

# 《计算机组成原理》的试题、习题、复习资料

## “计算机组成原理”资料

### 第1章 概论

一、名词解释：（第一章的名称解释是考试的重点）

- 1.主机：由 CPU、存储器与 I/O 接口合在一起构成的处理系统称为主机。
- 2.CPU：中央处理器，是计算机的核心部件，由运算器和控制器构成。
- 3.运算器：计算机中完成运算功能的部件，由 ALU 和寄存器构成。
- 4.ALU：算术逻辑运算单元，负责执行各种算术运算和逻辑运算。
- 5.外围设备：计算机的输入输出设备，包括输入设备，输出设备和外存储设备。
- 6.数据：编码形式的各种信息，在计算机中作为程序的操作对象。
- 7.指令：是一种经过编码的操作命令，它指定需要进行的操作，支配计算机中的信息传递以及主机与输入输出设备之间的信息传递，是构成计算机软件的基本元素。
- 8.透明：在计算机中，从某个角度看不到的特性称该特性是透明的。
- 9.位：计算机中的一个二进制数据代码，计算机中数据的最小表示单位。
- 10.字：数据运算和存储的单位，其位数取决于具体的计算机。
- 11.字节：衡量数据量以及存储容量的基本单位。1 字节等于 8 位二进制信息。
- 12.字长：一个数据字中包含的位数，反应了计算机并行计算的能力。一般为 8 位、16 位、32 位或 64 位。
- 13.地址：给主存储器中不同的存储位置指定的一个二进制编号。
- 14.存储器：计算机中存储程序和数据部件，分为内存和外存。
- 15.总线：计算机中连接功能单元的公共线路，是一束信号线的集合，包括数据总线、地址总线和控制总线。
- 16.硬件：由物理元器件构成的系统，计算机硬件是一个能够执行指令的设备。
- 17.软件：由程序构成的系统，分为系统软件和应用软件。
- 18.兼容：计算机部件的通用性。
- 19.软件兼容：一个计算机系统上的软件能在另一个计算机系统上运行，并得到相同的结果，则称这两个计算机系统是软件兼容的。
- 20.程序：完成某种功能的指令序列。
- 21.寄存器：是运算器中若干个临时存放数据的部件，由触发器构成，用于存储最频繁使用的数据。
- 22.容量：是衡量容纳信息能力的指标。
- 23.主存：一般采用半导体存储器件实现，速度较高。成本高且当电源断开时存储器的内容会丢失。
- 24.辅存：一般通过输入输出部件连接到主存储器的外围设备，成本低，存储时间长。
- 25.操作系统：主要的系统软件，控制其它程序的运行，管理系统资源并且为用户提供操作界面。
- 26.汇编程序：将汇编语言程序翻译成机器语言程序的计算机软件。
- 27.汇编语言：采用文字方式（助记符）表示的程序设计语言，其中大部分指令和机器

语言中的指令一一对应，但不能被计算机的硬件直接识别。

28.编译程序：将高级语言程序转换成机器语言程序的计算机软件。

29.解释程序：解释执行高级语言程序的计算机软件，解释并立即执行源程序的语句。

30.系统软件：计算机系统的一部分，进行命令解释、操作管理、系统维护、网络通信、软件开发和输入输出管理的软件，与具体的应用领域无关。

31.应用软件：完成应用功能的软件，专门为解决某个应用领域中的具体任务而编写。

32.指令流：在计算机的存储器与 CPU 之间形成的不断传递的**指令序列**。从存储器流向控制器。

33.数据流：在计算机的存储器与 CPU 之间形成的不断传递的**数据序列**。存在于运算器与存储器以及输入输出设备之间。

34.接口：计算机主机与外围设备之间传递数据与控制信息的电路。计算机可以与多种不同的外围设备连接，因而需要有多种不同的输入输出接口。

## 二、填空题：

系统软件主要包括：\_\_操作系统\_\_和\_\_语言处理程序\_\_及诊断程序等。

（2005 年）18.构成中央处理器的两大部件是\_\_运算器\_\_和\_\_控制器\_\_。

# 第 2 章 数据编码和数据运算

## 一、名词解释：

基数：在浮点数据编码中，对阶码所代表的指数值的数据，在计算机中是一个常数，不用代码表示。

移码：带符号数据表示方法之一，符号位用 1 表示正，0 表示负，其余位与补码相同。

溢出：指数的值超出了数据编码所能表示的数据范围。

偶校验码：让编码组代码中 1 的个数为偶数，违反此规律为校验错。

1.原码：带符号数据表示方法之一，一个符号位表示数据的正负，0 代表正号，1 代表负号，其余的代表数据的绝对值。

2.补码：带符号数据表示方法之一，正数的补码与原码相同，负数的补码是将二进制位按位取反后在最低位上加 1。

3.反码：带符号数据的表示方法之一，正数的反码与原码相同，负数的反码是将二进制位按位取反

4.阶码：在浮点数据编码中，表示小数点的位置的代码。

5.尾数：在浮点数据编码中，表示数据有效值的代码。

6.机器零：在浮点数据编码中，阶码和尾数都全为 0 时代表的 0 值。

7.上溢：指数的绝对值太大，以至大于数据编码所能表示的数据范围。

8.下溢：指数的绝对值太小，以至小于数据编码所能表示的数据范围。

9.规格化数：在浮点数据编码中，为使浮点数具有唯一的表示方式所作的规定，规定尾数部分用纯小数形式给出，而且尾数的绝对值应大于  $1/R$ ，即小数点后的第一位不为零。

10.Booth 算法：一种带符号数乘法，它采用相加和相减的操作计算补码数据的乘积。

11.海明距离：在信息编码中，两个合法代码对应位上编码不同的位数。

12.冯·诺依曼舍入法：浮点数据的一种舍入方法，在截去多余位时，将剩下数据的最低位

置 1.

13.检错码：能够发现某些错误或具有自动纠错能力的数据编码。

14.纠错码：能够发现某些错误并且具有自动纠错能力的数据编码。

15.奇校验码：让编码组代码中 1 的个数为奇数，违反此规律为校验错。

16.海明码：一种常见的纠错码，能检测出两位错误，并能纠正一位错误。

17.循环码：一种纠错码，其合法码字移动任意位后的结果仍然是一个合法码字。

18.桶形移位器：可将输入的数据向左、向右移动 1 位或多位的移位电路。

## 二、数制度的转换：

(2001 年) 1.若十进制数据为 137.5 则其八进制数为 ( )。

A.89.8

B.211.4

C.211.5

D.1011111.101

「分析」：十进制数转化为八进制数时，整数部分和小数部分要用不同的方法来处理。整数部分的转化采用除基取余法：将整数除以 8，所得余数即为八进制数的个位上数码，再将商除以 8，余数为八进制十位上的数码……如此反复进行，直到商是 0 为止；对于小数的转化，采用乘基取整法：将小数乘以 8，所得积的整数部分即为八进制数十分位上的数码，再将此积的小数部分乘以 8，所得积的整数部分为八进制数百分位上的数码，如此反复……直到积是 0 为止。此题经转换后得八进制数为 211.40。「答案」：B

(2002 年) 1.若十进制数为 132.75，则相应的十六进制数为 ( )。

A.21.3

B.84.c

C.24.6

D.84.6

「分析」：十进制数转化为十六进制数时，采用除 16 取余法；对于小数的转化，采用乘 16 取整法：将小数乘以 16，所得积的整数部分转换为十六进制。此题经转换后得十六进制数为 84.c。「答案」：B

(2003 年) 14.若十六进制数为 A3.5，则相应的十进制数为 ( )。

A.172.5

B.179.3125

C.163.3125

D.188.5

「分析」：将十六进制数 A3.5 转换为相应的十进制数，可采用乘幂相加法完成，即： $10 \times 16^1 + 3 \times 16^0 + 5/16^1 = 163.3125$ 。「答案」：C

(2004 年) 1.若二进制数为 1111.101，则相应的十进制数为 ( )。

A.15.625

B.15.5

C.14.625

D.14.5

「分析」：将二进制数 1111.101 转换为相应的十进制数，可采用乘幂相加法完成，即： $1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1/(2^1) + 1/(2^3) = 15.625$ 。「答案」：A

(2005 年) 2.若十六进制数为 B5.4，则相应的十进制数为 ( )。

A.176.5

B.176.25

C.181.25

D.181.5

「分析」：将十六进制数 B5.4 转换为相应的十进制数，可采用乘幂相加法完成，即： $11 \times 16^1 + 5 \times 16^0 + 4/16^1 = 181.25$ 。「答案」：C

(1) 十进制转换为二进制。方法：整数部分除 2 取余，小数部分乘 2 取整。

(2) 二进制转换为八进制

方法：以小数点为界，整数部分从右向左每三位分为一组，最左端不够三位补零；小数部分从左向右每三位分为一组，最右端不够三位补零；最后将每组转换位一位八进制数。

(3) 二进制转换为十六进制

方法：以小数点为界，整数部分从右向左每四位分为一组，最左端不够四位补零；小数部分从左向右每四位分为一组，最右端不够四位补零；最后将每小组转换位一位十六进制数。

三、数据编码：

定点数编码：

(2000 年) 2. 如果  $X$  为负数，由  $[X]$  补求  $[-X]$  补是将 ( )。

- A.  $[X]$  补各值保持不变
- B.  $[X]$  补符号位变反，其它各位不变
- C.  $[X]$  补除符号位外，各位变反，末尾加 1
- D.  $[X]$  补连同符号位一起各位变反，末尾加 1

「分析」：不论  $X$  是正数还是负数，由  $[X]$  补求  $[-X]$  补的方法是对  $[X]$  补求补，即连同符号位一起按位取反，末尾加 1。「答案」：D

(2001 年) 2. 若  $[x]$  补 = 0.1101010，则  $[x]$  原 = ( )。

- A. 1.0010101      B. 1.0010110      C. 0.0010110      D. 0.1101010

「分析」：正数的补码与原码相同，负数的补码是用正数的补码按位取反，末位加 1 求得。此题中  $X$  补为正数，则  $X$  原与  $X$  补相同。「答案」：D

(2002 年) 2. 若  $x=1011$ ，则  $[x]$  补 = ( )。

- A. 01011      B. 1011      C. 0101      D. 10101

「分析」： $x$  为正数，符号位为 0，数值位与原码相同，结果为 01011。「答案」：A

(2003 年) 8. 若  $[X]$  补 = 1.1011，则真值  $X$  是 ( )。

- A. -0.1011      B. -0.0101      C. 0.1011      D. 0.0101

「分析」： $[X]$  补 = 1.1011，其符号位为 1，真值为负；真值绝对值可由其补码经求补运算得到，即按位取后得 0.0100 再末位加 1 得 0.0101，故其真值为 -0.0101。「答案」：B

(2004 年) 13. 设有二进制数  $x = -1101110$ ，若采用 8 位二进制数表示，则  $[X]$  补 ( )。

- A. 11101101      B. 10010011      C. 00010011      D. 10010010

「分析」： $x = -1101110$  为负数，负数的补码是将二进制位按位取反后在最低位上加 1，故  $[x]$  补 = 10010010。「答案」：D

(2005 年) 1. 若  $[X]$  补 = 0.1011，则真值  $X =$  ( )。

- A. 0.1011      B. 0.0101      C. 1.1011      D. 1.0101

「分析」： $[X]$  补 = 0.1011，其符号位为 0，真值为正；真值就是 0.1011。「答案」：A

(2001) 3. 若定点整数 64 位，含 1 位符号位，补码表示，则所能表示的绝对值最大负数为 ( )。

- A.  $-2^{64}$       B.  $-(2^{64}-1)$       C.  $-2^{63}$       D.  $-(2^{63}-1)$

「分析」：字长为 64 位，符号位为 1 位，则数值位为 63 位。当表示负数时，数值位全 0 为负绝对值最大，为  $-2^{63}$ 。「答案」：C

(2002 年) 3. 某机字长 8 位, 含一位数符, 采用原码表示, 则定点小数所能表示的非零最小正数为 ( )

- A.  $2^{-9}$       B.  $2^{-8}$       C.  $1^{-}$       D.  $2^{-7}$

[分析]: 求最小的非零正数, 符号位为 0, 数值位取非 0 中的原码最小值, 此 8 位数据编码为: 00000001, 表示的值是:  $2^{-7}$ . 「答案」: D

(2003 年) 13.  $n+1$  位的定点小数, 其补码表示的是 ( )。

- A.  $-1 \leq x \leq 1-2^{-n}$       B.  $-1 < x \leq 1-2^{-n}$   
C.  $-1 \leq x < 1-2^{-n}$       D.  $-1 < x < 1-2^{-n}$

「分析」: 编码方式    最小值编码    最小值    最大值编码    最大值    数值范围

$n+1$  位无符号定点整数    000...000 0 111...111     $2^{n+1}-1$      $0 \leq x \leq 2^{n+1}-1$

$n+1$  位无符号定点小数    0.00...000 0 0.11...111     $1-2^{-n}$      $0 \leq x \leq 1-2^{-n}$

$n+1$  位定点整数原码    1111...111     $-2^{n+1}$  0 111...111     $2^n-1$      $-2^{n+1} \leq x \leq 2^n-1$

$n+1$  位点定小数原码    1.111...111     $-1+2^{-n}$  0.111...111     $1-2^{-n}$      $-1+2^{-n} \leq x \leq 1-2^{-n}$

$n+1$  位定点整数补码    1000...000     $-2^n$  0 111...111     $2^n-1$      $-2^n \leq x \leq 2^n-1$

$n+1$  位点定小数补码    1.000...000     $-1$  0.111...111     $1-2^{-n}$      $-1 \leq x \leq 1-2^{-n}$

$n+1$  位定点整数反码    1000...000     $-2^{n+1}$  0 111...111     $2^n-1$      $-2^{n+1} \leq x \leq 2^n-1$

$n+1$  位点定小数反码    1.000...000     $-1+2^{-n}$  0.111...111     $1-2^{-n}$      $-1+2^{-n} \leq x \leq 1-2^{-n}$

$n+1$  位定点整数移码    0000...000     $-2^n$  1111...111     $2^n-1$      $-2^n \leq x \leq 2^n-1$

$n+1$  位点定小数移码    小数没有移码定义 「答案」: A

(2004 年) 12. 定点小数反码  $[x]$  反  $= x_0.x_1 \cdots x_n$  表示的数值范围是 ( )。

- A.  $-1+2^{-n} < x \leq 1-2^{-n}$       B.  $-1+2^{-n} \leq x < 1-2^{-n}$   
C.  $-1+2^{-n} \leq x \leq 1-2^{-n}$       D.  $-1+2^{-n} < x < 1-2^{-n}$     答案: C

(2005 年) 3. 一个  $n+1$  位整数原码的数值范围是 ( )。

- A.  $-2^{n+1} < x < 2^{n+1}$       B.  $-2^{n+1} \leq x < 2^{n+1}$   
C.  $-2^{n+1} < x \leq 2^{n+1}$       D.  $-2^{n+1} \leq x \leq 2^{n+1}$     答案: D

浮点数编码:

(2002 年) 4. 设某浮点数共 12 位。其中阶码含 1 位阶符共 4 位, 以 2 为底, 补码表示; 尾数含 1 位数符共 8 位, 补码表示, 规格化。则该浮点数所能表示的最大正数是 ( )。

- A. 27      B. 28      C.  $28-1$       D.  $27-1$

「分析」: 为使浮点数取正数最大, 可使尾数取正数最大, 阶码取正数最大。尾数为 8 位补码 (含符号位), 正最大为 01111111, 为  $1-2^{-7}$ , 阶码为 4 位补码 (含符号位), 正最大为 0111, 为 7, 则最大正数为:  $(1-2^{-7}) \times 2^7 = 27-1$ . 「答案」: D

四、定点数加减法:

(2001 年) 5. 若采用双符号位, 则发生正溢的特征是: 双符号位为 ( )。

- A. 00      B. 01      C. 10      D. 11

「分析」: 采用双符号位时, 第一符号位表示最终结果的符号, 第二符号位表示运算结果是否溢出。当第二位和第一位符号相同, 则未溢出; 不同, 则溢出。若发生正溢出, 则双符号位为 01, 若发生负溢出, 则双符号位为 10. 「答案」: B

2005 年) 4.若采用双符号位补码运算, 运算结果的符号位为 01, 则 ( )。

- A.产生了负溢出(下溢)
- B.产生了正溢出(上溢)
- C.结果正确, 为正数
- D.结果正确, 为负数

「分析」: 采用双符号位时, 第一符号位表示最终结果的符号, 第二符号位表示运算结果是否溢出。当第二位和第一位符号相同, 则未溢出; 不同, 则溢出。若发生正溢出, 则双符号位为 01, 若发生负溢出, 则双符号位为 10。「答案」: B

两种判断溢出的方法:

- (1) 两正数相加结果为负或两负数相加结果为正就说明产生了溢出
- (2) 最高位进位和次高位进位不同则发生了溢出

五、定点数的乘法:

(2001 年) 请用补码一位乘中的 Booth 算法计算  $x \cdot y = ?$   $x = 0101$ ,  $y = -0101$ , 列出计算过程。

「分析」: 补码一位乘法中的 Booth 算法是一种对带符号数进行乘法运算的十分有效的处理方法, 采用相加和相减的操作计算补码数据的乘积。做法是从最低位开始, 比较相邻的数位, 相等时不加不减, 只进行右移位操作; 不相等(01)时加乘数, 不相等(10 时)相减乘数, 再右移位; 直到所有位均处理完毕

「答案」:  $x = 0101$ ,  $x$  补 = 0101,  $-x$  补 = 1011,  $y = -0101$ ,  $y$  补 = 1011

循环	步骤	乘积 (R0 R1 P)
0	初始值	0000 1011 0
1	减 0101	1011 1011 0
	右移 1 位	1101 1101 1
2	无操作	1101 1101 1
	右移 1 位	1110 1110 1
3	加 0101	0011 1110 1
	右移 1 位	0001 1111 0
4	减 0101	1100 1111 0
	右移 1 位	1110 0111 1

所以结果为  $[x \cdot y]_{\text{补}} = 11101111$ , 真值为 -00011001, 十进制值为 -25.

(2002 年) 已知  $x = 0011$ ,  $y = -0101$ , 试用原码一位乘法求  $xy = ?$  请给出规范的运算步骤, 求出乘积。

「分析」: 原码一位乘法中, 符号位与数值位是分开进行计算的。运算结果的数值部分是乘数与被乘数数值位的乘积, 符号是乘数与被乘数符号位的异或。原码一位乘法的每一次循环的操作是最低位为 1, 加被乘数的绝对值后右移 1 位; 最低位为 0, 加 0 后右移 1 位。几位乘法就循环几次。

「答案」:  $[x]_{\text{原}} = 00011$ ,  $[y]_{\text{原}} = 10101$ ,  $|x| = 0011$ ,  $|y| = 0101$ , 结果的符号位  $10 = 1$

循环	步骤	乘积 (R0 R1)
0	初始值	0000 0101
1	加 0011	0011 0101
	右移 1 位	0001 1010
2	加 0	0001 1010

	右移 1 位	0000 1101
3	加 0011	0011 1101
	右移 1 位	0001 1110
4	加 0	0001 1110
	右移 1 位	0000 1111

所以结果为-00001111

(2003 年) 32. 用 Booth 算法计算  $7 \times (-3)$ 。要求写出每一步运算过程及运算结果。

(2004 年) 32. 用原码的乘法方法进行  $0110 \times 0101$  的四位乘法。要求写出每一步运算过程及运算结果。

(2005 年) 32. 用原码加减交替一位除法进行  $7 \div 2$  运算。要求写出每一步运算过程及运算结果。

「答案」: 7 的原码 0111, 3 的原码 0011, 结果符号是  $0 \oplus 0 = 0$

原码加减交替除法求  $x/y$  的分步运算过程。

循环	步骤	余数 (R0 R1)
0	初始值	0000 0111
	左移, 商 0	0000 1110
1	减 0011	1101 1110
	加 0011, 商 0	0000 1110 (0)
	左移 1 位	0001 1100
2	减 0011	1110 1100
	加 0011, 商 0	0001 1100 (0)
	左移 1 位	0011 1000
3	减 0011	0000 1000
	商 1	0000 1000 (1)
	左移 1 位	0001 0001
4	减 0011	1110 0001
	加 0011, 商 0	0001 0001 (0)
	左移 1 位	0010 0010
	R0 右移 1 位	0001 0010

所以, 商是 0010, 即 2; 余数是 0001, 即 1.

