多路通道先选择设备 A,为其传送一个字节 A1;然后选择设备 B,传送字节 B1;再选择设备 C,传送字节 C1。再交叉地传送 A2、B2、C2……所以字节多路通道的功能好比一个多路开关,交叉(轮流)地接通各台设备。键盘终端或者打印机属于低速设备,适合字节多路通道这种 I/O 方式。

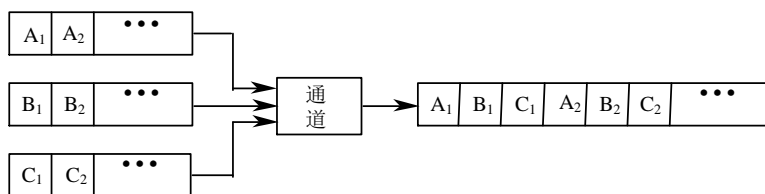


图 7.7 字节多路通道传送方式

(2). 选择通道

对于高速设备，字节多路通道显然是不合适的。选择通道（如图 7.8 所示）又称高速通道，在物理上它也可以连接多个设备，但这些设备不能同时工作，在一段时间内通道只能选择一台设备进行数据传送，此时该设备可以独占整个通道。因此，选择通道一次只能执行一个通道程序，只有当它与主存交换完信息后，才能再选择另一台外部设备并执行该设备的通道程序。选择通道先选择设备 A，成组连续地传送 A1A2……；当设备 A 传送完毕后，选择通道又选择设备 B，成组连续地传送 B1B2……；再选择设备 C，成组连续地传送 C1C2……。

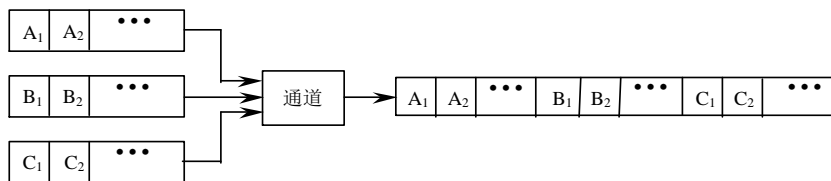


图 7.8 选择通道传送方式

(3). 数组多路通道

数组多路通道是把字节多路通道和选择通道的特点结合起来的一种通道结构。它的基本思想是：当某设备进行数据传送时，通道只为该设备服务；当设备在执行辅助操作时，通道暂时断开与这个设备的连接，挂起该设备的通道程序，去为其他设备服务。

数组多路通道有多个子通道，既可以执行多路通道程序，即像字节多路通道那样，所有子通道分时共享总通道，又可以用选择通道那样的方式成组地传送数据；既具有多路并行操作的能力，又具有很高的数据传输速率，使通道的效率充分得到发挥。

参考答案：C

二. 综合应用题部分

1. **(原书 第 1 题)**【2009 年 408 统考】某计算机的 CPU 主频为 500MHz，CPI 为 5（即执行每条指令平均需 5 个时钟周期）。假定某外设的数据传输率为 0.5 MB/s，采用中断方式与主机进行数据传送，以 32 位为传输单位。对应的中断服务程序包含 18 条指令，

中断服务的其他开销相当于 2 条指令的执行时间。请回答下列问题, 要求给出计算过程。

- (1). 在中断方式下, CPU 用于该外设 I/O 的时间占整个 CPU 时间的百分比是多少?
- (2). 当该外设的数据传输率达到 5MB/s 时, 改用 DMA 方式传送数据。假设每次 DMA 传送大小为 5000B, 且 DMA 预处理和后处理的总开销为 500 个时钟周期, 则 CPU 用于该外设 I/O 的时间占整个 CPU 时间的百分比是多少?(假设 DMA 与 CPU 之间没有访存冲突)。

【解析】

- (1). 在中断方式下, 每 32 位(4B)被中断一次, 故每秒中断

$$0.5\text{MB}/4\text{B} = 0.5 \times 10^6 / 4 = 12.5 \times 10^4 \text{ 次}$$

1 秒钟用于中断的时钟周期数为

$$(18+2) \times 12.5 \times 10^4 = 12.5 \times 10^6$$

因为 CPU 主频为 500MHz, 所以 CPU 用于该外设 I/O 的时间占整个 CPU 时间的百分比是

$$12.5 \times 10^6 / 500 \times 10^6 = 0.025 = 2.5\%$$

- (2). 在 DMA 方式下, 每秒进行 DMA 操作 $5\text{MB}/5000\text{B} = 10^3$ 次, 1 秒钟之内用于 DMA 操作的时钟周期数为 500×10^3 个。