(2000 年) 1. 在原码一位乘中, 当乘数  $Y_i$  为 1 时, ( )。

- A. 被乘数连同符号位与原部分积相加后, 右移一位
- B. 被乘数绝对值与原部分积相加后, 右移一位
- C. 被乘数连同符号位右移一位后, 再与原部分积相加
- D. 被乘数绝对值右移一位后, 再与原部分积相加

「分析」: 原码一位乘法中, 符号位与数值位是分开进行计算的。运算结果的数值部分是乘数与被乘数数值位的乘积, 符号是乘数与被乘数符号位的异或。数值位相乘时, 当乘数某位为 1 时, 将被乘数绝对值与原部分积相加后, 右移一位。「答案」: B

(2001 年) 7. 原码乘法是 ( A )。

- A. 先取操作数绝对值相乘, 符号位单独处理

- B.用原码表示操作数，然后直接相乘
- C.被乘数用原码表示，乘数取绝对值，然后相乘
- D.乘数用原码表示，被乘数取绝对值，然后相乘

「分析」：原码一位乘法中，符号位与数值位是分开进行计算的。运算结果的数值部分是乘数与被乘数数值位的乘积，符号是乘数与被乘数符号位的异或。「答案」：A

8.原码加减交替除法又称为不恢复余数法，因此（c）。

- A.不存在恢复余数的操作
- B.当某一步运算不够减时，做恢复余数的操作
- C.仅当最后一步余数为负时，做恢复余数的操作
- D.当某一步余数为负时，做恢复余数的操作

「分析」：在用原码加减交替法作除法运算时，商的符号位是由除数和被除数的符号位异或来决定的，商的数值是(湖北自考网)由除数、被除数的绝对值通过加减交替运算求得的。由于除数、被除数取的都是绝对值，那么最终的余数当然应是正数。如果最后一步余数为负，则应将该余数加上除数，将余数恢复为正数，称为恢复余数。「答案」：C

（2002 年）5.原码乘法是指（ B）。

- A.用原码表示乘数与被乘数，直接相乘
- B.取操作数绝对值相乘，符号位单独处理
- C.符号位连同绝对值一起相乘
- D.取操作数绝对值相乘，乘积符号与乘数符号相同答案：B

六、逻辑运算：

（2005 年）5.已知一个 8 位寄存器的数值为 11001010，将该寄存器小循环左移一位后，结果为（ ）。

- A.01100101
- B.10010100
- C.10010101
- D.01100100

「分析」：

移位种类 运算规则

算术左移 每位左移一位，最右位移入 0，最高位移出进入标志寄存器 C 位

算术右移 每位右移一位，最高位符号复制，最低位移出进入标志寄存器 C 位

逻辑左移 每位左移一位，最右位移入 0，最高位移出进入标志寄存器 C 位

逻辑右移 每位右移一位，最右位移入 0，最低位移出进入标志寄存器 C 位

小循环左移 每位左移一位，最高位进入最低位和标志寄存器 C 位

小循环右移 每位右移一位，最低位进入最高位和标志寄存器 C 位

大循环左移 每位左移一位，最高位进入标志寄存器 C 位，C 位进入最低位

大循环右移 每位右移一位，最低位进入标志寄存器 C 位，C 位进入最高位「答案」：C

七、浮点数运算：

（2001）6.浮点加减中的对阶的（A）。

- A.将较小的一个阶码调整到与较大的一个阶码相同
- B.将较大的一个阶码调整到与较小的一个阶码相同
- C.将被加数的阶码调整到与加数的阶码相同



D.将加数的阶码调整到与被加数的阶码相同

「分析」：浮点加减法中的对阶是向较大阶码对齐，即将较小的一个阶码调整到与较大的一个阶码相同。「答案」：A

例：用浮点数运算步骤对  $56+5$  进行二进制运算，浮点数格式为 1 位符号位、5 位阶码、10 位尾码，基数为 2.

「答案」：

$$56_{10}=111000_2=0.111000\times 2^6 \quad 5_{10}=101_2=0.101\times 2^3$$

① 对阶： $0.101\times 2^3=0.000101\times 2^6$

② 尾数相加： $0.111000+0.000101=0.111101$

③ 规格化结果： $0.111101\times 2^6$

④ 舍入：数据已适合存储，不必舍入

⑤ 检查溢出：数据无溢出。

## 第 3 章 存储系统

一、名词解释：

1.RAM：随机访问存储器，能够快速方便的访问地址中的内容，访问的速度与存储位置无关。

2.ROM：只读存储器，一种只能读取数据不能写入数据的存储器。

3.SRAM：静态随机访问存储器，采用双稳态电路存储信息。

4.DRAM：动态随机访问存储器，利用电容电荷存储信息。

5.EDO DRAM：增强数据输出动态随机访问存储，采用快速页面访问模式并增加了一个数据锁存器以提高数据传输速率。

6.PROM：可编程的 ROM，可以被用户编程一次。

7.EPROM：可擦写可编程的 ROM，可以被用户编程多次。靠紫外线激发浮置栅上的电荷以达到擦除的目的。

8.EEPROM：电可擦写可编程的 ROM，能够用电子的方法擦除其中的内容。

9.SDRAM：同步型动态随机访问存储器，在系统时钟控制下进行数据的读写。

10.快闪存储器：一种非挥发性存储器，与 EEPROM 类似，能够用电子的方法擦除其中的内容。

11.相联存储器：一种按内容访问的存储器，每个存储单元有匹配电路，可用于在 cache 中查找数据。

12.多体交叉存储器：由多个相互独立、容量相同的存储体构成的存储器，每个存储体独立工作，读写操作重叠进行。

13.访存局部性：CPU 对存储空间的 90% 的访问局限于存储空间的 10% 的区域中，而另外 10% 的访问则分布在 90% 的区域中。

14.直接映象：cache 的一种地址映象方式，一个主存块只能映象到 cache 中的唯一一个指定块。

15.全相联映象：cache 的一种地址映象方式，一个主存块可映象到任何 cache 块。

16.组相联映象：cache 的一种地址映象方式，将存储空间分成若干组，各组之间用直接

**映象，组内各块之间用全相联映象。**

17.全写法（写直达法）：cache 命中时的一种更新策略，写操作时将数据既写入 cache 又写入主存，但块变更时不需要将调出的块写回主存。

18.写回法：cache 命中时的一种更新策略，写 cache 时不写主存，而当 cache 数据被替换出去时才写回主存。

19.按写分配：cache 不命中时的一种更新策略，写操作时把对应的数据块从主存调入 cache。

20.不按写分配：cache 不命中时的一种更新策略，写操作时该地址的数据块不从主存调入 cache。

21.虚拟存储器：为了扩大容量，把辅存当作主存使用，所需要的程序和数据由辅助的软件和硬件自动地调入主存，对用户来说，好像机器有一个容量很大的内存，这个扩大的存储空间称为虚拟存储器

22.层次化存储体系：把各种不同存储容量、不同访问速度、不同成本的存储器件按层次构成多层的存储器，并通过软硬件的管理将其组成统一的整体，使所存储的程序和数据按层次分布在各种存储器件中。

23.访问时间：从启动访问存储器操作到操作完成的时间。

24.访问周期时间：从一次访问存储的操作到操作完成后可启动下一次操作的时间。

25.带宽：存储器在连续访问时的数据吞吐率。

26.段式管理：一种虚拟存储器的管理方式，把虚拟存储空间分成段，段的长度可以任意设定，并可以放大或缩小。

27.页式管理：一种虚拟存储器的管理方式，把虚拟存储空间和实际存储空间等分成固定容量的页，需要时装入内存，各页可装入主存中不同的实际页面位置。

28.段页式管理：一种虚拟存储器的管理方式，将存储空间逻辑模块分成段，每段又分成若干页。

29.固件：固化在硬件中的固定不变的常用软件。

30.逻辑地址：程序员编程所用的地址以及 CPU 通过指令访问主存时所产生的地址。

31.物理地址：实际的主存储器的地址称为“真实地址”。

二、选择填空题：

5.动态半导体存储器的特点是（C）。

- A.在工作中存储器内容会产生变化
- B.每次读出后，需要根据原存内容重新写入一遍
- C.每隔一定时间，需要根据原存内容重新写入一遍
- D.在工作中需要动态地改变访存地址

「分析」：动态半导体存储器是利用电容存储电荷的特性记录信息，由于电容会放电，必须在电荷流失前对电容充电，即刷新。方法是每隔一定时间，根据原存内容重新写入一遍。

8.地址线 A15~A0（低），若选取用 16K×1 存储芯片构成 64KB 存储器则应由地址码译码产生片选信号。

「分析」：用 16K×1 芯片构成 64KB 的存储器，需要的芯片数量为： $(64K \times 8) / (16K \times 1) = 32$ ，每 8 片一组分成 4 组，每组按位扩展方式组成一个 16K×8 位的模块，4 个模块按字扩展方式构成 64KB 的存储器。存储器的容量为 64K=216，需要 16 位地址，选用 A15-A0 为地址线；每个模块的容量为 16K=214 需要 14 位地址，选用 A13-A0 为每个模块提供地址；A15、A14 通过 2-4 译码器对 4 个模块进行片选。「答案」：A15，A14

9.有静态 RAM 与动态 RAM 可供选择，在构成大容量主存时，一般就选择（ ）。

「分析」：静态 RAM 特点是存取速度快，单位价格（每字节存储空间的价格）较高；动态 RAM 则是存取速度稍慢，单位价格较低。所以考虑价格因素，在构成大容量的存储器时一般选择动态存储器。「答案」：动态 RAM

11.高速缓冲存储器 Cache 一般采取（ A ）。

A.随机存取方式 B.顺序存取方式 C.半顺序存取方式 D.只读不写方式

「分析」：Cache 是为提高存储器带宽而在主存储器和 CPU 之间增加的存储器，目的是用来存储使用频繁的数据和指令，存取方式应与主存储器相同，均为随机存取方式。「答案」：A

12.若存储周期 250ns，每次读出 16 位，则该存储器的数据传送率为（ ）。

A.  $4 \times 10^6$  字节 / 秒                      B. 4M 字节 / 秒  
C.  $8 \times 10^6$  字节 / 秒                      D. 8M 字节 / 秒

「分析」：存储周期 250ns，换算为  $250 \times 10^{-9}$  秒；每个存储周期可读出 16 位，为两个字节，则数据传送率为：2 字节 /  $(250 \times 10^{-9})$  秒，即  $8 \times 10^6$  字节 / 秒。「答案」：C

13.半导体静态存储器 SRAM 的存储原理是（ A ）。

A.依靠双稳态电路                      B.依靠定时刷新  
C.依靠读后再生                      D.信息不再变化

「分析」：半导体静态存储器 SRAM 是由双稳态电路构成，并依靠其稳态特性来保存信息；动态存储器 DRAM 是利用(湖北自考网)用电容器存储电荷的特性存储数据，依靠定时刷新和读后再生对信息进行保存，而 ROM 中的信息一经写入就不再变化。「答案」：A

6.一般来讲，直接映象常用在（ B ）。

A.小容量高速 Cache                      B.大容量高速 Cache  
C.小容量低速 Cache                      D.大容量低速 Cache

「分析」：直接映象的地址转换速度快，但块的冲突概率较高。在大容量高速 Cache 系统中使用直接映象方式，即可以发挥 Cache 的高速度，又可以减少块的冲突概率。「答案」：B

7.下列存储器中，（D）速度最快。

A.硬盘                      B.光盘                      C.磁带                      D.半导体存储器

「分析」：由于存储器原理和结构的不同，各种存储器的访问速度各不相同。以上存储器中访问速度由快到慢的顺序为：半导体存储器、硬盘、光盘、磁带。「答案」：D

15.在下列 Cache 替换算法中，一般说来哪一种比较好（d ）。

A.随机法                      B.先进先出法  
C.后进先出法                      D.近期最少使用法

「分析」：在 Cache 替换算法中，随机法是随机地确定替换的存储单元，先进先出法是替换最早调入的存储单元，它们都没有根据程序访存局部性原理，命中率较低；近期最少使用法比较正确地利用了程序访存局部性原理，替换出近期用得最少的存储块，命中率较高，是一种比较好的替换算法。而后进先出法不是 Cache 所使用的替换算法，此法在堆栈存储结构中使用。「答案」：D

8. 表示主存容量的常用单位为 (B)。

- A.数据块数      B.字节数      C.扇区数      D.记录项数

「分析」：表示主存容量的常用单位字节 B，是基本单位。此外还有 KB、MB、GB、

TB。「答案」：B

11. 存储器的随机访问方式是指 (D)。

- A.可随意访问存储器  
B.按随机文件访问存储器  
C.可对存储器进行读出与写入  
D.可按地址访问存储器任一编址单元，其访问时间相同且与地址无关

「分析」：存储器的随机访问方式是指可按地址访问存储器任一编址单元，其访问时间相同且与地址无关。「答案」：D

6.动态存储器的特点是 (D)。

- A.工作中存储内容会产生变化  
B.工作中需要动态改变访存地址  
C.工作中需要动态地改变供电电压  
D.需要定期刷新每个存储单元中存储的信息

「分析」：此题与 2000 年考题基本相同。动态半导体存储器是利用电容存储电荷的特性记录信息，由于电容会放电，必须在电荷流失前对电容充电，即刷新。方法是每隔一定时间，根据原存内容重新写入一遍。「答案」：D

7.组相联映象和全相联映象通常适合于 (A)。

- A.小容量 Cache      B.大容量 Cache  
C.小容量 ROM      D.大容量 ROM

「分析」：直接映象的地址转换速度快，但块的冲突概率较高。在大容量高速 Cache 系统中使用直接映象方式，即可以发挥 Cache 的高速度，又可以减少块的冲突概率。组相联映象和全相联映象速度较低，通常适合于小容量 Cache。

## 第 4 章 指令系统

一、名词解释：

1. **指令系统**：计算机中各种指令的集合，它反映了计算机硬件具备的基本功能。

2. 计算机指令：计算机硬件能识别并能直接执行操作的命令，描述一个基本操作。

3. 指令编码：将指令分成操作码和操作数地址码的几个字段来编码。

4. 指令格式：指定指令字段的个数，字段编码的位数和编码的方式。

5. **立即数**：在指令中直接给出的操作数。

6. 指令字长度：一个指令字所占有的位数。

7. **助记符**：用容易记忆的符号来表示指令中的操作码和操作数。

8. 汇编语言：采用文字方式（助记符）表示的程序设计语言，其中大部分指令和机器语言中的指令一一对应，但是不能被计算机的硬件直接识别。

9. 伪指令：汇编语言程序所提供的装入内存中的位置信息，表示程序段和数据段开始信

息及结束信息等。且不转换成 2 进制机器指令。

10. **大数端**：将数据的最低字节存储在最大地址位置的存储方式。

11. **小数端**：将数据的最低字节存储在最小地址位置的存储方式。

13. 系统指令：改变计算机系统的工作状态的指令。

14. 特权指令：改变执行特权的指令，用于操作系统对系统资源的控制。

15. 自陷指令：特殊的处理程序，又叫中断指令。

16. **寻址方式**：对指令的地址码进行编码，以得到操作数在存储器中的地址的方式。

17. 相对转移：转移到的目标指令的地址与当前指令的地址有关，是用当前指令的 PC 与一个偏移量相加，和为目标指令的 PC。

18. 绝对转移：转移到的目标指令的地址与当前指令的地址无关，指令中给定的目标地址即为目标指令的 PC。

19. **无条件转移**：一种转移指令类型，不管状态如何，一律进行转移操作。

20. 条件转移：一种转移指令类型，根据计算机中的状态决定是否转移。

21. **RISC**：精简指令系统计算机，即指令系统中的指令数量少，且指令功能相对简单。

22. **CISC**：复杂指令系统计算机，即指令系统中的指令数量多，且指令功能相对较强。

23. **堆栈**：数据的写入写出不需要地址，按先进后出的顺序读取数据的存储区。

## 二、选择填空题：

3. 在堆栈寻址中，设 A 为累加器，SP 为堆栈指示器，M<sub>sp</sub> 为 SP 指示的栈顶单元。如果进栈操作顺序是：(SP) - 1 → SP, (A) → M<sub>sp</sub>；那么出栈操作的顺序应是 (A)。

A. (M<sub>sp</sub>) → A, (SP) + 1 → SP

B. (SP) + 1 → SP, (M<sub>sp</sub>) → A

C. (SP) - 1 → SP, (M<sub>sp</sub>) → A

D. (M<sub>sp</sub>) → A, (SP) - 1 → SP

「分析」：堆栈是按特定顺序进行访问的存储区，其访问方式是后进先出，即先存入的数据后读出。对堆栈的操作有入栈和出栈两种，两者的操作完全相反，包括功能和顺序均相反。「答案」：A

6. 在按字节编址的存储器中，每个编址单元中存放 (B)。

A. 1 位

B. 8 位

C. 16 位

D. 32 位

「分析」：在按字节编址在存储器中，每个编址单元的容量为一个字节，一个字节由 8 位二进制数组成，一个字节存储单元可以存放 8 位二进制位。「答案」：B

4. 在 CPU 的状态寄存器中，常设置以下状态位：零标志位 (Z)，负标志位 (N)，( ) 和 ( )。

「分析」：在 CPU 中专门设置有一个存储计算机状态的寄存器，称为状态寄存器 SR，其中通常包括如下标志位：零标志位 (Z)、负标志位 (N)、溢出标志位 (V)、进位或借位标志位 (C) 等。

13. 如果说变址寻址方式主要是面向用户的，那么基址寻址一般是面向 (系统) 的。

「分析」：变址寻址方式是面向用户的，常用于访问字符串、向量数据结构和循环程序设计；而基址寻址方式是面向系统的，对由逻辑地址空间到物理地址空间的变换提供支持，用以解决程序在存储器中再定位和扩大寻址空间等问题。「答案」：系统

9.为了缩短指令中某个地址段的位数，有效的方法是采取（D）。

- A.立即寻址
- B.变址寻址
- C.间接寻址
- D.寄存器寻址

「分析」：由于计算机中寄存器的数量一般很少，采用寄存器寻址时可用少量的代码来指定寄存器，这样可以减少对应地址段的代码位数，也可减少整个指令的代码长度。「答案」：D

10.堆栈指针 SP 的内容是（B）。

- A.栈顶单元内容
- B.栈顶单元地址
- C.栈底单元内容
- D.栈底单元地址

「分析」：堆栈是按特定顺序进行访问的存储区，其访问方式是后进先出，即先存入的数据后读出。对堆栈的访问由堆栈指针寄存器 SP 控制，其内容为堆栈中栈顶单元的地址，即入栈时数据保存在 SP 指向的单元，出栈时将 SP 指向单元的内容取出。「答案」：B

8.采用直接寻址方式，则操作数在（A）中。

- A.主存
- B.寄存器
- C.直接存取存储器
- D.光盘

「分析」：直接寻址方式是指在指令中直接给出操作数在存储器中的地址，操作数在主存储器中，指令中的地址直接作为有效地址，对存储器进行访问即可取得操作数。「答案」：A

9.零地址指令的操作数一般隐含在（C）中。

- A.磁盘
- B.磁带
- C.寄存器
- D.光盘

分析:零地址指令只有操作码，没有操作数。这种指令有两种情况：一是无需操作数，另一种是操作数为隐含的，默认为操作数在寄存器中，指令可直接访问寄存器。「答案」：C

3.假设寄存器 R 中的数值为 200，主存地址为 200 和 300 的地址单元中存效的内容分别是 300 和 400，则什么方式下访问到的操作数为 200（D）。

- A.直接寻址 200
- B.寄存器间接寻址（R）
- C.存储器间接寻址（200）
- D.寄存器寻址 R

「分析」：直接寻址 200 的操作数为 300，寄存器间接寻址（R）的操作数 300，存储器间接寻址（200）的操作数为 400，寄存器寻址 R 的操作数为 200。「答案」：D

5.单地址指令（D）。

- A.只能对单操作数进行加工处理
- B.只能对双操作数进行加工处理
- C.无处理双操作数的功能
- D.既能对单操作数进行加工处理，也能在隐含约定另一操作数（或地址）时，对双操作数进行运算

「分析」：单地址指令既能对单操作数进行加工处理，也能对双操作数进行运算。当处理双操作数时，一个操作数在指令中给出，另一个操作数则是隐含约定的，例如堆栈操作指令中的入栈指令 PUSH，指令中只给出源操作数，而目的操作数则由计算机中的堆栈指针（SP）确定，在指令中不需要指定。「答案」：D

「分析」：单地址指令既能对单操作数进行加工处理，也能对双操作数进行运算。当处理双操作数时，一个操作数在指令中给出，另一个操作数则是隐含约定的，例如堆栈操作指令中的入栈指令 PUSH，指令中只给出源操作数，而目的操作数则由计算机中的堆栈指针（SP）确定，在指令中不需要指定。「答案」：D

14.反映计算机基本功能的是（C）。

- A.操作系统
- B.系统软件
- C.指令系统
- D.数据库系统



指令系统：计算机中各种指令的集合，它反映了计算机硬件具备的基本功能。答案：C

8.在大多数情况下，一条机器指令中是不直接用二进制代码来指定（ A ）。

- A.下一条指令的地址
- B.操作的类型
- C.操作数地址
- D.结果存放地址 答案：A

9.在存储器堆栈中，若栈底地址为A，SP指针初值为A-1，当堆栈采用从地址小的位置向地址大的位置生成时，弹出操作应是（A ）。

- A.先从堆栈取出数据，然后SP指针减1
- B.先从堆栈取出数据，然后SP指针加1
- C.SP指针先加1，然后从堆栈取出数据
- D.SP指针先减1，然后从堆栈取出数据

「分析」：堆栈是按特定顺序进行访问的存储区，其访问方式是后进先出，即先存入的数据后读出。对堆栈的(湖北自考网)访问由堆栈指针寄存器SP控制，当堆栈采用从地址小的位置向地址大的位置生成时，入栈操作是SP指针先加1，然后将数据存入堆栈，从堆栈取出弹出操作是先从堆栈取出数据，然后SP指针减1。「答案」：A

10.转移指令执行结束后，程序计数器PC中存放的是（C ）。

- A.该转移指令的地址 B.顺序执行的下条指令地址
- C.转移的目标地址 D.任意指令地址

「分析」：转移指令执行过程中，将转移指令所指的子程序的起始地址装入PC，因此转移指令执行结束后，程序计数器PC中存放的是转移的目标地址。「答案」：C

### 三、改错题：

3.在寄存器寻址方式中，指定寄存器中存放的是操作数地址。（2000）

「分析」：在寄存器间接寻址方式中，指定寄存器中存放的是操作数地址；而在寄存器寻址方式中，指定寄存器中存放着操作数。

「答案」：在寄存器寻址方式中，指定寄存器中存放着操作数。

1.在计算机中，各指令周期的时间长度是相同的。（2002）

「分析」：在计算机中，由于指令的种类不同，功能不同，执行每条指令时机器所进行的操作可能就不同，所需要的时间长短也可能不相同，所以各指令周期的时间长度不一定相同。

「答案」：一般说，由于各指令功能的不同，它们的指令周期有长有短，不一定相同。

22.转移指令执行结束后，目标地址可放在任意寄存器中。（2004年）

「分析」：转移指令执行过程中，将转移指令所指的子程序的起始地址装入PC，因此转移指令执行结束后，程序计数器PC中存放的是转移的目标地址。

「答案」：转移指令执行结束后，目标地址放在程序计数器PC中。

## 第5章 控制器

### 一、名词解释：

- 1.指令周期：从一条指令的启动到下一条指令的启动的间隔时间。
- 2.机器周期：指令执行中每一步操作所需的时间。
- 3.指令仿真：通过改变微程序实现不同机器指令系统的方式，使得在一种计算机上可以运行另一种计算机上的指令代码。
- 4.指令模拟：在一种计算机上用软件来解释执行另一种计算机的指令。
- 5.硬连线逻辑：一种控制器逻辑，用一个时序电路产生时间控制信号，采用组合逻辑电路实现各种控制功能。
- 6.微程序：完成某一个指令的一系列微指令结合。
- 7.微指令：控制器存储的控制代码，分为操作控制部分和顺序控制部分。
- 8.微操作：在微程序控制器中，执行部件接受微指令后所进行的操作。
- 9.微地址：微指令在控制存储器中的存储地址。
- 10.控制存储器：存放微程序的专用存储器。
- 11.相容性微操作：在同时或同一个 CPU 周期内可以并行执行的微操作。
- 12.相斥性微操作：不能在同时或不能在同一个 CPU 周期内并行执行的微操作。

### 二、选择题和填空题：

4.在取指周期中，是按照（ D ）的内容访问主存，以读取指令。

A.指令寄存器 IR

B.程序状态寄存器 PS

C.存储器数据寄存器 MDR

D.程序计数器 PC

「分析」：每一条指令的执行都是从取指令开始，需要对主存储器进行访问。程序计数器 PC 是用来存放将要读取并执行的指令在主存储器中的地址，对主存储器访问时所需要的地址由程序计数器 PC 来提供，即需要按程序计数器 PC 的内容来访问主存储器。「答案」：D

7.在微程序控制中，一个节拍中所需要的一组微命令，被编成一条（ ）

「分析」：控制部件通过控制总线向执行部件发出的控制命令称为微命令，它是计算机中最基本的、不可再分的命令单元。在一个节拍中，一组实现一定功能的微命令的组合构成一条微指令。「答案」：微指令

10.微程序存放在（ C ）。

A.主存中

B.堆栈中

C.只读存储器中

D.磁盘中

「分析」：微程序控制的基本思想是把指令执行所需的所有控制信号存放在存储器中，需要时从这个存储器中读取。由于每一条微指令执行时所发出的控制信号是事先设计好的，不需要改变，故此存放所有控制信号的存储器应为只读存储器，并将其集成到 CPU 内，称其为控制存储器。「答案」：C

11.在微程序控制方式中，机器指令和微指令的关系是（ B ）。

A.每一条机器指令由一条微指令来解释执行

B.每一条机器指令由一段（或一个）微程序来解释执行

C.一段机器指令组成的工作程序可由一条微指令来解释执行

D.一条微指令由若干条机器指令组成



「分析」：在微程序控制方式中，控制部件通过控制总线向执行部件发出的各种控制命令称为微命令，在一个 CPU 周期中，一组实现一定功能的微命令的组合构成一条微指令，有序的微指令序列构成一段微程序。微程序的作用是实现一条对应的机器指令，即每一条机器指令是由一段（或一个）微程序来解释执行的。「答案」：B

7.下列说法中，合理的是（ C）。

- A.执行各条指令的机器周期数相同，各机器周期的长度均匀
- B.执行各条指令的机器周期数相同，各机器周期的长度可变
- C.执行各条指令的机器周期数可变，各机器周期的长度均匀
- D.执行各条指令的机器周期数可变，各机器周期的长度可变

「分析」：机器周期是指令执行中每一步操作所需要的时间，一般以 CPU 中完成一个运算操作所需的时间作为机器周期的基本时间，其长度是均匀的，而各种指令的功能不同，因而各指令执行时所需的机器周期数是可变的。「答案」：C

10.微地址是指微指令（D）。

- A.在主存的存储位置
- B.在堆栈的存储位置
- C.在磁盘的存储位置
- D.在控制存储器的存储位置

「分析」：微程序控制的基本思想是：把指令执行所需要的所有控制信号存放在控制存储器中，需要时从这个存储器中读取，即把操作控制信号编成微指令，存放在控制存储器中。一条机器指令的功能通常用许多条微指令组成的序列来实现，这个微指令序列称为微程序。微指令在控制存储器中的存储位置称为微地址。「答案」：D

5.在微程序控制中，把操作控制信号编成（ A ）。

- A.微指令
- B.微地址
- C.操作码
- D.程序

「分析」：微程序控制的基本思想是：把指令执行所需要的所有控制信号存放在控制存储器中，需要时从这个存储器中读取，即把操作控制信号编成微指令，存放在控制存储器中。一条机器指令的功能通常用许多条微指令组成的序列来实现，这个微指令序列称为微程序。微指令在控制存储器中的存储位置称为微地址。「答案」：A

6.从一条指令的启动到下一条指令的启动的间隔时间称为（D ）。

- A.时钟周期
- B.机器周期
- C.工作周期
- D.指令周期

「分析」：指令周期：从一条指令的启动到下一条指令的启动的间隔时间。机器周期：指令执行中每一步操作所需的时间，又称 CPU 周期。时钟周期：计算机主频周期。「答案」：D

11.通常，微指令的周期对应一个（ C）。

- A.指令周期
- B.主频周期
- C.机器周期
- D.工作周期

「分析」：指令周期：从一条指令的启动到下一条指令的启动的间隔时间。机器周期：指令执行中每一步操作所需的时间，又称 CPU 周期。时钟周期：计算机主频周期。微指令周期等于读出一条微指令加上执行该微指令的所需时间。通常微指令周期与指令的机器周期相等。「答案」：C

19.在微程序控制器中，控制存储器由（ 只读存储器）构成，用于存放（程序）。

「分析」：CPU 内用于存放实现指令系统全部指令的微程序的只读存储器称为控制存储器。

### 三、改错题：

（2000 年）单总线结构系统是指：各大功能部件之间用一根信号线连接。

「答案」：单总线结构系统是指各寄存器及 ALU 之间的数据通路只用一条总线构成。

（2002 年）CPU 只是计算机的控制器。

「分析」：计算机硬件系统是由运算器、控制器、存储器、输入设备和输出设备等五大部分组成，其中将运算器和控制器合在一起称为中央处理器，简称为 CPU。

「答案」：CPU 是由控制器和运算器组成的。

（2003 年）硬连线方式是用时序电路产生时间控制信号，用存储逻辑电路实现各种控制功能。

「分析」：在采用组合逻辑和时钟信号相结合的硬连线控制器中，时间控制信号是由时序电路产生，而各种控制功能则是由组合逻辑电路实现。

「答案」：硬连线方式是用时序电路产生时间控制信号，用组合逻辑电路实现各种控制功能。

（2004 年）在一条微指令中，顺序控制部分的作用是发出指挥全机工作的控制信号。

「分析」：在一条微指令中，控制字部分的作用是发出指挥全机工作的控制信号；顺序控制部分的作用是产生后继微指令的地址。

「答案」：在一条微指令中，顺序控制部分的作用是产生后继微指令的地址。

### 四、简答题：

**在 CPU 中，哪些寄存器属于控制用的指令部件？它们各起什么作用？（5 分）**

「答案」：（1）程序计数器 PC，提供取指地址，从而控制程序执行顺序。

（2）指令寄存器 IR，存放现行指令，作为产生各种微操作命令的基本逻辑依据。

（3）程序状态寄存器 PS，记录程序运行结果的某些特征标志，或用来设置程序运行方式与优先级，参与形成某些微操作命令。

**（2001 年）1.硬连线控制器如何产生微命令？产生微命令的主要条件有哪些？**

「答案」：硬连线控制器依靠组合逻辑电路产生命令；（1 分）

组合逻辑电路的输入是产生微命令的条件，

主要有：① 指令代码；② 时序信号；③ 程序状态信息与标志位；④ 外部请求信号。

**3.微程序控制器怎么产生操作控制信号，这种控制器有何优缺点？**

「答案」：操作控制信号的产生：事先把操作控制信号以代码形式构成微指令，然后存放到控制存储器中，取出微指令时，其代码直接或译码产生操作控制信号。

优点：规整、易于修改和扩展。 缺点：速度较慢。

**.当读取并执行一条指令时，控制器的主要功能是什么？**

「答案」：① 从主存取指令，并计算下一条指令在主存中的地址；② 对指令进行译码，产生相应的操作控制信号；③ 控制指令执行的步骤和数据流动的方向。

与硬连线控制器相比，微程序控制器有哪些优缺点？

「答案」：与硬连线控制器相比，微程序控制器的优点是设计规整、易于修改和扩展。缺点是比硬连线控制器速度慢。

**28.硬连线控制器主要由哪几部分构成？它是如何产生控制信号的？**

「答案」：硬连线控制器主要由时钟源、环形脉冲发生器、控制信号编码器电路和指令译码器电路构成。硬连线控制器采用组合逻辑与时钟信号结合的方式产生控制信号。

**微程序控制器的构成**：控制存储器、微指令寄存器、微地址寄存器、地址转移逻辑等。

微指令控制字编码的方式：直接表示法、编码表示法、混合表示法。

直接表示法是将每个控制信号都作为微指令中的一个位。这种方法的特点是简单直观，其输出直接用于控制，但编码效率低。

编码表示法是将微指令进行分组编码，将不同时出现的相斥信号分在一个组中，然后将其编码成较短的代码。这种方法减少了控制存储器所需要的存储器的代码的数量，但是编码的指令代码需要译码器译码，增加了控制信号的延迟，影响 CPU 的工作频率。

混合表示法是把直接表示法与编码方法相结合使用，即采用部分直接表示部分编码的方法，将一些速度要求(湖北自考网)较高，或与其他控制信号都相容的控制信号以直接方式表示，而将剩余信号以编码方式。混合表示法便于综合考虑指令字长、灵活性和执行速度方面的要素。

微地址的形成方法：计数器方式、断定方式和结合方式。

计数器方式，又称增量方式。用微程序计数器  $\mu PC$  来产生指令的微地址，将微程序中的各条微指令按顺序安排在控制存储器中，后继地址由现行微地址加上一个增量形成。

断定方式，根据机器状态决定下一条微指令的地址，下一条微指令的地址包含在当前微指令的代码中。

结合方式，是将计数器方式和断定方式相结合。

中央处理器的基本功能：计算机的中央处理器（CPU）具有以下 4 个方面的基本功能：

- （1）指令控制，即对程序运行的控制；
- （2）操作控制，即对指令内操作步骤的控制；
- （3）数据运算，即对数据进行算术运算和逻辑运算，这是 CPU 的最基本功能；
- （4）异常处理和中断处理，如处理运算中的溢出等错误情况以及处理外部设备的服务请求等 此外，CPU 还具有存储管理、总线管理、电源管理等扩展功能

## 第 6 章 总线系统

一、名词解释：

**总线**：计算机中连接功能单元的公共线路，是一束信号线的集合，包括数据总线、地址总线和控制总线。

**同步通信方式**：采用这种方式的总线传输中，所有的设备都从一个公共的时钟信号中获得定时信息。

**主设备**：获得总线控制权的设备。

**猝发数据传输方式**：在一个总线周期内传输存储地址连续的多个数据字的总线传输方式。

**总线的同步通信方式**：采用这种方式的总线传输中，所有的设备都从一个公共的时钟信号中获得定时信息。

**总线从设备**：被主设备访问的设备。

- 1、猝发传输方式：在一个总线周期内传输存储地址连续的多个数据字的总线传输方式。
- 2、**四边沿协议**（全互锁）：全互锁的总线通信异步方式，就绪信号和应答信号的上升边沿和下降边沿都是触发边沿。
- 3、码元：信息传输通道中，携带数据信息的信号单元。
- 4、波特率：码元传输速率，每秒通过信道传输的码元数。（传的是信号）
- 5、比特率：信息位传输速率，每秒钟通过信道传输的有效信息量。（传的是信息）
- 6、**UART**：通用异步接收器/发送器，一种典型的集成电路异步串行接口电路。
- 7、主设备：获得总线控制权的设备。
- 8、从设备：被主设备访问的设备。
- 9、总线事务：从总线的请求到完成总线的使用的操作序列。
- 10、总线协议：总线通信同步方式规则，规定实现总线数据传输的定时规则。
- 11、总线访问延迟：是主设备为获得总线控制权而等待的时间。
- 12、**总线周期**：是主设备占用总线的时间。
- 13、总线裁决方式：决定总线由哪个设备进行控制的方式。
- 14、**系统总线**：是用来连接系统内各大功能模块或设备，实现系统种各电路板的连接。
- 15、**数据帧**：串行数据传输的位格式，包括起始位，数据位，校验位，结束位和空闲位。
- 16、**同步通信**：所有的设备都从一个公共的时钟信号中获得定时信息。
- 17、**异步通信**：使用一个在 CPU 和设备之间的“握手”信号，去除了公共的时钟信号，从而使得操作变成异步的。非互锁、半互锁、全互锁。
- 18、链式查询方式（菊花链方式）：各申请总线的设备合用一条总线作为请求信号线，而总线控制设备的响应信号线则串接在各设备间。
- 19、计数器定时查询方式：集中式总线裁决方式之一，设备要求使用总线时通过一条公用请求线发出，总线控制器按计数的值对各设备进行查询。
- 20、独立请求方式：集中式总线裁决方式之一，每一个设备都有一个独立的总线请求信号线送到总线控制器，控制器也给各设备分别发送一个总线响应信号。
- 21、串行传输：是指数据的传输在一条线路上按位进行。（只需一条数据传输线，线路的成本低，适合于长距离的数据传输）
- 22、并行传输：每个数据位都需要单独一条传输线，所有的数据位同时进行传输。（在采用并行传输方式的总线中，除了有传输数据的线路外，还可以具有传输地址和控制信号的线路，地址线用于选择存储单元和设备，控制线用于传递操作信号）
- 23、复合传输：又称总线复用的传输方式，它使不同的信号在同一条信号线上传输，不同的信号在不同的时间片中轮流地身总线的同一条信号线上发出。（它与并串传输的区别在于分时地传输同一数据源的不同信息。）
- 24、消息传输方式：总线的信息传输方式之一，将总线需要传送的数据信息、地址信息、和控制信息等组合成一个固定的数据结构以猝发方式进行传输。
- 25、总线：一组可由多个部件分时共享的信息传输线。

## 二、选择填空题：

8. “总线忙”信号由（ A ）建立。

A.获得总线控制权的设备

B.发出“总线请求”的设备

C.总线控制器

D.CPU

「分析」：在总线控制机制中，准备使用总线的设备向总线控制器发出“总线请求”由总线控制器进行裁决。如果经裁决允许该设备使用总线，就由总线控制器向该设备发出一个

“总线允许”信号。该设备接收到此信号后，发出一个“总线忙”信号用来通知其他设备总线已被占用。当该设备使用完总线时，将“总线忙”信号撤销，释放总线。 「答案」：A

12.系统总线是用来连接（ 系统内各大功能模块或设备 ）的总线。

「分析」：按总线的连线类型不同，总线可分为：① 芯片级总线（CPU 内部总线）：连接 CPU 内部运算器、控制器、寄存器等的通路。② 板级总线：连接主板中的 CPU 和主存等部件，也称局部总线。③ 系统总线是用来连接系统内各大功能模块或设备。

「答案」：系统内各大功能模块或设备

14.并行接口与 I/O 设备之间同时传送的位数，大多是 （ 8 ） 位。

「分析」：并行接口与 I/O 设备之间同时传送的 8 位数（1 个字节）

14.在不同速度的设备之间传送数据，（C ）。

- A.必须采用同步控制方式
- B.必须采用异步控制方式
- C.可以选用同步方式，也可选用异步方式
- D.必须采用应答方式

「分析」：在不同速度的设备之间进行数据传送，既可以使用同步方式，也可以使用异步方式。异步方式主要是用于在不同的设备之间进行通信，而如果两种速度的设备使用同一个时钟信号进行控制，采用同步的数据传送方式，同样可以进行数据的传送，只是快速设备的速度性能发挥不出来。「答案」：C

15.挂接在总线上的多个部件（B ）。

- A.只能分时而向总线发送数据，并只能分时而从总线接收数据
- B.只能分时而向总线发送数据，但可同时从总线接收数据
- C.可同时向总线发送数据，并同时从总线接收数据
- D.可同时向总线发送数据，但只能分时而从总线接收数据

「分析」：为了使总线上的数据不发生“碰撞”，挂接在总线上的多个设备只能分时而向总线发送数据，即每一个时刻只能有一个设备可以向总线传送数据，而从总线上接收数据的设备可有多，因为接收数据的设备不会对总线产生“干扰”。「答案」：B

12.异步传送方式常用于（ D ）中，作为主要控制方式。

- A.微型机的 CPU 内部控制
- B.硬连线控制器
- C.微程序控制器
- D.串行 I/O 总线

「分析」：异步传输方式主要用于控制两种速度有一定差别的设备的信息传送，一般用在快速 CPU 与慢速的外设之间进行串行通信的场合。「答案」：D

13.串行总线主要用于（ A ）。

- A.连接主机与外围设备
- B.连接主存与 CPU
- C.连接运算器与控制器
- D.连接 CPU 内部各部件



「分析」：串行通信方式由于其信息传送速度慢、信息传送的距离较长、所使用的信号线数量较少等特点，主要用于连接主机和慢速的外围设备，例如主机与串行鼠标之间的信息传送。「答案」：A

4.下列说法中正确的是（ C ）。

- 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

「分析」：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。「答案」：C

9.在总线上，同一时刻（A）。

- A.只能有一个主设备控制总线传输操作
- B.只能有一个从设备控制总线传输操作
- C.只能有一个主设备和一个从设备控制总线传输操作
- D.可以有多个主设备控制总线传输操作

分析：总线上的设备要控制总线必须先获得总线的控制权，获得总线控制权的设备称为主设备，被主设备访问的设备称为从设备。在总线上信息的传输由主设备启动，一条总线上可以有多个设备能成为主设备，但在同一时刻只能有一个主设备控制总线的传输操作。答案：A