因为 CPU 主频为 500MHz, 故在 DMA 方式下, CPU 用于该外设 I/O 的时间占整个 CPU 时间的百分比是: $500 \times 10^3 / 500 \times 10^6 = 10^{-3} = 0.1\%$ 。

2. **(原书 第 11 题)** 在程序查询方式输入输出的系统中, 有三个不同的外围设备。假定一个查询操作需要 100 个时钟周期, CPU 的时钟频率为 50MHz。求 CPU 在以下三种情况下为输入输出查询所花费的时间比率(百分率), 假定进行足够的查询以避免数据丢失。

- (1). 鼠标器必须在每秒钟进行 30 次查询。
- (2). 软盘与 CPU 的数据传输以 16 位为单位进行, 数据传输速率为 50KB/s。
- (3). 硬盘传输数据以 32 位的字为单位, 传输速率为 2MB/s。
- (4). 假设采用中断方法进行控制, 每次传输仍以 16 位的单位以 50KB/s 的速率进行, 每次传输的开销(包括中断处理)为 100 个时钟周期。求 CPU 为传输软盘数据花费的时间比率。

【解析】

- (1). 对于鼠标器, 每秒查询 30 次, 则每秒的时钟周期数为:

$$30 \times 100 = 3000 \text{ 时钟周期/秒}$$

占用的 CPU 时间比率为：

$$3000 / (50 \times 1000000) = 0.006\%$$

(2). 对于软盘，每秒查询的时钟周期数为：

$$50\text{KB}/2\text{B} \times 100 = 2560000 \text{ 时钟周期/秒}$$

占用的 CPU 时间比率为：

$$2560000 / (50 \times 1000000) = 5.12\%$$

(3). 对于硬盘，每秒查询的时钟周期数为：

$$2\text{MB}/4\text{B} \times 100 = 52.4 \times 10^6 \text{ 时钟周期/秒}$$

占用的 CPU 时间比率为：

$$52.4 \times 10^6 / (50 \times 10^6) \approx 105\%$$

(4). CPU 被中断的速率为：

$$50\text{KB}/2\text{B} = 25600 \text{ 次/秒}$$

传输占用的周期数为：

$$25600 \times 100 = 2560000 \text{ 时钟周期/秒}$$

占用的 CPU 时间比率为：

$$2560000 / (50 \times 1000000) = 5.12\%$$

3. **(原书 第 12 题)** 在一个 8 级中断系统中，硬件中断响应从高到低的优先顺序是：L1 > L2 > L3 > L4 > L5 > L6 > L7 > L8，现在要通过设置中断屏蔽字的方法将中断处理的优先顺序改变为：L1 > L5 > L8 > L3 > L2 > L4 > L6 > L7。

- (1). 应如何设置中断屏蔽字？（请在右面所给的表中填写相应内容，并规定某位为“1”表示中断屏蔽，为“0”表示中断开放。）
- (2). 如果 CPU 在执行一个应用程序的某一时刻 t 有 L5、L6 和 L7 三级中断请求同时提出（见下图），在该 L6 级中断尚未处理完时，又有 L8 级中断请求到达，在处理该 L8 级中断过程中，又有 L2 级中断请求提出。试在下图中画出 CPU 运行程序的轨迹。

【解析】

(1). 中断屏蔽字设置如表 7.1 所示。

表 7.1 中断系统的中断屏蔽字

中断处理程序	中断屏蔽字							
	L1级	L2级	L3级	L4级	L5级	L6级	L7级	L8级
L1级	1	1	1	1	1	1	1	1
L2级	0	1	0	1	0	1	1	0
L3级	0	1	1	1	0	1	1	0
L4级	0	0	0	1	0	1	1	0
L5级	0	1	1	1	1	1	1	1
L6级	0	0	0	0	0	1	1	0
L7级	0	0	0	0	0	0	1	0
L8级	0	1	1	1	0	1	1	1