4.系统级的总线是用来连接（D）。

- A.CPU 内部的运算器和寄存器
- B.主机系统板上的所有部件
- C.主机系统板上的各个芯片
- D.系统中的各个功能模块或设备

分析」：按总线的连线类型不同，总线可分为：① 芯片级总线（CPU 内部总线）：连接 CPU 内部运算器、控制器、寄存器等的数据通路。② 板级总线：连接主板中的 CPU 和主存等部件，也称局部总线。③ 系统总线是用来连接系统内各大功能模块或设备。「答案」：D

15.总线从设备是（ C ）。

- A. 掌握总线控制权的设备
- B.申请作为从设备的设备
- C. 被主设备访问的设备
- D.总线裁决部件

「分析」：主设备：获得总线控制权的设备。从设备：被主设备访问的设备。「答案」：C

12.波特率表示传输线路上（ A ）。

- A.信号的传输速率
- B.有效数据的传输速率

C.校验信号的传输速率

D.干扰信号的传输速率

「分析」：波特率是码元传输速率，每秒通过信道传输的码元数。（传的是信号）。比特率是信息位传输速率，每秒钟通过信道传输的有效信息量。（传的是信息）「答案」：A

13.不同信号在同一条信号线上分时传输的方式称为（A）。

A.总线复用方式

B.并串行传输方式

C.并行传输方式

D.串行传输方式

「分析」：串行传输是指数据的传输在一条线路上按位进行。并行传输是每个数据位都需要单独一条传输线，所有的数据位同时进行传输。不同信号在同一条信号线上分时传输的方式称为总线复用方式。「答案」：A

17.按照传输定时的方法划分，总线数据通信方式可分为（同步通信）和（异步通信）两类。

「分析」：按照传输定时的方法划分，总线数据通信方式可分为：① 同步通信：所有的设备都从一个公共的时钟信号中获得定时信息。② 异步通信：使用一个在 CPU 和设备之间的“握手”信号，去除了公共的时钟信号，从而使得操作变成异步的，有非互锁、半互锁、全互锁三种方式。

三、改错题：

（2002 年）.按时序控制方式分，总线可分为串行总线和并行总线。

「分析」：对总线的分类有不同的分类标准：按传送格式分为：串行总线、并行总线；按时序控制方式分为：同步总线（含同步扩展总线）、异步总线；按功能分为：系统总线、CPU 内部总线、各种局部总线。按时序控制方式分，总线可分成同步总线和异步总线。

（2003 年）23.串行通信只能采用异步方式。

「分析」：串行通信是指数据的传输是在一条传输线路上按位进行，它可以采用异步方式，也可以采用同步（湖北自考网）方式。采用异步方式时，发送与接收设备之间采用“握手”信号来进行同步，而采用同步方式时，发送与接收设备都从同一个公共的时钟信号中获得定时信息。「答案」：串行通信可以采用异步方式，也可以采用同步方式。

（2004 年总线周期是指：任意总线设备为获取总线控制权而等待的时间与占用总线的时间之和。

「分析」：总线访问延迟：是主设备为获得总线控制权而等待的时间。总线周期：是主设备占用总线的时间。「答案」：总线周期是指主设备占用总线的时间。

四、简答题：

（2000 年）1.何谓存储总线？何谓 I/O 总线？各有何特点？（4 分）

「答案」 存储总线是连接 CPU 和主存储器之间的专用总线，速度高。

I/O 总线是连接主机（CPU）与 I/O 设备之间的总线，可扩展性好。

（2001 年）4.总线的分类方法主要有哪几种？请分别按这几种法说明总线的分类。

答案：① 按传送格式分为：串行总线、并行总线；② 按时序控制方式分为：同步总线（含同步扩展总线）、异步总线；③ 按功能分为：系统总线、CPU 内部总线、各种局部总线。

（2002 年）4.何谓串行传输，有何优缺点？适用什么场合？

「答案」： 串行传输是指数据的传输在一条线路上按位进行。

优点：线路成本低。缺点：传送速度慢。

适用场合：主机与低速外设间的传送、远距离通信总线的数据传送、系统之间的数据传送。

（2003 年）28.总线的同步通信方式与异步通信方式有什么区别？各适用于哪些场合？

答案」：同步通信方式中：数据传送操作由统一的时序信号同步定时控制，有严格的时钟周期划分，总线操作有固定的时序，设备之间没有应答信号。适合各设备速度固定且一致（或差异不大）的场合。

异步通信方式中：数据传送操作所需时间视需要而定，总线操作周期时间不固定，没有时钟周期划分，设备之间采用握手信号的应答方式。适合：各设备速度差异较大的场合。

（2004 年）29.串行总线和并行总线有何区别？ 各适用于什么场合？

「答案」：串行总线的数据传输是在一条线路上按位进行。线路成本低，传送速度慢。适用场合：主机与低速外设间的传送、远距离通信总线的数据传送、系统之间的数据传送。

并行总线的每个数据位都需要单独一条传输线，所有的数据位同时进行传输。线路成本高，传送速度快。适用场合：短距离的高速数据传输。

（2005 年）29.系统总线接口有哪几项基本功能？

「答案」：① 控制：传递总线上的控制信息，主设备会通过总线接口向从设备发出控制信息。② 数据缓存：在总线传递信息时，在总线接口中临时存放数据。③ 状态设置通过总线和转换从设备的工作信息，便于主设备了解从设备的信息。④ 数据转换：某些总线接口需要对传递的数据进行转换。⑤ 整理：对接口本身进行调整。⑥ 程序中断。

1.什么是总线裁决？总线裁决有哪几种方式？

「答案」：总线裁决就是决定总线由哪个设备进行控制。

总线裁决方式可分为集中式裁决和分布式裁决两种。

集中式裁决将总线的控制功能用一个专门的部件实现，这个部件可以位于连接在总线的某个设备上。当一个设备需要向共享总线传输数据时，它必须先发出请求，在得到许可时才能发出数据。裁决部件接收来自各个设备的总线使用请求信号，向其中某一个设备发出总线许可信号。

分布式裁决将控制功能分布在连接在总线上的各设备中，一般是固定优先级的。每个设备分配一个优先号，发出总线请求的设备将自己的优先号送往请求线上，与其他设备的请求信号构成一个合成信号，并将这个合成裁决信号读入以判断是否有优先级更高的设备申请总线。这样可使得优先级最高的设备获得总线使用权。

2.集中式裁决有哪几种方式：

「答案」：（1）链式查询方式（菊花链方式）：各申请总线的设备合用一条总线作为请求信号线，而总线控制设备的响应信号线则串接在各设备间。

（2）计数器定时查询方式：集中式总线裁决方式之一，设备要求使用总线时通过一条公用请求线发出，总线控制器按计数的值对各设备进行查询。



(3) 独立请求方式：集中式总线裁决方式之一，每一个设备都有一个独立的总线请求信号线送到总线控制器，控制器也给各设备分别发送一个总线响应信号。

独立请求方式可以和链式查询方式结合，构成分组链式查询方式。

### 3.提高总线速度的措施。

「答案」：从物理层次：1.增加总线宽度；2.增加传输的数据长度；3.缩短总线长度；4.降低信号电平；5.采用差分信号；6.采用多条总线。从逻辑层次：1.简化总线传输协议；2.采用总线复用技术；3.采用消息传输协议。

4.什么是串行接口？什么是并行接口？他们与系统总线及 I/O 设备之间的传递格式分别是什么？

「答案」：串行接口和并行接口都是总线与设备之间的接口部件，但与设备间的数据格式不同。串行接口与外设之间串行，与系统总线之间并行。并行接口与外设之间并行，与系统总线之间并行。

## 第 7 章 输入输出系统

### 一、名词解释：

9.DMA 方式：直接存储器访问，直接依靠硬件实现主存与外设之间的数据直接传输，传输过程本身不需 CPU 程序干预。

5.I/O 接口：是指连接主机和外围设备的逻辑部件。

20.中断屏蔽：CPU 处理一个中断的过程中，对其他一些外部设备的中断进行阻止。

17.统一编址：将输入输出设备中控制寄存器、数据寄存器、状态寄存器等与内存单元一样看待，将它们和内存单元联合在一起编排地址，用访问内存的指令来访问输入输出设备接口的某个寄存器，从而实现数据的输入输出。

25.通道程序：通道命令构成通道程序。在通道程序的控制下，通道对外围设备进行数据传输控制。

1.统一编址：将输入输出设备中控制寄存器、数据寄存器、状态寄存器等与内存单元一样看待，将它们和内存单元联合在一起编排地址，用访问内存的指令来访问输入输出设备接口的某个寄存器，从而实现数据的输入输出。

2.单独编址：将输入输出设备中控制寄存器、数据寄存器、状态寄存器单独编排地址，用专门的控制信号进行输入输出操作。

3.单级中断：CPU 在执行中断服务程序的过程中禁止所有其他外部中断。

4.多级中断：CPU 在执行中断服务程序的过程中可以响应级别更高的中断请求。

5.中断屏蔽：CPU 处理一个中断的过程中，对其他一些外部设备的中断进行阻止。

6.DMA：直接存储器访问，直接依靠硬件实现主存与外设之间的数据直接传输，传输过程本身不需 CPU 程序干预。

7.现场保护：CPU 在响应中断请求时，将程序计数器和有关寄存器内容等系统的状态信息存储起来，以使中断处理结束后能恢复原来的状态继续执行程序，称为现场保护。

8.中断向量：外设在向 CPU 发出中断请求时，由该设备通过输入输出总线主动向 CPU 发出一个识别代码，这个识别代码通常称为中断向量。

9.自陷：当 CPU 出现有算术操作异常、非法指令、越权操作和访存中的异常等某种内部情况时自己引起的中断称为自陷。

10.软件中断：由自陷指令引起的中断称为软件中断，又称为系统调用。

11.通道命令：通道用于执行输入输出操作的指令，也叫通道控制字（CCW）。

## 二、选择填空题：

7.设置中断排队 判优逻辑 的目的是（B ）。

A.产生中断源编码

B.使同时提出的请求中的优先级别最高者，得到及时响应

C.使 CPU 能方便地转入中断服务子程序

D.提高中断响应速度

分析：当有多个中断请求同时出现，中断服务系统必须能从中选出当前最需要给予响应的最重要的中断请求，这就需要预先对所有的中断进行优先级排队，这个工作可由中断优先级判断逻辑来完成，排队的规则可由软件通过对中断屏蔽寄存器进行设置来确定。答案：B

10.通道程序在内存中的首地址由（通道地址字）给出。

「分析」：CPU 使用通道进行一个输入输出操作时，先发出一个通道启动信号。通道收到启动信号后，到指定的内存单元中取通道地址字，并将其放入通道地址寄存器中。此通道地址字为通道程序在内存中的首地址。「答案」：通道地址字

11.在不改变中断响应次序的条件下，通过（改写中断屏蔽字）可以改变中断处理次序。

【分析】：在多重中断系统中，可以通过设置中断优先级来决定各个中断的级别。在实际的计算机系统中是通过 CPU 内部的一个中断屏蔽字寄存器来实现对不同中断的分别禁止的，这个寄存器可在中断处理程序中重新设置，这样就可以改变原有的中断优先级别。

16.采用 DMA 方式传送数据是由 DMA 接口来控制数据在外设 和 主存储器之间传输。

[分析]：DMA 是指直接存储器访问，是利用一个专门的接口电路将计算机的主存储器与高速的外设相连接，当计算机要与外设进行数据传送时，由 CPU 发出一个控制信号启动 DMA 之后由 DMA 来控制完成外设与主存储器之间的数据传送，其传送方式为数据块（数据成组）传送，传送过程为连续的，中间没有停止等待的时间，所以数据的传送速度较高。

## 三、改错题：

（2000 年）7.对 I/O 数据传送的控制方式，可分为程序中断控制方式和独立编址传送控制方式两种。

「分析」：对 I/O 数据传送的控制方式，可分为程序直接控制方式、程序中断控制方式、DMA 控制方式、通道控制方式等。程序中断控制方式只是其中的一种方法，独立编址是指对 I/O 设备的控制寄存器、数据寄存器、状态寄存器等单独进行地址编排，使用专门的指令对其进行操作，可用在各种数据传送的控制方式中。

「答案」：对 I/O 数据传送的控制方式，可分为：程序直接控制方式、程序中断方式、DMA 方式、通道控制方式等。

（2002）5.对外设统一编址是指给每个外设设置一个地址码。

「分析」：CPU 与外设之间的信息传送是通过硬件接口来实现的，各种外设的硬件接口上又都包含有多个寄存器，如控制寄存器、数据寄存器、状态寄存器等。统一编址是将外设

接口上的各种寄存器等同于内存储器的存储单元,通过使用访问内存单元的指令来访问外设接口上的各个寄存器,这样就可以使用访存指令来访问外设,输入输出操作简单,程序设计比较简便。由于外设接口上的寄存器种类和数量通常不止一个,所以一个外设至少对应一个以上的内存地址。

「答案」: 对外设统一编址是将外设接口上的寄存器等内存单元,给每个外设设置至少一个地址码。

(2003 年) 25.在常见的微机系统中,磁盘常采用通道方式与主存交换信息。

「分析」: 通道传输方式是采用通道处理器将多个输入输出设备与 CPU 和主存储器相连接,并控制其信息的传输,主要用于大型计算机以及网络服务器等含有许多输入输出设备并对输入输出有较高要求的场合;而 DMA 方式是采用 DMA 控制器将外围设备与主存储器相连接,并控制其信息的传输,主要用于微型计算机中外设与主存之间需要成批传输数据的场合,如微机系统中磁盘与主存之间的数据传输。

「答案」: 在常见的微机系统中, **磁盘常采用 DMA 方式与主存交换数据。**

(2004 年) 25.通道就是一组输入输出传送线。

「分析」: 通道是一种比 DMA 更高级的 I/O 控制部件,具有更强的独立处理数据的输入 / 输出功能,能同时控制多台同类型或不同类型的设备。它在一定的硬件基础上,利用通道程序实现对 I/O 的控制,更多地免去了 CPU 的介入,使系统的并行性能更高。

「答案」: **通道是具有更强的独立处理数据的输入 / 输出功能,能同时控制多台同类型或不同类型的设备。**

#### 四、简答题:

**7.以 DMA 方式实现传送,大致可分为哪几个阶段? (3 分)**

「答案」:

- ① DMA 传送前的预置阶段 (DMA 初始化);
- ② 数据传送阶段 (DMA 传送);
- ③ 传送后的结束处理。

**2.何谓中断方式? 它主要应用在什么场合? 请举二例。(解释的不好)**

「答案」:

- ① 中断方式指: CPU 在接到随机产生的中断请求信号后,暂停原程序,转去执行相应的中断处理程序,以处理该随机事件,处理完毕后返回并继续执行原程序;
- ② 主要应用于处理复杂随机事件、控制中低速 I/O;
- ③ 例: 打印机控制,故障处理。

**3.在 DMA 方式预处理(初始化)阶段, CPU 通过程序送出哪些信息?**

「答案」: ① 测试设备状态,预置 DMA 控制器工作方式;

② 主存缓冲区首址,交换量,传送方向;

③ 设备寻址信息,启动读 / 写。

**6.中断接口一般包含哪些基本组成? 简要说明它们的作用。**

答案: ① 地址译码。选取接口中有关寄存器,也就是选择了 I/O 设备。

② 命令字 / 状态字寄存器。供 CPU 输出控制命令,调回接口与设备的状态信

息。

③ 数据缓存。提供数据缓冲，实现速度匹配。

④ 控制逻辑。如中断控制逻辑、与设备特性相关的控制逻辑等。

### 5.何谓 DAM 方式？说明它的适用场合。

「答案」：定义：由 DMA 控制器控制系统总线，直接依靠硬件实现主存与 I/O 设备之间的数据直传，传送期间不需要 CPU 程序干预。适用场合：高速、批量数据的简单传送。

### 6.何谓多重中断？如何保证它的实现？

「答案」：多重中断：CPU 在响应处理中断过程中，允许响应处理更高级别的中断请求，这种方式称为多重中断。

实现方法：在中断服务程序的起始部分用一段程序来保存现场、送新屏蔽字以屏蔽同级别和低级别的中断请求、然后开中断，这样 CPU 就可响应更高级别的中断请求，实现多重中断。

### 31.试对程序中断方式和 DMA 方式各分别举出二种应用例子。

「答案」：中断方式常用于打印机输出、键盘输入等；

DMA 方式常用于读 / 写磁盘、读 / 写磁带等。

### 30.主机与外围设备之间信息传送的控制方式有哪几种？采用哪种方式 CPU 效率最低？

「答案」：主机与外围设备之间信息传送的控制方式有四种：程序查询方式、中断方式、DMA 方式和通道方式。程序查询方式 CPU 效率最低。

### 31.试比较中断方式与 DMA 方式的主要异同，并指出它们各自应用在什么性质的场合。

相同点：这两种方式下，主机和 I/O 设备都是并行工作。

不同点：中断方式在 CPU 响应了 I/O 设备的中断请求后，要暂停现行程序的执行，转为 I/O 设备服务。DMA 方式(湖北自考网)直接依靠硬件实现主存与 I/O 设备之间的数据直传，传送期间不需要 CPU 程序干预，CPU 可继续执行原来的程序，CPU 效率比中断方式。

DMA 方式适用场合：高速、批量数据的简单传送。

中断方式适用场合：处理复杂随机事件、控制中低速 I/O 设备。

### 30.基本的 DMA 控制器的主要部件有哪些？

「答案」：基本的 DMA 控制器的主要部件有：地址寄存器、数据寄存器、命令寄存器、长度计数器、标志寄存器、控制逻辑等。

### 31.简述多重中断系统中 CPU 响应处理一次中断的步骤。

「答案」：① 关中断；② 保存现场信息；③ 判别中断条件；④ 开中断；⑤ 执行中断服务程序；⑥ 关中断；⑦ 恢复现场信息；⑧ 开中断。

1.中断方式的接口控制器功能：能向 CPU 发出中断请求信号；能发出识别代码提供提供引导 CPU 在响应中断请求后转入相应服务程序的地址；CPU 要能够对中断请求进行允许或禁止的控制；能使中断请求参加优先级排队。

### 2.CPU 与外围设备进行通信有三种类型：① CPU 向外围设备发出操作控制命令；② 外

围设备向 CPU 提供状态信息；③ 数据在 CPU 和外围设备之间传递。

3.中断裁决机制：轮询、菊花链、独立请求。

4.CPU 与 DMA 访问内存冲突的裁决的三种方法：① CPU 等待 DMA 的操作；② DMA 乘存储器空闲时访问存储器；③ CPU 与 DMA 交替访问存储器。

5.CPU 启动 DMA 的步骤：① 测试设备状态；② 写存储器地址寄存器；③ 写长度计数器；④ 启动 DMA 控制逻辑。

6.通道的三种类型：

(1)选择通道：它与设备之间的传输一直维持到设备请求的传输完成为止，然后为其它外围设备传输数据。数据宽度是可变的，通道中包含一个保存 IO 数据传输所需的参数寄存器。

(2)数组多路通道：以数组为单元在若干高速传输操作之间进行交叉复用。

(3)字节多路通道：用于连接多个慢速的和中速的设备，这些设备的数据传送以字节为单位，字节交叉模式、猝发模式。

7.字节多路通道与数组多路通道的区别：

(1) 数组多路通道允许多个设备同时工作，但只允许一个设备进行传输型操作，而其它设备进行控制型操作；字节多路通道不仅允许多个设备操作，而且允许它们同时进行传输型操作。

(2)数组多路通道与设备之间的数据传送的基本单位是数据块，而字节多路通道与设备之间的数据传送基本单位是字节。

8.通道的功能：① 接受 CPU 的 I/O 操作指令 ② 从内存中读取通道程序，并执行。③内存与外设之间的传送操作；④ 读取外设的状态信息，形成整个通道的状态信息，⑤ 向 CPU 发出 IO 操作中断请求

## 第 8 章 外围设备

9.在大多数磁盘中（ C ）。

A.各磁道的位密度相同

B.最外圈磁道的位密度最大

C.最内圈磁道的位密度最大

D.写入时选择较高的位密度，以增加记录信息；读出时选择低的位密度，以提高可靠性。

「分析」：位密度是指磁道中单位长度所存储的信息量。在磁盘存储器中，每个磁道所存储的信息是相同的，而最内圈磁道的长度最小，所以该磁道的位密度最大。答案」：C

10.在调频制记录方式中，是利用（D ）来写 0 或 1。

A.电平高低的变化

B.电流幅值的变化

C.电流相位的变化

D.电流频率的变化



「分析」：在调频制记录方式中，信息的写入是依靠写入电流频率的变化来实现的，写 1 时的电流变化频率是写 0 时电流变化频率的 2 倍。「答案」：D

- 14.在常用磁盘中，（ C ）。
- A.外圈磁道容量大于内圈磁道容量
  - B.各道容量不等
  - C.各磁道容量相同
  - D.内圈磁道容量大于外圈磁道容量

「分析」：位密度是指磁道中单位长度所存储的信息量。在磁盘存储器中，每个磁道所存储的信息是相同的。「答案」：C

- 15.在下列存储器中，（ A ）可以作为主存储器。
- A.半导体存储器
  - B.硬盘
  - C.光盘
  - D.磁带
- 「答案」：A

- 1.CRT 图形显示器的分辨率表示（ B ）。
- A.一个图像点（像素）的物理尺寸
  - B.显示器一行能显示的最大图像点数与一列能显示的最大图像点数
  - C.显示器屏幕可视区域的大小
  - D.显示器能显示的字符个数
- 「答案」：B

「分析」：CRT 图形显示器的分辨率是衡量显示器显示清晰度的指标，是以图像点（像素）的个数为标志，即显示器一行能显示的最大图像点数与一列能显示的最大图像点数的乘积。

- 6.下列各种记录方式中，不具自同步能力的是（ A ）。
- A.不归零制
  - B.改进型调频制 MFM
  - C.调相制 PM
  - D.调频制

「分析」：自同步能力是指从读出的数据中自动提取同步信号的能力，其能力大小可用最小磁化翻转间隔与最大磁化翻转间隔的比值来表示，比值越大，则自同步能力越强。在各种记录方式中，NRZ，NRZ 记录方式没有自同步能力，PM，FM，MFM 记录方式具有自同步能力。「答案」：A

- 11.在下列存储器中，哪种速度最快（ C ）。
- A.磁盘
  - B.磁带
  - C.主存
  - D.光盘

「分析」：各种存储器由于存储介质和内部结构的不同，其读写速度也不同。读写速度由快到慢的次序为：高速缓冲存储器、主存储器、辅助存储器。各种辅助存储器的读写速度由快到慢次序为：硬盘存储器、光盘存储器、磁带存储器。「答案」：C

- 2.在下列设备中，属于图形输入设备的是（ C ）。「答案」：C
- A.键盘
  - B.条形码阅读机
  - C.数字化仪
  - D.显示器

「分析」：图形输入设备有鼠标、数字画仪和触摸屏；文字输入设备有键盘、磁卡阅读机、条形码阅读机、纸带阅读机、卡片阅读机。图像输入设备有扫描仪、数码相机和摄像头等。

3.磁表面存储器记录信息是利用磁性材料的（A）。

- A.磁滞回归线特性
- B.磁场渗透特性
- C.磁场分布特性
- D.磁场吸引力特性

「分析」：磁表面存储器记录信息是利用磁性材料的磁滞回归线特性。「答案」：A

24 针打印机的打印头的针排列是（ B）。 答案」：B

- A.24 根针排成一行
- B.24 根针排成 2 列
- C.24 根针排成 3 列
- D.24 根针排成 4 列

「分析」：9 针打印机的打印头的针排列是单列，24 针打印机的打印头的针排列是双列。

15.在常用磁盘的各磁道中（B）。

- A.最外圈磁道的位密度最大
- B.最内圈磁道的位密度最大
- C.中间磁道的位密度最大
- D.所有磁道的位密度一样大

「分析」：（和 2000 年题目一样）位密度是指磁道中单位长度所存储的信息量。在磁盘存储器中，每个磁道所存储的信息是相同的，而最内圈磁道的长度最小，所以该磁道的位密度最大。「答案」：B

二、填空题：

20. 磁盘存储设备主要由磁记录介质、（ 磁盘控制器 ）和（ 盘驱动器 ）三个部分组成。

三、改错题：

（2000 年）8.在计算机系统中，除 CPU 外的其它部件和设备都称为外围设备。

「分析」：计算机硬件系统是由运算器、控制器、存储器、输入设备和输出设备 5 个部分组成的，将运算器和控制器合在一起称为 CPU；存储器又分为内部存储器和外部存储器两类，内部存储器与 CPU 一起组成计算机的主机，外存储器、输入设备、输出设备等合称为外围设备。

「答案」：在计算机系统中，除 CPU 和主存之外的其他部件和设备，常被称为外围设备。

（2000 年）10.写入硬盘时，若一个文件的长度超出一个磁道的容量，则继续写入同面的相邻磁道中。

「分析」：硬盘存储器一般由多个盘片组成，每个盘片的两个面都用来记录信息（最外层的两个盘片有时只用一个面），每个盘面中相同磁道号的各个磁道构成一个柱面。硬盘中信息的存储是按柱面顺序，即只有在一个柱面都被记录上信息后才进入下一个柱面。在柱面内信息的存储是按磁道顺序，即只有在一个磁道都被记录上信息后才进入下一个相邻面的磁道中。

「答案」：写入硬盘时，若一个文件的长度超过一个磁道的容量，则继续写入同一柱面的相邻面的磁道中。

（2002 年）4.显示适配器中的显示缓冲存储器用于存放显示器将要向 CPU 输入的信息。

「分析」：显示器是计算机硬件系统中的输出设备，由于 CPU 的处理速度较快，而显示器显示输出的速度较慢，为了将两者间的速度差别协调起来，在显示适配器中配置一定容量

的显示缓冲存储器，CPU 将要显示输出的信息先送入此缓冲存储器，再由显示适配器控制显示输出，而 CPU 则可以转去完成其他的任务。由此可见，显示适配器中的显示缓冲存储器用于存放 CPU 向显示器输出的信息。

「答案」：显示适配器中的显示缓冲存储器用于存放 CPU 向显示器输出的信息。

(2003 年) 24. 磁盘是存储器，不是外围设备。

「分析」：磁盘是计算机系统辅助存储器，当计算机从磁盘中读出数据信息时，磁盘属于输入设备，而当计算机向磁盘写入信息时，磁盘又属于输出设备。所以磁盘既是存储器，又属于外围设备。「答案」：磁盘既是存储器，又属于外围设备。

1. **归零制 (RZ)**：磁表面存储器记录信息时，不论某存储单元的代码是 0 或者 1，在记录下一个信息之前记录电流要恢复到零电流。在给磁头线圈送入的脉冲电流中，正脉冲表示 1，负脉冲表示 0。不具有自同步能力

2. **不归零制 (NRZ)**：磁表面存储器记录信息时，磁头线圈上始终有电流，不是正向电流就是反向电流，正向电流代表 1，反向电流代表 0。不具有自同步能力

3. **调相制 (PM)**：磁表面存储器记录信息时，在一个磁化元的中间位置，利用电流相位的变化进行写 1 或者写 0，所以(湖北自考网)通过磁头中的电流方向一定要改变一次。规定在记录数据 1 时，磁化翻转的方向是由负变正，记录数据 0 时磁化翻转的方向为由正变负。具有自同步能力

4. **调频制 (FM)**：磁表面存储器记录信息时，无论记录的代码是 1 还是 0，或者是连续的 1 或连续的 0 在相邻的两个存储元交界处电流要改变方向。在记录数据 1 时，还要在位与位之间再翻转一次，写 1 的电流频率是写 0 的 2 倍。具有自同步能力

5. **改进调频制 (MFM)**：只有连续记录两个或两个以上 0 时在位周期的起始位置处翻转一次，而不是在每个位周期的起始处都翻转。

6. **RLL 码**：在高密度磁盘中采用的信息编码技术，将原始数据序列变换成 0，1 游程长度受限制的代码，然后再用不归零制方式进行调制和写入。具有自同步能力

7. **磁盘访问时间**：指从发出读写命令，磁头从某一起始位置移动到新的记录位置，到结束从盘片读出或写入信息所花的时间。

**磁盘访问时间 = 寻道时间 + 旋转延迟 + 控制延迟 + 数据传输时间。**

8. **寻道时间**：是将磁头定位到所要求的磁道上所需的时间。

9. **旋转延迟**：是找道完成后到磁道上需要访问的信息到达磁头的时间。

10. **平均旋转延迟**：是磁盘旋转半周的时间，也称磁盘的寻址时间。

数据传输时间取决于读扇区数据时间和传输数据时间，等于两者的最大值。

**磁盘数据传输率 = 转速/秒 \* 每道容量**

11. **磁盘存储设备的主要技术指标**：存储密度、存储容量、寻址时间和数据传输等。

12. **光盘的结构包括**：光盘基片、存储介质和密封层。

13. **光盘存储设备有**：只读型 CD-ROM、WORM（写一次读多次）型和可擦写型三种。

## 第一章 计算机系统概论



1. 什么是计算机系统、计算机硬件和计算机软件？硬件和软件哪个更重要？

解：P3

计算机系统：由计算机硬件系统和软件系统组成的综合体。

计算机硬件：指计算机中的电子线路和物理装置。

计算机软件：计算机运行所需的程序及相关资料。

硬件和软件在计算机系统中相互依存，缺一不可，因此同样重要。

5. 冯·诺依曼计算机的特点是什么？

解：冯·诺依曼计算机的特点是：P8

计算机由运算器、控制器、存储器、输入设备、输出设备五大部件组成；

指令和数据以同等地位存放于存储器内，并可以按地址访问；

指令和数据均用二进制表示；

指令由操作码、地址码两大部分组成，操作码用来表示操作的性质，地址码用来表示操作数在存储器中的位置；

指令在存储器中顺序存放，通常自动顺序取出执行；

机器以运算器为中心（原始冯·诺依曼机）。

7. 解释下列概念：

主机、CPU、主存、存储单元、存储元件、存储基元、存储元、存储字、存储字长、存储容量、机器字长、指令字长。

解：P9-10

主机：是计算机硬件的主体部分，由 CPU 和主存储器 MM 合成为主机。

CPU：中央处理器，是计算机硬件的核心部件，由运算器和控制器组成；（早期的运算器和控制器不在同一芯片上，现在的 CPU 内除含有运算器和控制器外还集成了 CACHE）。

主存：计算机中存放正在运行的程序和数据存储器，为计算机的主要工作存储器，可随机存取；由存储体、各种逻辑部件及控制电路组成。

存储单元：可存放一个机器字并具有特定存储地址的存储单位。

存储元件：存储一位二进制信息的物理元件，是存储器中最小的存储单位，又叫存储基元或存储元，不能单独存取。

存储字：一个存储单元所存二进制代码的逻辑单位。

存储字长：一个存储单元所存二进制代码的位数。

存储容量：存储器中可存二进制代码的总量；（通常主、辅存容量分开描述）。

机器字长：指 CPU 一次能处理的二进制数据的位数，通常与 CPU 的寄存器位数有关。

指令字长：一条指令的二进制代码位数。

8. 解释下列英文缩写的中文含义：

CPU、PC、IR、CU、ALU、ACC、MQ、X、MAR、MDR、I/O、MIPS、CPI、FLOPS

解：全面的回答应分英文全称、中文名、功能三部分。

CPU：Central Processing Unit，中央处理机（器），是计算机硬件的核心部件，主要由运算器和控制器组成。

PC：Program Counter，程序计数器，其功能是存放当前欲执行指令的地址，并可自动计数形成下一条指令地址。

IR：Instruction Register，指令寄存器，其功能是存放当前正在执行的指令。

CU：Control Unit，控制单元（部件），为控制器的核心部件，其功能是产生微操作命令序

列。

ALU: Arithmetic Logic Unit, 算术逻辑运算单元, 为运算器的核心部件, 其功能是进行算术、逻辑运算。

ACC: Accumulator, 累加器, 是运算器中既能存放运算前的操作数, 又能存放运算结果的寄存器。

MQ: Multiplier-Quotient Register, 乘商寄存器, 乘法运算时存放乘数、除法时存放商的寄存器。

X: 此字母没有专指的缩写含义, 可以用作任一部件名, 在此表示操作数寄存器, 即运算器中工作寄存器之一, 用来存放操作数;

MAR: Memory Address Register, 存储器地址寄存器, 在主存中用来存放欲访问的存储单元的地址。

MDR: Memory Data Register, 存储器数据缓冲寄存器, 在主存中用来存放从某单元读出、或要写入某存储单元的数据。

I/O: Input/Output equipment, 输入/输出设备, 为输入设备和输出设备的总称, 用于计算机内部和外界信息的转换与传送。

MIPS: Million Instruction Per Second, 每秒执行百万条指令数, 为计算机运算速度指标的一种计量单位。

9. 画出主机框图, 分别以存数指令“STA M”和加法指令“ADD M”(M 均为主存地址)为例, 在图中按序标出完成该指令(包括取指令阶段)的信息流程(如→①)。假设主存容量为 256M\*32 位, 在指令字长、存储字长、机器字长相等的条件下, 指出图中各寄存器的位数。

解: 主机框图如 P13 图 1.11 所示。

(1) STA M 指令: PC→MAR, MAR→MM, MM→MDR, MDR→IR,

OP(IR)→CU, Ad(IR)→MAR, ACC→MDR, MAR→MM, WR

(2) ADD M 指令: PC→MAR, MAR→MM, MM→MDR, MDR→IR,

OP(IR)→CU, Ad(IR)→MAR, RD, MM→MDR, MDR→X, ADD, ALU→ACC,

ACC→MDR, WR

假设主存容量 256M\*32 位, 在指令字长、存储字长、机器字长相等的条件下, ACC、X、IR、MDR 寄存器均为 32 位, PC 和 MAR 寄存器均为 28 位。

10. 指令和数据都存于存储器中, 计算机如何区分它们?

解: 计算机区分指令和数据有以下 2 种方法:

- 通过不同的时间段来区分指令和数据, 即在取指令阶段(或取指微程序)取出的为指令, 在执行指令阶段(或相应微程序)取出的即为数据。

- 通过地址来源区分, 由 PC 提供存储单元地址的取出的是指令, 由指令地址码部分提供存储单元地址的取出的是操作数。

## 第2章 计算机的发展及应用

1. 通常计算机的更新换代以什么为依据？

答：P22

主要以组成计算机基本电路的元器件为依据，如电子管、晶体管、集成电路等。

2. 举例说明专用计算机和通用计算机的区别。

答：按照计算机的效率、速度、价格和运行的经济性和实用性可以将计算机划分为通用计算机和专用计算机。通用计算机适应性强，但牺牲了效率、速度和经济性，而专用计算机是最有效、最经济和最快的计算机，但适应性很差。例如个人电脑和计算器。

3. 什么是摩尔定律？该定律是否永远生效？为什么？

答：P23，否，P36

## 第3章 系统总线

1. 什么是总线？总线传输有何特点？为了减轻总线负载，总线上的部件应具备什么特点？

答：P41. 总线是多个部件共享的传输部件。

总线传输的特点是：某一时刻只能有一路信息在总线上传输，即分时使用。

为了减轻总线负载，总线上的部件应通过三态驱动缓冲电路与总线连通。

4. 为什么要设置总线判优控制？常见的集中式总线控制有几种？各有何特点？哪种方式响应时间最快？哪种方式对电路故障最敏感？

答：总线判优控制解决多个部件同时申请总线时的使用权分配问题；

常见的集中式总线控制有三种：链式查询、计数器定时查询、独立请求；

特点：链式查询方式连线简单，易于扩充，对电路故障最敏感；计数器定时查询方式优先级设置较灵活，对故障不敏感，连线及控制过程较复杂；独立请求方式速度最快，但硬件器件用量大，连线多，成本较高。

5. 解释下列概念：总线宽度、总线带宽、总线复用、总线的主设备（或主模块）、总线的从设备（或从模块）、总线的传输周期和总线的通信控制。

答：P46。

总线宽度：通常指数据总线的根数；

总线带宽：总线的数据传输率，指单位时间内总线上传输数据的位数；

总线复用：指同一条信号线可以分时传输不同的信号。

总线的主设备（主模块）：指一次总线传输期间，拥有总线控制权的设备（模块）；

总线的从设备（从模块）：指一次总线传输期间，配合主设备完成数据传输的设备（模块），它只能被动接受主设备发来的命令；

总线的传输周期：指总线完成一次完整而可靠的传输所需时间；

总线的通信控制：指总线传送过程中双方的时间配合方式。

6. 试比较同步通信和异步通信。

答：同步通信：指由统一时钟控制的通信，控制方式简单，灵活性差，当系统中各部件工作速度差异较大时，总线工作效率明显下降。适合于速度差别不大的场合。

异步通信：指没有统一时钟控制的通信，部件间采用应答方式进行联系，控制方式较同步复杂，灵活性高，当系统中各部件工作速度差异较大时，有利于提高总线工作效率。

8. 为什么说半同步通信同时保留了同步通信和异步通信的特点？

答：半同步通信既能像同步通信那样由统一时钟控制，又能像异步通信那样允许传输时间不一致，因此工作效率介于两者之间。

10. 为什么要设置总线标准？你知道目前流行的总线标准有哪些？什么叫 plug and play？哪些总线有这一特点？

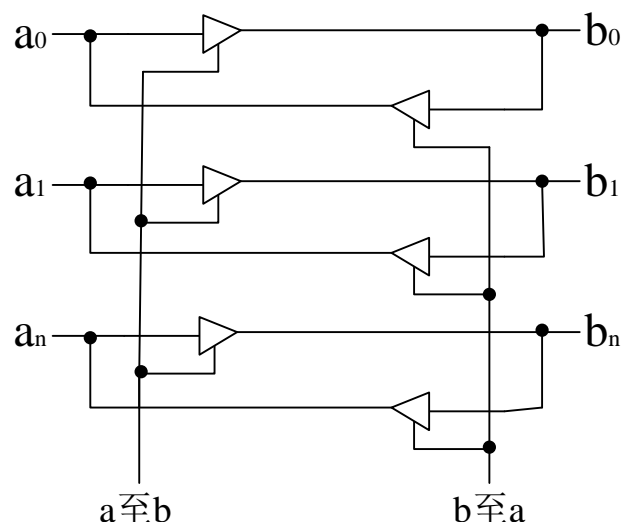
答：总线标准的设置主要解决不同厂家各类模块化产品的兼容问题；

目前流行的总线标准有：ISA、EISA、PCI 等；

plug and play：即插即用，EISA、PCI 等具有此功能。

11. 画一个具有双向传输功能的总线逻辑图。

答：在总线的两端分别配置三态门，就可以使总线具有双向传输功能。



12. 设数据总线上接有 A、B、C、D 四个寄存器，要求选用合适的 74 系列芯片，完成下列逻辑设计：

(1) 设计一个电路，在同一时间实现  $D \rightarrow A$ 、 $D \rightarrow B$  和  $D \rightarrow C$  寄存器间的传送；

(2) 设计一个电路，实现下列操作：

T0 时刻完成  $D \rightarrow$  总线；

T1 时刻完成 总线  $\rightarrow A$ ；

T2 时刻完成  $A \rightarrow$  总线；

T3 时刻完成 总线  $\rightarrow B$ 。

解：(1) 由 T 打开三态门将 D 寄存器中的内容送至总线 bus，由 cp 脉冲同时将总线上的数据打入到 A、B、C 寄存器中。T 和 cp 的时间关系如图 (1) 所示。

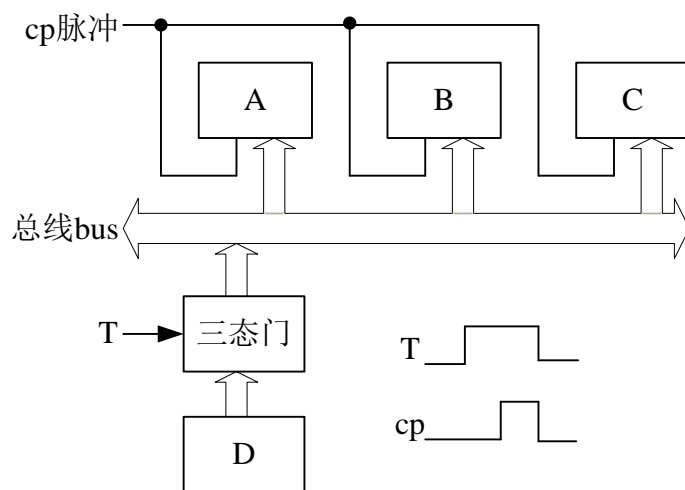
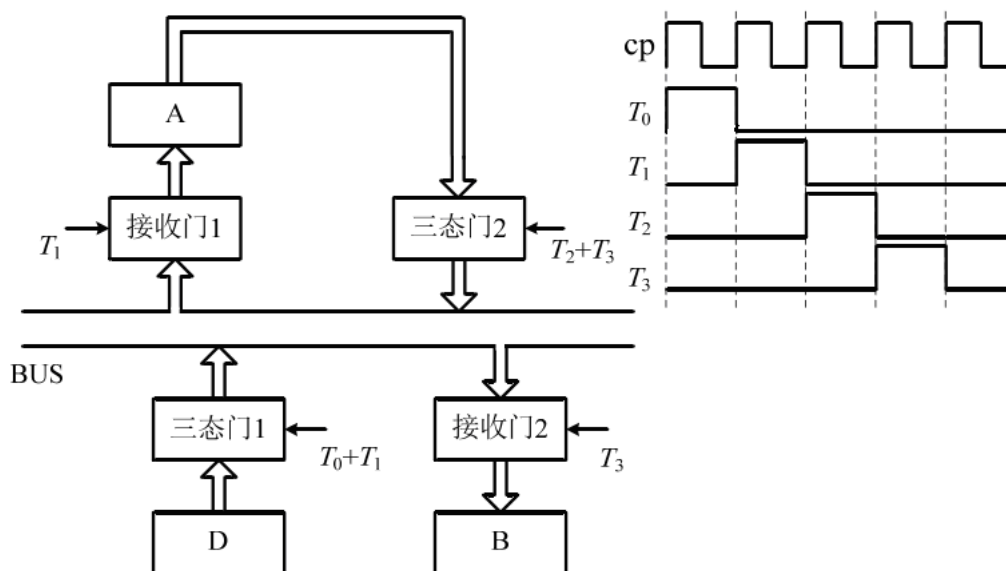