(2). 依题意, CPU 的执行轨迹如图 7.9 所示。

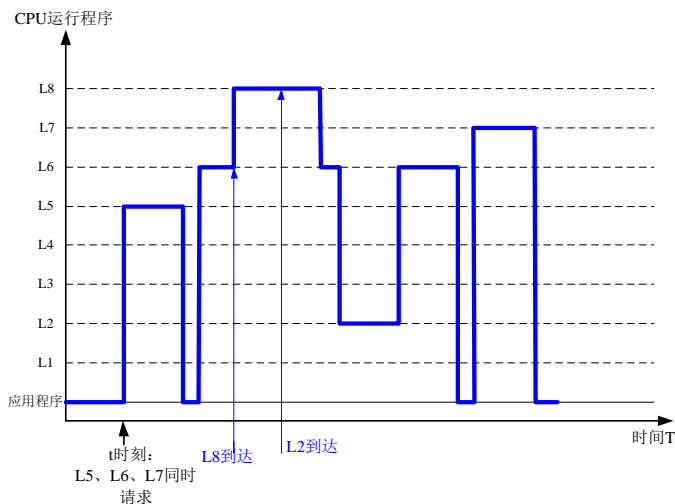


图 7.9 CPU 的运行轨迹

本章到此就结束了, 请问您有什么疑问吗? 任何问题, 欢迎您与我们作者进行交流!



梦享团队微信号



梦享团队官方微信公众账号



梦享团队新浪微博

“梦享考研系列”辅导书——答疑解惑

尊敬的读者朋友！

您好！首先，欢迎您使用我梦享团队编写的“梦享考研系列”辅导书，我们衷心地感谢您的支持！

让每一个考研人圆梦，是我们团队的梦想；默默陪伴着您走过每一个寻梦的日日夜夜，做您不离不弃的朋友，是我们的愿望。

宝剑锋从磨砺出，梅花香自苦寒来。每一个甜蜜的果实，都在挥洒汗水之后才能摘取到。对于每一个考生来说，一本绝佳的考研辅导书，就像一把利剑，拥有这把利剑，我们能够百战百胜，所向披靡。我们深知我们团队的能力有限，但我们一直希望，我们这套书，就是大家手中的那一把利剑！为此，我们团队特意向各位考生征求好的教材写法，好的题目解析方法等，希望把大家的精华汇聚到一本书里，为更多的考生提供更好的辅导书。

除了“梦享考研系列”辅导书之外，梦享团队致力于给大家更好的考研服务。为了给广大读者提供一个答疑和交流的平台，梦享团队开通了微信号、微信公众号和新浪微博账号等三种平台。欢迎大家来跟我们交流，一起成长。



梦享团队微信号



梦享团队官方微信公众号



梦享团队新浪微博

因为有你，所以有梦享！

梦享团队祝愿每一个考研人梦想成真！

参考文献

- [1] 汤子瀛. 计算机操作系统[M]. 西安: 西安电子科技大学出版社, 2001.
- [2] Tanenbaum A.S. 现代操作系统[M]. 北京: 机械工业出版社, 2009.
- [3] 翔高教育. 计算机学科专业基础综合习题精编[M]. 上海: 复旦大学出版社, 2010.
- [4] 崔巍, 等. 计算机学科专业基础综合辅导讲义[M]. 北京: 原子能出版社, 2011.
- [5] 严蔚敏. 数据结构(C语言版)[M]. 北京: 清华大学出版社, 2007.
- [6] 严蔚敏, 等. 数据结构题集(C语言版)[M]. 北京: 清华大学出版社, 1999.
- [7] 谢希仁. 计算机网络(第5版)[M]. 北京: 电子工业出版社, 2008.
- [8] 谢希仁. 计算机网络释疑与习题解答[M]. 北京: 电子工业出版社, 2011.
- [9] 唐朔飞. 计算机组成原理(第2版)[M]. 北京: 高等教育出版社, 2008.
- [10] 唐朔飞. 计算机组成原理学习指导与习题解答(第2版)[M]. 北京: 高等教育出版社, 2012.
- [11] 白中英. 计算机组成原理(第四版)[M]. 北京: 科学出版社, 2008.
- [12] 蒋本珊. 计算机组成原理(第三版)[M]. 北京: 清华大学出版社, 2013.