图 (1)

(2) 三态门 1 受  $T_0+T_1$  控制，以确保  $T_0$  时刻  $D \rightarrow$  总线，以及  $T_1$  时刻总线  $\rightarrow$  接收门 1  $\rightarrow A$ 。三态门 2 受  $T_2+T_3$  控制，以确保  $T_2$  时刻  $A \rightarrow$  总线，以及  $T_3$  时刻总线  $\rightarrow$  接收门 2  $\rightarrow B$ 。 $T_0$ 、 $T_1$ 、 $T_2$ 、 $T_3$  波形图如图 (2) 所示。



图(2)

## 第 四 章

3. 存储器的层次结构主要体现在什么地方？为什么要分这些层次？计算机如何管理这些层次？

答：存储器的层次结构主要体现在 Cache-主存和主存-辅存这两个存储层次上。

Cache-主存层次在存储系统中主要对 CPU 访存起加速作用，即从整体运行的效果分析，CPU 访存速度加快，接近于 Cache 的速度，而寻址空间和位价却接近于主存。

主存-辅存层次在存储系统中主要起扩容作用，即从程序员的角度看，他所使用的存储器其容量和位价接近于辅存，而速度接近于主存。

综合上述两个存储层次的作用，从整个存储系统来看，就达到了速度快、容量大、位价低的优化效果。

主存与 CACHE 之间的信息调度功能全部由硬件自动完成。而主存与辅存层次的调度目前广泛

采用虚拟存储技术实现，即将主存与辅存的一部分通过软硬结合的技术组成虚拟存储器，程序员可使用这个比主存实际空间（物理地址空间）大得多的虚拟地址空间（逻辑地址空间）编程，当程序运行时，再由软、硬件自动配合完成虚拟地址空间与主存实际物理空间的转换。因此，这两个层次上的调度或转换操作对于程序员来说都是透明的。

4. 说明存取周期和存取时间的区别。

解：存取周期和存取时间的主要区别是：存取时间仅为完成一次操作的时间，而存取周期不仅包含操作时间，还包含操作后线路的恢复时间。即：

$$\text{存取周期} = \text{存取时间} + \text{恢复时间}$$

5. 什么是存储器的带宽？若存储器的数据总线宽度为 32 位，存取周期为 200ns，则存储器的带宽是多少？

解：存储器的带宽指单位时间内从存储器进出信息的最大数量。

$$\text{存储器带宽} = 1/200\text{ns} \times 32 \text{ 位} = 160\text{M 位/秒} = 20\text{MB/秒} = 5\text{M 字/秒}$$

**注意：**字长 32 位，不是 16 位。（注：1ns=10<sup>-9</sup>s）

6. 某机字长为 32 位，其存储容量是 64KB，按字编址它的寻址范围是多少？若主存以字节编址，试画出主存字地址和字节地址的分配情况。

解：存储容量是 64KB 时，按字节编址的寻址范围就是 64K，如按字编址，其寻址范围为：

$$64\text{K} / (32/8) = 16\text{K}$$

主存字地址和字节地址的分配情况：（略）。

7. 一个容量为 16K×32 位的存储器，其地址线和数据线的总和是多少？当选用下列不同规格的存储芯片时，各需要多少片？

1K×4 位，2K×8 位，4K×4 位，16K×1 位，4K×8 位，8K×8 位

解：地址线和数据线的总和 = 14 + 32 = 46 根；

选择不同的芯片时，各需要的片数为：

$$1\text{K} \times 4: (16\text{K} \times 32) / (1\text{K} \times 4) = 16 \times 8 = 128 \text{ 片}$$

$$2\text{K} \times 8: (16\text{K} \times 32) / (2\text{K} \times 8) = 8 \times 4 = 32 \text{ 片}$$

$$4\text{K} \times 4: (16\text{K} \times 32) / (4\text{K} \times 4) = 4 \times 8 = 32 \text{ 片}$$

$$16\text{K} \times 1: (16\text{K} \times 32) / (16\text{K} \times 1) = 1 \times 32 = 32 \text{ 片}$$

$$4\text{K} \times 8: (16\text{K} \times 32) / (4\text{K} \times 8) = 4 \times 4 = 16 \text{ 片}$$

$$8\text{K} \times 8: (16\text{K} \times 32) / (8\text{K} \times 8) = 2 \times 4 = 8 \text{ 片}$$

8. 试比较静态 RAM 和动态 RAM。

答：略。（参看课件）

9. 什么叫刷新？为什么要刷新？说明刷新有几种方法。

解：刷新：对 DRAM 定期进行的全部重写过程；

刷新原因：因电容泄漏而引起的 DRAM 所存信息的衰减需要及时补充，因此安排了定期刷新操作；

常用的刷新方法有三种：集中式、分散式、异步式。

集中式：在最大刷新间隔时间内，集中安排一段时间进行刷新，存在 CPU 访存死时间。

分散式：在每个读/写周期之后插入一个刷新周期，无 CPU 访存死时间。

异步式：是集中式和分散式的折衷。

10. 半导体存储器芯片的译码驱动方式有几种？

解：半导体存储器芯片的译码驱动方式有两种：线选法和重合法。

线选法：地址译码信号只选中同一个字的所有位，结构简单，费器材；

重合法：地址分行、列两部分译码，行、列译码线的交叉点即为所选单元。这种方法通过行、列译码信号的重合来选址，也称矩阵译码。可大大节省器材用量，是最常用的译码驱动方式。

11. 一个  $8K \times 8$  位的动态 RAM 芯片，其内部结构排列成  $256 \times 256$  形式，存取周期为  $0.1 \mu s$ 。试问采用集中刷新、分散刷新和异步刷新三种方式的刷新间隔各为多少？

解：采用分散刷新方式刷新间隔为： $2ms$ ，其中刷新死时间为： $256 \times 0.1 \mu s = 25.6 \mu s$

采用分散刷新方式刷新间隔为： $256 \times (0.1 \mu s + 0.1 \mu s) = 51.2 \mu s$

采用异步刷新方式刷新间隔为： $2ms$

12. 画出用  $1024 \times 4$  位的存储芯片组成一个容量为  $64K \times 8$  位的存储器逻辑框图。要求将  $64K$  分成 4 个页面，每个页面分 16 组，指出共需多少片存储芯片。

解：设采用 SRAM 芯片，则：

总片数 =  $(64K \times 8 \text{ 位}) / (1024 \times 4 \text{ 位}) = 64 \times 2 = 128$  片

题意分析：本题设计的存储器结构上分为总体、页面、组三级，因此画图时也应分三级画。

首先应确定各级的容量：

页面容量 = 总容量 / 页面数 =  $64K \times 8 / 4 = 16K \times 8$  位，4 片  $16K \times 8$  字串联成  $64K \times 8$  位

组容量 = 页面容量 / 组数 =  $16K \times 8 \text{ 位} / 16 = 1K \times 8$  位，16 片  $1K \times 8$  位字串联成  $16K \times 8$  位

组内片数 = 组容量 / 片容量 =  $1K \times 8 \text{ 位} / 1K \times 4 \text{ 位} = 2$  片，两片  $1K \times 4$  位芯片位并联成  $1K \times 8$  位

存储器逻辑框图：(略)。

13. 设有一个  $64K \times 8$  位的 RAM 芯片，试问该芯片共有多少个基本单元电路（简称存储基元）？欲设计一种具有上述同样多存储基元的芯片，要求对芯片字长的选择应满足地址线和数据线的总和为最小，试确定这种芯片的地址线和数据线，并说明有几种解答。

解：存储基元总数 =  $64K \times 8 \text{ 位} = 512K \text{ 位} = 2^{19}$  位；

**思路：**如要满足地址线和数据线总和最小，应尽量把存储元安排在字向，因为地址位数和字数成 2 的幂的关系，可较好地压缩线数。

解：设地址线根数为  $a$ ，数据线根数为  $b$ ，则片容量为： $2^a \times b = 2^{19}$ ； $b = 2^{19-a}$ ；

若  $a = 19$ ， $b = 1$ ，总和 =  $19+1 = 20$ ；

$a = 18$ ， $b = 2$ ，总和 =  $18+2 = 20$ ；

$a = 17$ ， $b = 4$ ，总和 =  $17+4 = 21$ ；

$a = 16$ ， $b = 8$ ，总和 =  $16+8 = 24$ ；

.....

由上可看出：片字数越少，片字长越长，引脚数越多。片字数减 1、片位数均按 2 的幂变化。

结论：如果满足地址线和数据线的总和为最小，这种芯片的引脚分配方案有两种：地址线 = 19 根，数据线 = 1 根；或地址线 = 18 根，数据线 = 2 根。

14. 某 8 位微型机地址码为 18 位，若使用  $4K \times 4$  位的 RAM 芯片组成模块板结构的存储器，试问：

- (1) 该机所允许的最大主存空间是多少？
- (2) 若每个模块板为  $32K \times 8$  位，共需几个模块板？
- (3) 每个模块板内共有几片 RAM 芯片？
- (4) 共有多少片 RAM？
- (5) CPU 如何选择各模块板？

解：(1) 该机所允许的最大主存空间是： $2^{18} \times 8 \text{ 位} = 256K \times 8 \text{ 位} = 256KB$

(2) 模块板总数 =  $256K \times 8 / 32K \times 8 = 8 \text{ 块}$

(3) 板内片数 =  $32K \times 8 \text{ 位} / 4K \times 4 \text{ 位} = 8 \times 2 = 16 \text{ 片}$

(4) 总片数 =  $16 \text{ 片} \times 8 = 128 \text{ 片}$

(5) CPU 通过最高 3 位地址译码输出选择模板，次高 3 位地址译码输出选择芯片。地址格式分配如下：

模板号 (3位)	芯片号 (3位)	片内地址 (12位)
----------	----------	------------

15. 设 CPU 共有 16 根地址线，8 根数据线，并用  $\overline{MREQ}$  (低电平有效) 作访存控制信号， $R/\overline{W}$

作读写命令信号 (高电平为读，低电平为写)。现有下列存储芯片：ROM ( $2K \times 8$  位， $4K \times 4$  位， $8K \times 8$  位)，RAM ( $1K \times 4$  位， $2K \times 8$  位， $4K \times 8$  位)，及 74138 译码器和其他门电路 (门电路自定)。

试从上述规格中选用合适芯片，画出 CPU 和存储芯片的连接图。要求：

- (1) 最小 4K 地址为系统程序区，4096~16383 地址范围为用户程序区；
- (2) 指出选用的存储芯片类型及数量；
- (3) 详细画出片选逻辑。

解：(1) 地址空间分配图：

系统程序区 (ROM 共 4KB)：0000H-0FFFH

用户程序区 (RAM 共 12KB)：1000H-FFFFH

(2) 选片：ROM：选择  $4K \times 4$  位芯片 2 片，位并联

RAM：选择  $4K \times 8$  位芯片 3 片，字串联 (RAM1 地址范围为：1000H-1FFFH，RAM2 地址范围为 2000H-2FFFH，RAM3 地址范围为：3000H-3FFFH)

(3) 各芯片二进制地址分配如下：

	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1	A0
ROM1, 2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
RAM1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
RAM2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1
RAM3	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1

CPU 和存储器连接逻辑图及片选逻辑如下图 (3) 所示：



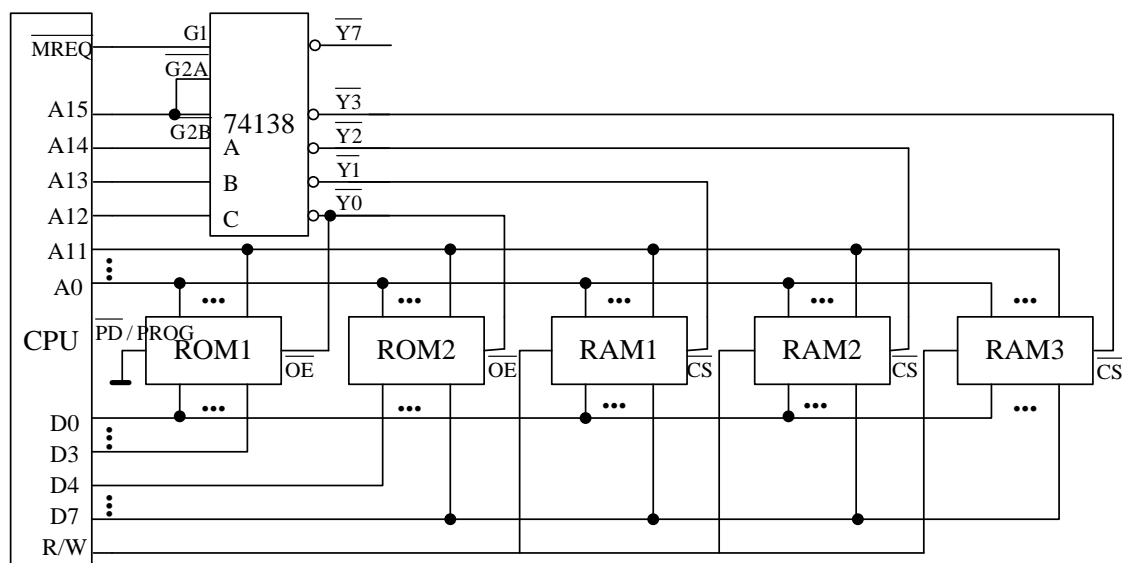
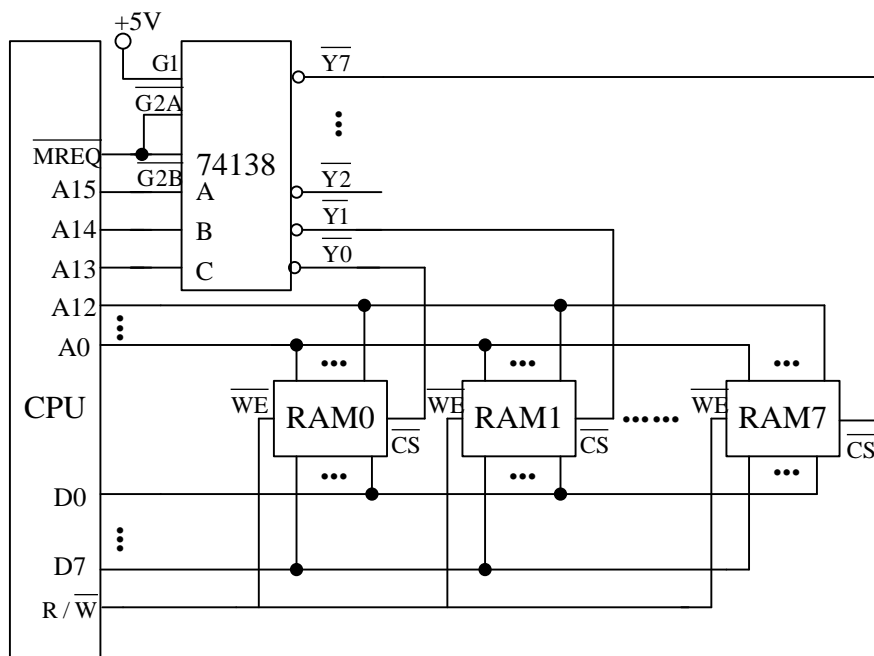


图 (3)

16. CPU 假设同上题，现有 8 片  $8K \times 8$  位的 RAM 芯片与 CPU 相连，试回答：

- (1) 用 74138 译码器画出 CPU 与存储芯片的连接图；
- (2) 写出每片 RAM 的地址范围；
- (3) 如果运行时发现不论往哪片 RAM 写入数据后，以 A000H 为起始地址的存储芯片都有与其相同的数据，分析故障原因。
- (4) 根据 (1) 的连接图，若出现地址线 A13 与 CPU 断线，并搭接到高电平上，将出现什么后果？

解：(1) CPU 与存储器芯片连接逻辑图：



(2) 地址空间分配图：

RAM0: 0000H-1FFFH

RAM1: 2000H-3FFFH

RAM2:4000H-5FFFH  
RAM3:6000H-7FFFH  
RAM4:8000H-9FFFH  
RAM5:A000H-BFFFH  
RAM6:C000H-DFFFH  
RAM7:E000H-FFFFH

(3) 如果运行时发现不论往哪片 RAM 写入数据后, 以 A000H 为起始地址的存储芯片(RAM5) 都有与其相同的数据, 则根本的故障原因为: 该存储芯片的片选输入端很可能总是处于低电平。假设芯片与译码器本身都是好的, 可能的情况有:

- 1) 该片的-CS 端与-WE 端错连或短路;
- 2) 该片的-CS 端与 CPU 的-MREQ 端错连或短路;
- 3) 该片的-CS 端与地线错连或短路。

(4) 如果地址线 A13 与 CPU 断线, 并搭接到高电平上, 将会出现 A13 恒为“1”的情况。此时存储器只能寻址 A13=1 的地址空间(奇数片), A13=0 的另一半地址空间(偶数片) 将永远访问不到。若对 A13=0 的地址空间(偶数片) 进行访问, 只能错误地访问到 A13=1 的对应空间(奇数片) 中去。

17. 写出 1100、1101、1110、1111 对应的汉明码。

解: 有效信息均为  $n=4$  位, 假设有效信息用  $b_4b_3b_2b_1$  表示

校验位位数  $k=3$  位, ( $2^k \geq n+k+1$ )

设校验位分别为  $c_1$ 、 $c_2$ 、 $c_3$ , 则汉明码共  $4+3=7$  位, 即:  $c_1c_2b_4c_3b_3b_2b_1$

校验位在汉明码中分别处于第 1、2、4 位

$$c_1 = b_4 \oplus b_3 \oplus b_1$$

$$c_2 = b_4 \oplus b_2 \oplus b_1$$

$$c_3 = b_3 \oplus b_2 \oplus b_1$$

当有效信息为 1100 时,  $c_3c_2c_1=011$ , 汉明码为 1110100。

当有效信息为 1101 时,  $c_3c_2c_1=100$ , 汉明码为 0011101。

当有效信息为 1110 时,  $c_3c_2c_1=101$ , 汉明码为 1011110。

当有效信息为 1111 时,  $c_3c_2c_1=010$ , 汉明码为 0110111。

18. 已知收到的汉明码(按配偶原则配置) 为 1100100、1100111、1100000、1100001, 检查上述代码是否出错? 第几位出错?

解: 假设接收到的汉明码为:  $c_1' c_2' b_4' c_3' b_3' b_2' b_1'$

纠错过程如下:

$$P_1 = c_1' \oplus b_4' \oplus b_3' \oplus b_1'$$

$$P_2 = c_2' \oplus b_4' \oplus b_2' \oplus b_1'$$

$$P_3 = c_3' \oplus b_3' \oplus b_2' \oplus b_1'$$

如果收到的汉明码为 1100100, 则  $p_3p_2p_1=011$ , 说明代码有错, 第 3 位( $b_4'$ ) 出错, 有效信息为: 1100

如果收到的汉明码为 1100111, 则  $p_3p_2p_1=111$ , 说明代码有错, 第 7 位( $b_1'$ ) 出错, 有效信息为: 0110

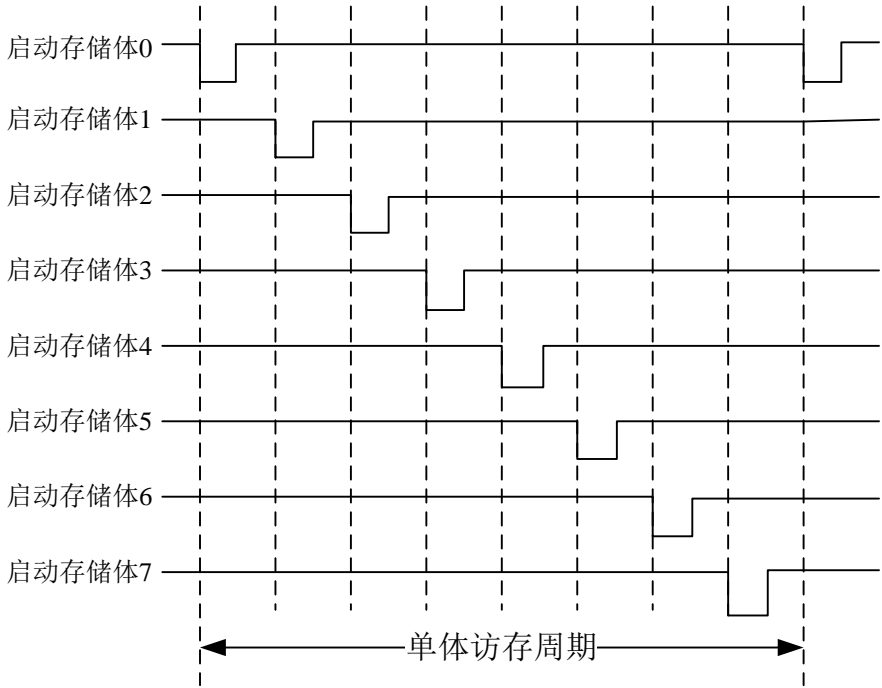
如果收到的汉明码为 1100000, 则  $p_3p_2p_1=110$ , 说明代码有错, 第 6 位( $b_2'$ ) 出错, 有效信息为: 0010

如果收到的汉明码为 1100001, 则  $p_3p_2p_1=001$ , 说明代码有错, 第 1 位( $c_1'$ ) 出错, 有

效信息为：0001

22. 某机字长 16 位，常规的存储空间为 64K 字，若想不改用其他高速的存储芯片，而使访存速度提高到 8 倍，可采取什么措施？画图说明。

解：若想不改用高速存储芯片，而使访存速度提高到 8 倍，可采取八体交叉存取技术，8 体交叉访问时序如下图：



18. 什么是“程序访问的局部性”？存储系统中哪一级采用了程序访问的局部性原理？

解：程序运行的局部性原理指：在一小段时间内，最近被访问过的程序和数据很可能再次被访问；在空间上，这些被访问的程序和数据往往集中在一小片存储区；在访问顺序上，指令顺序执行比转移执行的可能性大（大约 5:1）。存储系统中 Cache—主存层次采用了程序访问的局部性原理。

25. Cache 做在 CPU 芯片内有什么好处？将指令 Cache 和数据 Cache 分开又有什么好处？

答：Cache 做在 CPU 芯片内主要有下面几个好处：

1) 可提高外部总线的利用率。因为 Cache 在 CPU 芯片内，CPU 访问 Cache 时不必占用外部总线。

2) Cache 不占用外部总线就意味着外部总线可更多地支持 I/O 设备与主存的信息传输，增强了系统的整体效率。

3) 可提高存取速度。因为 Cache 与 CPU 之间的数据通路大大缩短，故存取速度得以提高。

将指令 Cache 和数据 Cache 分开有如下好处：

1) 可支持超前控制和流水线控制，有利于这类控制方式下指令预取操作的完成。

2) 指令 Cache 可用 ROM 实现，以提高指令存取的可靠性。

3) 数据 Cache 对不同数据类型的支持更为灵活，既可支持整数（例 32 位），也可支持浮点数据（如 64 位）。

**补充：**

Cache 结构改进的第三个措施是分级实现，如二级缓存结构，即在片内 Cache（L1）和主存之间再设一个片外 Cache（L2），片外缓存既可以弥补片内缓存容量不够大的缺点，又可在主存

与片内缓存间起到平滑速度差的作用，加速片内缓存的调入调出速度。

30. 一个组相连映射的 CACHE 由 64 块组成，每组内包含 4 块。主存包含 4096 块，每块由 128 字组成，访存地址为字地址。试问主存和高速存储器的地址各为几位？画出主存地址格式。

解：cache 组数：64/4=16，Cache 容量为：64\*128=2<sup>13</sup>字，cache 地址 13 位

主存共分 4096/16=256 区，每区 16 块

主存容量为：4096\*128=2<sup>19</sup>字，主存地址 19 位，地址格式如下：

主存字块标记（8 位）	组地址（4 位）	字块内地址（7 位）
-------------	----------	------------

## 第 六 章

12. 设浮点数格式为：阶码 5 位（含 1 位阶符），尾数 11 位（含 1 位数符）。写出 51/128、-27/1024 所对应的机器数。要求如下：

- （1）阶码和尾数均为原码。
- （2）阶码和尾数均为补码。
- （3）阶码为移码，尾数为补码。

解：据题意画出该浮点数的格式：

阶符 1 位	阶码 4 位	数符 1 位	尾数 10 位
--------	--------	--------	---------

将十进制数转换为二进制：x1= 51/128= 0.0110011B= 2<sup>-1</sup> \* 0.110 011B

x2= -27/1024= -0.0000011011B = 2<sup>-5</sup>\*(-0.11011B)

则以上各数的浮点规格化数为：

- （1）[x1]浮=1, 0001; 0.110 011 000 0  
[x2]浮=1, 0101; 1.110 110 000 0
- （2）[x1]浮=1, 1111; 0.110 011 000 0  
[x2]浮=1, 1011; 1.001 010 000 0
- （3）[x1]浮=0, 1111; 0.110 011 000 0  
[x2]浮=0, 1011; 1.001 010 000 0

16. 设机器数字长为 16 位，写出下列各种情况下它能表示的数的范围。设机器数采用一位符号位，答案均用十进制表示。

- （1）无符号数；
- （2）原码表示的定点小数。
- （3）补码表示的定点小数。
- （4）补码表示的定点整数。
- （5）原码表示的定点整数。

（6）浮点数的格式为：阶码 6 位（含 1 位阶符），尾数 10 位（含 1 位数符）。分别写出其正数和负数的表示范围。

（7）浮点数格式同（6），机器数采用补码规格化形式，分别写出其对应的正数和负数的真值范围。

解：（1）无符号整数：0 —— 2<sup>16</sup> - 1，即：0 —— 65535；

无符号小数：0 —— 1 - 2<sup>-16</sup>，即：0 —— 0.99998；

(2) 原码定点小数:  $-1 + 2^{-15}$  ——  $1 - 2^{-15}$ , 即:  $-0.99997$  ——  $0.99997$

(3) 补码定点小数:  $-1$  ——  $1 - 2^{-15}$ , 即:  $-1$  ——  $0.99997$

(4) 补码定点整数:  $-2^{15}$  ——  $2^{15} - 1$ , 即:  $-32768$  ——  $32767$

(5) 原码定点整数:  $-2^{15} + 1$  ——  $2^{15} - 1$ , 即:  $-32767$  ——  $32767$

(6) 据题意画出该浮点数格式, 当阶码和尾数均采用原码, 非规格化数表示时:

最大负数= 1, 11 111; 1.000 000 001, 即  $-2^{-9} \times 2^{-31}$

最小负数= 0, 11 111; 1.111 111 111, 即  $-(1-2^{-9}) \times 2^{31}$

则负数表示范围为:  $-(1-2^{-9}) \times 2^{31}$  ——  $-2^{-9} \times 2^{-31}$

最大正数= 0, 11 111; 0.111 111 111, 即  $(1-2^{-9}) \times 2^{31}$

最小正数= 1, 11 111; 0.000 000 001, 即  $2^{-9} \times 2^{-31}$

则正数表示范围为:  $2^{-9} \times 2^{-31}$  ——  $(1-2^{-9}) \times 2^{31}$

(7) 当机器数采用补码规格化形式时, 若不考虑隐藏位, 则

最大负数=1, 00 000; 1.011 111 111, 即  $-2^{-1} \times 2^{-32}$

最小负数=0, 11 111; 1.000 000 000, 即  $-1 \times 2^{31}$

则负数表示范围为:  $-1 \times 2^{31}$  ——  $-2^{-1} \times 2^{-32}$

最大正数=0, 11 111; 0.111 111 111, 即  $(1-2^{-9}) \times 2^{31}$

最小正数=1, 00 000; 0.100 000 000, 即  $2^{-1} \times 2^{-32}$

则正数表示范围为:  $2^{-1} \times 2^{-32}$  ——  $(1-2^{-9}) \times 2^{31}$

17. 设机器数字长为 8 位 (包括一位符号位), 对下列各机器数进行算术左移一位、两位, 算术右移一位、两位, 讨论结果是否正确。

[x1]原=0.001 1010; [y1]补=0.101 0100; [z1]反=1.010 1111;

[x2]原=1.110 1000; [y2]补=1.110 1000; [z2]反=1.110 1000;

[x3]原=1.001 1001; [y3]补=1.001 1001; [z3]反=1.001 1001。

解: 算术左移一位:

[x1]原=0.011 0100; 正确

[x2]原=1.101 0000; 溢出 (丢 1) 出错

[x3]原=1.011 0010; 正确

[y1]补=0.010 1000; 溢出 (丢 1) 出错

[y2]补=1.101 0000; 正确

[y3]补=1.011 0010; 溢出 (丢 0) 出错

[z1]反=1.101 1111; 溢出 (丢 0) 出错

[z2]反=1.101 0001; 正确

[z3]反=1.011 0011; 溢出 (丢 0) 出错

算术左移两位:

[x1]原=0.110 1000; 正确

[x2]原=1.010 0000; 溢出 (丢 11) 出错

[x3]原=1.110 0100; 正确

[y1]补=0.101 0000; 溢出 (丢 10) 出错

[y2]补=1.010 0000; 正确

[y3]补=1.110 0100; 溢出 (丢 00) 出错

[z1]反=1.011 1111; 溢出 (丢 01) 出错

[z2]反=1.010 0011; 正确

[z3]反=1.110 0111; 溢出 (丢 00) 出错

算术右移一位:

[x1]原=0.000 1101; 正确  
[x2]原=1.011 0100; 正确  
[x3]原=1.000 1100(1); 丢 1, 产生误差  
[y1]补=0.010 1010; 正确  
[y2]补=1.111 0100; 正确  
[y3]补=1.100 1100(1); 丢 1, 产生误差  
[z1]反=1.101 0111; 正确  
[z2]反=1.111 0100(0); 丢 0, 产生误差  
[z3]反=1.100 1100; 正确

算术右移两位:

[x1]原=0.000 0110 (10); 产生误差  
[x2]原=1.001 1010; 正确  
[x3]原=1.000 0110 (01); 产生误差  
[y1]补=0.001 0101; 正确  
[y2]补=1.111 1010; 正确  
[y3]补=1.110 0110 (01); 产生误差  
[z1]反=1.110 1011; 正确  
[z2]反=1.111 1010 (00); 产生误差  
[z3]反=1.110 0110 (01); 产生误差

19. 设机器数字长为 8 位 (含 1 位符号位), 用补码运算规则计算下列各题。

- (1)  $A=9/64$ ,  $B=-13/32$ , 求  $A+B$ 。
- (2)  $A=19/32$ ,  $B=-17/128$ , 求  $A-B$ 。
- (3)  $A=-3/16$ ,  $B=9/32$ , 求  $A+B$ 。
- (4)  $A=-87$ ,  $B=53$ , 求  $A-B$ 。
- (5)  $A=115$ ,  $B=-24$ , 求  $A+B$ 。

解: (1)  $A=9/64=0.001\ 0010B$ ,  $B=-13/32=-0.011\ 0100B$

[A]补=0.001 0010, [B]补=1.100 1100

[A+B]补= 0.0010010 + 1.1001100 = 1.1011110 ——无溢出

$A+B=-0.010\ 0010B=-17/64$

(2)  $A=19/32=0.100\ 1100B$ ,  $B=-17/128=-0.001\ 0001B$

[A]补=0.100 1100, [B]补=1.110 1111, [-B]补=0.001 0001

[A-B]补= 0.1001100 + 0.0010001= 0.1011101 ——无溢出

$A-B=0.101\ 1101B=93/128B$

(3)  $A=-3/16=-0.001\ 1000B$ ,  $B=9/32=0.010\ 0100B$

[A]补=1.110 1000, [B]补= 0.010 0100

[A+B]补= 1.1101000 + 0.0100100 = 0.0001100 —— 无溢出

$A+B=0.000\ 1100B=3/32$

(4)  $A=-87=-101\ 0111B$ ,  $B=53=110\ 101B$

[A]补=1 010 1001, [B]补=0 011 0101, [-B]补=1 100 1011

[A-B]补= 1 0101001 + 1 1001011 = 0 1110100 —— 溢出

(5)  $A=115=111\ 0011B$ ,  $B=-24=-11\ 000B$

[A]补=0 1110011, [B]补=1, 110 1000

$$[A+B]_{\text{补}} = 0\ 1110011 + 1\ 1101000 = 0\ 1011011 \text{——无溢出}$$

$$A+B = 101\ 1011B = 91$$

26. 按机器补码浮点运算步骤, 计算 $[x \pm y]_{\text{补}}$ .

- (1)  $x=2^{-011} \times 0.101\ 100$ ,  $y=2^{-010} \times (-0.011\ 100)$ ;  
 (2)  $x=2^{-011} \times (-0.100\ 010)$ ,  $y=2^{-010} \times (-0.011\ 111)$ ;  
 (3)  $x=2^{101} \times (-0.100\ 101)$ ,  $y=2^{100} \times (-0.001\ 111)$ 。

解: 先将  $x$ 、 $y$  转换成机器数形式:

(1)  $x=2^{-011} \times 0.101\ 100$ ,  $y=2^{-010} \times (-0.011\ 100)$

$$[x]_{\text{补}}=1, 101; 0.101\ 100, [y]_{\text{补}}=1, 110; 1.100\ 100$$

$$[Ex]_{\text{补}}=1, 101, [y]_{\text{补}}=1, 110, [Mx]_{\text{补}}=0.101\ 100, [My]_{\text{补}}=1.100\ 100$$

1) 对阶:

$$[\Delta E]_{\text{补}}=[Ex]_{\text{补}}+[-Ey]_{\text{补}} = 11, 101+ 00, 010=11, 111 < 0,$$

应  $Ex$  向  $Ey$  对齐, 则:  $[Ex]_{\text{补}}+1=11, 101+00, 001=11, 110 = [Ey]_{\text{补}}$

$$[x]_{\text{补}}=1, 110; 0.010\ 110$$

2) 尾数运算:

$$[Mx]_{\text{补}}+[My]_{\text{补}}= 0.010\ 110 + 11.100\ 100=11.111010$$

$$[Mx]_{\text{补}}+[-My]_{\text{补}}=0.010\ 110 + 00.011100= 00.110\ 010$$

3) 结果规格化:

$$[x+y]_{\text{补}}=11, 110; 11.111\ 010 = 11, 011; 11.010\ 000 \quad (\text{尾数左规 3 次, 阶码减 3})$$

$$[x-y]_{\text{补}}=11, 110; 00.110\ 010, \text{已是规格化数。}$$

4) 舍入: 无

5) 溢出: 无

$$\text{则: } x+y=2^{-101} \times (-0.110\ 000)$$

$$x-y = 2^{-010} \times 0.110\ 010$$

(2)  $x=2^{-011} \times (-0.100010)$ ,  $y=2^{-010} \times (-0.011111)$

$$[x]_{\text{补}}=1, 101; 1.011\ 110, [y]_{\text{补}}=1, 110; 1.100\ 001$$

1) 对阶: 过程同(1)的 1), 则

$$[x]_{\text{补}}=1, 110; 1.101\ 111$$

2) 尾数运算:

$$[Mx]_{\text{补}}+[My]_{\text{补}}= 11.101111 + 11.100001 = 11.010000$$

$$[Mx]_{\text{补}}+[-My]_{\text{补}}= 11.101111 + 00.011111 = 00.001110$$

3) 结果规格化:

$$[x+y]_{\text{补}}=11, 110; 11.010\ 000, \text{已是规格化数}$$

$$[x-y]_{\text{补}}=11, 110; 00.001\ 110 = 11, 100; 00.111000 \quad (\text{尾数左规 2 次, 阶码减 2})$$

4) 舍入: 无

5) 溢出: 无

$$\text{则: } x+y=2^{-010} \times (-0.110\ 000)$$

$$x-y = 2^{-100} \times 0.111\ 000$$

(3)  $x=2^{101} \times (-0.100\ 101)$ ,  $y=2^{100} \times (-0.001\ 111)$

$$[x]_{\text{补}}=0, 101; 1.011\ 011, [y]_{\text{补}}=0, 100; 1.110\ 001$$

1) 对阶:



$[\Delta E]_{\text{补}}=00, 101+11, 100=00, 001 >0$ , 应  $E_y$  向  $E_x$  对齐, 则:

$[E_y]_{\text{补}}+1=00, 100+00, 001=00, 101=[E_x]_{\text{补}}$

$[y]_{\text{补}}=0, 101; 1.111\ 000\ (1)$

2) 尾数运算:

$[M_x]_{\text{补}}+[M_y]_{\text{补}}= 11.011011+ 11.111000\ (1) = 11.010011\ (1)$

$[M_x]_{\text{补}}+[-M_y]_{\text{补}}= 11.011011+ 00.000111\ (1) = 11.100010\ (1)$

2) 结果规格化:

$[x+y]_{\text{补}}=00, 101; 11.010\ 011\ (1)$ , 已是规格化数

$[x-y]_{\text{补}}=00, 101; 11.100\ 010\ (1) =00, 100; 11.000\ 101$  (尾数左规 1 次, 阶码

减 1)

4) 舍入:

$[x+y]_{\text{补}}=00, 101; 11.010\ 011$  (舍)

$[x-y]_{\text{补}}$  不变

5) 溢出: 无

则:  $x+y=2^{101} \times (-0.101\ 101)$

$x-y=2^{100} \times (-0.111\ 011)$

32. 设机器字长为 16 位, 分别按 4、4、4、4 和 5、5、3、3 分组后,

(1) 画出按两种分组方案的单重分组并行进位链框图, 并比较哪种方案运算速度快。

(2) 画出按两种分组方案的双重分组并行进位链框图, 并对这两种方案进行比较。

(3) 用 74181 和 74182 画出单重和双重分组的并行进位链框图。

解: (1) 4—4—4—4 分组的 16 位单重分组并行进位链框图见教材 286 页图 6.22。

5—5—3—3 分组的 16 位单重分组并行进位链框图如下:

(2) 4—4—4—4 分组的 16 位双重分组并行进位链框图见教材 289 页图 6.26。

5—5—3—3 分组的 16 位双重分组并行进位链框图如下:

5—5—3—3 分组的进位时间  $=2.5t_{\text{y}} \times 3 = 7.5t_{\text{y}}$ ;

4—4—4—4 分组的进位时间  $=2.5t_{\text{y}} \times 3 = 7.5t_{\text{y}}$ ;

可见, 两种分组方案最长加法时间相同。

结论: 双重分组并行进位的最长进位时间只与组数和级数有关, 与组内位数无关。

(3) 单重分组 16 位并行加法器逻辑图如下 (正逻辑):

**注意:** 1) 74181 芯片正、负逻辑的引脚表示方法;

2) 为强调可比性, 5-5-3-3 分组时不考虑扇入影响;

3) 181 芯片只有最高、最低两个进位输入/输出端, 组内进位无引脚;

4) 181 为 4 位片, 无法 5-5-3-3 分组, 只能 4-4-4-4 分组;

5) 单重分组跳跃进位只用到 181, 使用 182 的一定是双重以上分组跳跃进位;

6) 单重分组跳跃进位是并行进位和串行进位技术的结合; 双重分组跳跃进位是二级并行进位技术; 特别注意在位数较少时, 双重分组跳跃进位可以采用全先行进位技术实现; 位数较多时, 可采用双重分组跳跃进位和串行进位技术结合实现。

## 第 七 章

1. 什么叫机器指令？什么叫指令系统？为什么说指令系统与机器的主要功能以及与硬件结构之间存在着密切的关系？

答：参考 P300。

2. 什么叫寻址方式？为什么要学习寻址方式？

答：参看 P310。

3. 什么是指令字长、机器字长和存储字长？

答：略。

4. 零地址指令的操作数来自哪里？？各举一例说明。

答：零地址指令的操作数来自 ACC，为隐含约定。

在一地址指令中，另一个操作数的地址通常可采用 ACC 隐含寻址方式获得。

5. 对于二地址指令而言，操作数的物理地址可安排在什么地方？举例说明。

答：对于二地址指令而言，操作数的物理地址可安排在寄存器内、指令中或内存单元内等。

8. 某机指令字长 16 位，每个操作数的地址码为 6 位，设操作码长度固定，指令分为零地址、一地址和二地址三种格式。若零地址指令有 M 条，一地址指令有 N 种，则二地址指令最多有几种？若操作码位数可变，则二地址指令最多允许有几种？

解：1) 若采用定长操作码时，二地址指令格式如下：

OP (4 位)	A1 (6 位)	A2 (6 位)
----------	----------	----------

设二地址指令有 K 种，则： $K=2^4-M-N$

当  $M=1$  (最小值)， $N=1$  (最小值) 时，二地址指令最多有： $K_{\max}=16-1-1=14$  种

3) 若采用变长操作码时，二地址指令格式仍如 1) 所示，但操作码长度可随地址码的个数而变。此时， $K=2^4-(N/2^6 + M/2^{12})$ ；

当  $(N/2^6 + M/2^{12}) \leq 1$  时 ( $N/2^6 + M/2^{12}$  向上取整)，K 最大，则二地址指令最多有： $K_{\max}=16-1=15$  种 (只留一种编码作扩展标志用。)

9. 试比较间接寻址和寄存器间接寻址。

答：略。

10. 试比较基址寻址和变址寻址。

略。

11. 画出先变址再间址及先间址再变址的寻址过程示意图。

解：1) 先变址再间址寻址过程简单示意如下：

$$EA = [(IX) + A], IX \rightarrow (IX) + 1$$

2) 先间址再变址寻址过程简单示意如下： $EA = (IX) + (A)$ ， $IX \rightarrow (IX) + 1$

16. 某机主存容量为  $4M \times 16$  位，且存储字长等于指令字长，若该机指令系统可完成 108 种操作，

操作码位数固定，且具有直接、间接、变址、基址、相对、立即等六种寻址方式，试回答：（1）画出一地址指令格式并指出各字段的作用；

- （2）该指令直接寻址的最大范围；
- （3）一次间址和多次间址的寻址范围；
- （4）立即数的范围（十进制表示）；
- （5）相对寻址的位移量（十进制表示）；
- （6）上述六种寻址方式的指令哪一种执行时间最短？哪一种最长？为什么？哪一种便于程序浮动？哪一种最适合处理数组问题？
- （7）如何修改指令格式，使指令的寻址范围可扩大到 4M？
- （8）为使一条转移指令能转移到主存的任一位置，可采取什么措施？简要说明之。

解：（1）单字长一地址指令格式：

OP（7 位）	M（3 位）	A（6 位）
---------	--------	--------

OP 为操作码字段，共 7 位，可反映 108 种操作；

M 为寻址方式字段，共 3 位，可反映 6 种寻址操作；

A 为地址码字段，共  $16-7-3=6$  位。

- （2）直接寻址的最大范围为  $2^6=64$ 。
- （3）由于存储字长为 16 位，故一次间址的寻址范围为  $2^{16}$ ；若多次间址，需用存储字的最高位来区别是否继续间接寻址，故寻址范围为  $2^{15}$ 。
- （4）立即数的范围为 -32——31（有符号数），或 0——63（无符号数）。
- （5）相对寻址的位移量为 -32——31。
- （6）上述六种寻址方式中，因立即数由指令直接给出，故立即寻址的指令执行时间最短。间接寻址在指令的执行阶段要多次访存（一次间接寻址要两次访存，多次间接寻址要多次访存），故执行时间最长。变址寻址由于变址寄存器的内容由用户给定，而且在程序的执行过程中允许用户修改，而其形式地址始终不变，故变址寻址的指令便于用户编制处理数组问题的程序。相对寻址操作数的有效地址只与当前指令地址相差一定的位移量，与直接寻址相比，更有利于程序浮动。
- （7）**方案一**：为使指令寻址范围可扩大到 4M，需要有效地址 22 位，此时可将单字长一地址指令的格式改为双字长，如下图示：

OP（7 位）	MOD（3 位）	A（高 6 位）
A（低 16 位）		

**方案二**：如果仍采用单字长指令（16 位）格式，为使指令寻址范围扩大到 4M，可通过段寻址方案实现。安排如下：

硬件设段寄存器 DS（16 位），用来存放段地址。在完成指令寻址方式所规定的寻址操作后，得有效地址 EA（6 位），再由硬件自动完成段寻址，最后得 22 位物理地址。即：物理地址 =  $(DS) \times 2^6 + EA$

注：段寻址方式由硬件隐含实现。在编程指定的寻址过程完成、EA 产生之后由硬件自动完成，对用户是透明的。

**方案三**：在采用单字长指令（16 位）格式时，还可通过页面寻址方案使指令寻址范围扩大到 4M。安排如下：

硬件设页面寄存器 PR（16 位），用来存放页面地址。指令寻址方式中增设页面寻址。当需要使指令寻址范围扩大到 4M 时，编程选择页面寻址方式，则：EA = (PR) || A（有效地址 = 页面地址“拼接”6 位形式地址），这样得到 22 位有效地址。

（8）为使一条转移指令能转移到主存的任一位置，寻址范围须达到 4M，除了采用（7）方案一中的双字长一地址指令的格式外，还可配置 22 位的基址寄存器或 22 位的变址寄存器，使 EA =

$(BR) + A$  (BR 为 22 位的基址寄存器) 或  $EA = (IX) + A$  (IX 为 22 位的变址寄存器), 便可访问 4M 存储空间。还可以通过 16 位的基址寄存器左移 6 位再和形式地址 A 相加, 也可达到同样的效果。

总之, 不论采取何种方式, 最终得到的实际地址应是 22 位。

19. 某 CPU 内有 32 个 32 位的通用寄存器, 设计一种能容纳 64 种操作的指令系统。假设指令字长等于机器字长, 试回答以下问题:

(1) 如果主存可直接或间接寻址, 采用寄存器—存储器型指令, 能直接寻址的最大存储空间是多少? 画出指令格式并说明各字段的含义。

(2) 在满足 (1) 的前提下, 如果采用通用寄存器作基址寄存器, 则上述寄存器—存储器型指令的指令格式有何特点? 画出指令格式并指出这类指令可访问多大的存储空间?

解: (1) 如采用 RS 型指令, 则此指令一定是二地址以上的地址格式, 指令格式如下:

OP (6 位)	R (5 位)	I (1 位)	A (20 位)
----------	---------	---------	----------

操作码字段 OP 占 6 位, 因为  $2^6=64$ ;

寄存器编号 R 占 5 位, 因为  $2^5=32$ ;

间址位 I 占 1 位, 当 I=0, 存储器寻址的操作数为直接寻址, 当 I=1 时为间接寻址;

形式地址 A 占 20 位, 可以直接寻址  $2^{20}$  字。

(2) 如采用基址寻址, 则指令格式中应给出基址寄存器号, 以指定哪一个通用寄存器用作基址寄存器。指令格式变为:

OP (6 位)	源 R (5 位)	I (1 位)	X (1 位)	目标 R (5 位)	A (14 位)
----------	-----------	---------	---------	------------	----------

增加寻址特征位 X, 当 X=1 时, 以目标寄存器 R 作为基址寄存器进行基址寻址。

基址寻址可访问存储空间为:  $2^{32}$  字。

## 第八章

1. CPU 有哪些功能? 画出其结构框图并简要说明各个部件的作用。

答: 参考 P328 和图 8.2。

2. 什么是指令周期? 指令周期是否有一个固定值? 为什么?

解: 指令周期是指取出并执行完一条指令所需的时间。

由于计算机中各种指令执行所需的时间差异很大, 因此为了提高 CPU 运行效率, 即使在同步控制的机器中, 不同指令的指令周期长度都是不一致的, 也就是说指令周期对于不同的指令来说不是一个固定值。

3. 画出指令周期的流程图, 分析说明图中每个子周期的作用。

答: 参看 P343 及图 8.8。

4. 设 CPU 内有下列部件: PC、IR、SP、AC、MAR、MDR 和 CU。

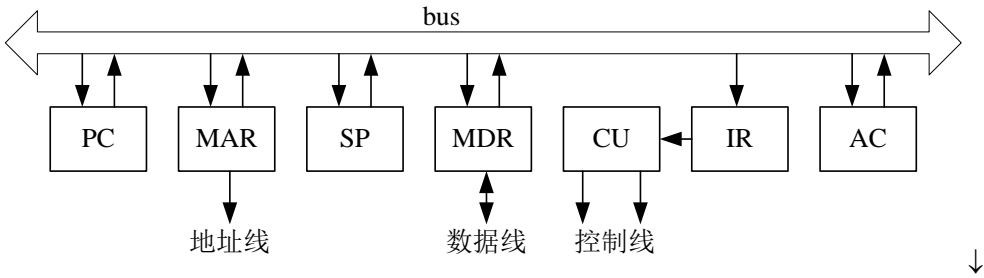
(1) 画出完成间接寻址的取数指令 LDA@X (将主存某地址单元 X 的内容取至 AC 中) 的数据流 (从取指令开始)。

(2) 画出中断周期的数据流。

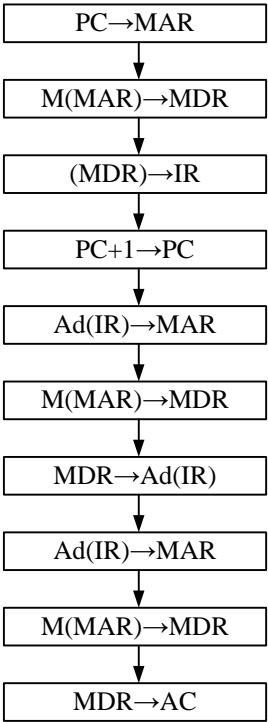
解: CPU 中的数据流向与所采用的数据通路结构直接相关, 不同的数据通路中的数据流是不一样的。

的。常用的数据通路结构方式有直接连线、单总线、双总线、三总线等形式，目前大多采用总线结构，直接连线方式仅适用于结构特别简单的机器中。

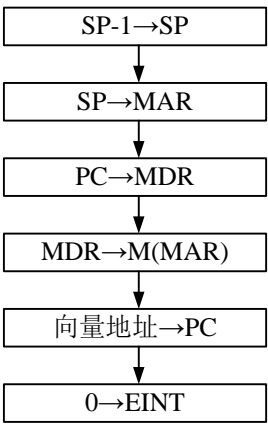
为简单起见，本题采用单总线将题中所给部件连接起来，框图如下：



(1) LDA@X 指令周期数据流程图：



(2) 中断周期流程图如下：



**注：**解这道题有两个要素，首先要根据所给部件设计好数据通路，即确定信息流动的载体。其次选择好描述数据流的方法，无论采用什么样的表达方式，其关键都要能清楚地反映数据在通路上流动的顺序，即强调一个“流”字。较好的表达方式是流程图的形式。

5. 中断周期前是什么阶段？中断周期后又是什么阶段？在中断周期 CPU 应完成什么操作？

答：中断周期前是执行周期，中断周期后是取指周期。在中断周期，CPU 应完成保存断点、将中断向量送 PC 和关中断等工作。

7. 什么叫系统的并行性？粗粒度并行和细粒度并行有何区别？

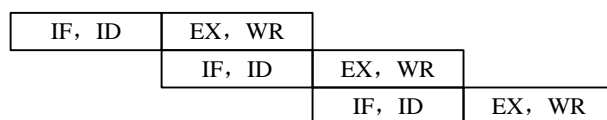
答：所谓并行性包含同时性和并发性。同时性是指两个或两个以上的事件在同一时刻发生，并发性是指两个或多个事件在同一时间段发生。即在同一时刻或同一时间段内完成两个或两个以上性质相同或性质不同的功能，只要在时间上存在相互重叠，就存在并行性。

并行性又分为粗粒度并行和细粒度并行两类。粗粒度并行是指在多个处理机上分别运行多个进程，由多台处理机合作完成一个程序，一般用算法实现。细粒度并行是指在处理机的指令级和操作级的并行性。

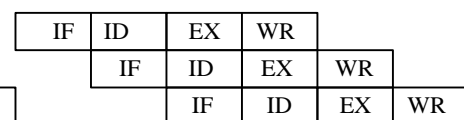
8. 什么是指令流水？画出指令二级流水和四级流水的示意图，它们中哪个更能提高处理机速度，为什么？

答：指令流水是指将一条指令的执行过程分为  $n$  个操作时间大致相等的阶段，每个阶段由一个独立的功能部件来完成，这样  $n$  个部件就可以同时执行  $n$  条指令的不同阶段，从而大大提高 CPU 的吞吐率。

指令二级流水和四级流水示意图如下：



二级指令流水示意图



四级指令流水示意图

四级流水更能提高处理机的速度。分析如下：

假设 IF、ID、EX、WR 每个阶段耗时为  $t$ ，则连续执行  $n$  条指令

采用二级流水线时，耗时为： $4t+(n-1)2t=(2n+2)t$

采用四级流水线时，耗时为： $4t+(n-1)t=(n+3)t$

在  $n>1$  时， $n+3<2n+2$ ，可见四级流水线耗时比二级流水线耗时短，因此更能提高处理机速度。

17. 在中断系统中 INTR、INT、EINT 三个触发器各有何作用？

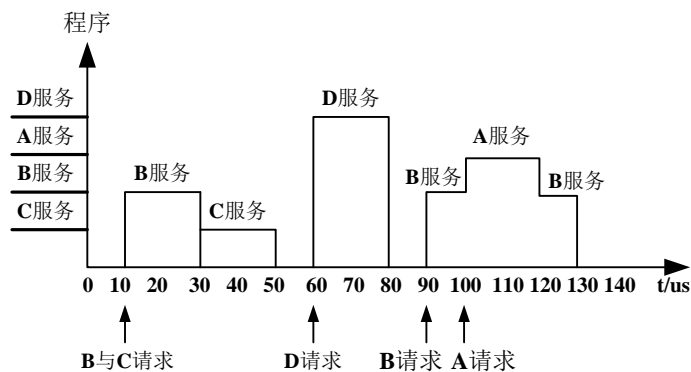
解：INTR——中断请求触发器，用来登记中断源发出的随机性中断请求信号，以便为 CPU 查询中断及中断排队判优线路提供稳定的中断请求信号。

EINT——中断允许触发器，CPU 中的中断总开关。当  $EINT=1$  时，表示允许中断（开中断），当  $EINT=0$  时，表示禁止中断（关中断）。其状态可由开、关中断等指令设置。

INT——中断标记触发器，控制器时序系统中周期状态分配电路的一部分，表示中断周期标记。当  $INT=1$  时，进入中断周期，执行中断隐指令的操作。

24. 现有 A、B、C、D 四个中断源，其优先级由高向低按 A、B、C、D 顺序排列。若中断服务程序的执行时间为  $20\mu s$ ，请根据下图所示时间轴给出的中断源请求中断的时刻，画出 CPU 执行程序的轨迹。

解：A、B、C、D 的响优先级即处理优先级。CPU 执行程序的轨迹图如下：



25. 某机有五个中断源 L0、L1、L2、L3、L4，按中断响应的优先次序由高向低排序为 L0→L1→L2→L3→L4，根据下示格式，现要求中断处理次序改为 L1→L4→L2→L0→L3，根据下面的格式，写出各中断源的屏蔽字。

解：各中断源屏蔽状态见下表：

中断源	屏蔽字				
	0	1	2	3	4
I0	1	0	0	1	0
I1	1	1	1	1	1
I2	1	0	1	1	0
I3	0	0	0	1	0
I4	1	0	1	1	1

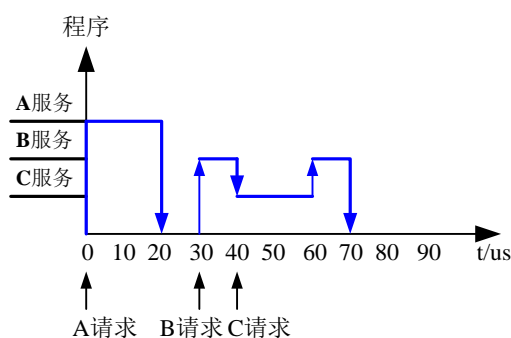
表中：设屏蔽位=1，表示屏蔽；屏蔽位=0，表示中断开放。

26. 设某机配有 A、B、C 三台设备，其优先顺序按 A→B→C 降序排列，为改变中断处理次序，它们的中断屏蔽字设置如下：

设备	屏蔽字
A	111
B	010
C	011

请按下图所示时间轴给出的设备请求中断的时刻，画出 CPU 执行程序轨迹。设 A、B、C 中断服务程序的执行时间均为  $20\mu s$ 。

解：A、B、C 设备的响应优先级为 A 最高、B 次之、C 最低，处理优先级为 A 最高、C 次之、B 最低。CPU 执行程序的轨迹图如下：





## 习题和解析

第一部分《计算机组成原理》中各章习题的解析及补充题的解析。

### 第1章 计算机系统概论

#### 1. 1 习题解析

##### 一、选择题

1. 在下列四句话中, 最能准确反映计算机主要功能的是\_\_\_\_\_。

- A. 计算机可以存储大量信息
- B. 计算机能代替人的脑力劳动
- C. 计算机是一种信息处理机
- D. 计算机可实现高速运算

**解:** 答案为 C。

2. 1946 年 2 月, 在美国诞生了世界上第一台电子数字计算机, 它的名字叫\_\_\_\_\_(1)\_\_\_\_\_, 1949 年研制成功的世界上第一台存储程序式的计算机称为\_\_\_\_\_(2)\_\_\_\_\_。

(1) A. EDVAC    B. EDSAC    C. ENIAC    D. UNIVAC-I

(2) A. EDVAC    B. EDSAC    C. ENIAC    D. UNIVAC-I

**解:** 答案为(1) C, (2) A。

3. 计算机硬件能直接执行的只能是\_\_\_\_\_。

- A. 符号语言
- B. 机器语言
- C. 汇编语言
- D. 机器语言和汇编语言

**解:** 答案为 B。

4. 运算器的核心部件是\_\_\_\_\_。

- A. 数据总线
- B. 数据选择器
- C. 累加寄存器
- D. 算术逻辑运算部件

**解:** 答案为 D。

5. 存储器主要用来\_\_\_\_\_。

- A. 存放程序
- B. 存放数据
- C. 存放微程序
- D. 存放程序和数据

**解:** 答案为 D。

6. 目前我们所说的个人台式商用机属于\_\_\_\_\_。

- A. 巨型机
- B. 中型机
- C. 小型机
- D. 微型机

**解:** 答案为 D。

7. 至今为止, 计算机中所含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:** 答案为 C。

8. 对计算机软、硬件资源进行管理, 是\_\_\_\_\_的功能。

- A. 操作系统
- B. 数据库管理系统
- C. 语言处理程序
- D. 用户程序

**解:** 答案为 A。

9. 企事业单位用计算机计算、管理职工工资, 这属于计算机的\_\_\_\_\_应用领域。

- A. 科学计算
- B. 数据处理
- C. 过程控制
- D. 辅助设计

10. 微型计算机的发展以 技术为标志。

- 解：答案为 B。

1. 操作系统是一种 (1) , 用于 (2) , 是 (3) 的接口。

- 解: 答案为(1) A (2) C (3) C。

- 解:** 答案为运算器、控制器、存储器、输入设备、输出设备。

- 解:** 答案为内（主）存储器，外（辅助）存储器，内存。

- 解: 答案为字节,  $128 \times 2^{20}$  字节。

- 解: 答案为字长, 字长, 长。

- 解：** 答案为程序存储。

- 解：**答案为数据总线、地址总线、控制总线。

- 解：**答案为系统软件、应用软件、工具软件。

- 解：** 答案为模式识别。

- 解：答案为规格。

1. 会计电算化属于计算机应用的哪个领域?

2. 兼容性是计算机的一个重要性能，请解释一下何为向上兼容？

3. 按照冯·诺依曼原理, 现代计算机应具备哪些功能?

(1) 输入输出功能：能把原始数据和解题步骤及中间结果接收下来（输入），把计算结果与计算过程中出现的情况告诉（输出）给用户。

(2) 记忆功能：应能“记住”原始数据、解题步骤及中间结果。

(3) 计算功能：应能进行一些最基本的运算。这些基本运算能组成人们所需要的复杂运算。

(4) 判断功能：计算机在进行一步操作后，应能从预先无法确定的几种方案中选择一种操作方案。

(5) 自我控制功能：计算机应能保证程序执行的正确性和各部件间的协调性。

4. 答：计算机的大部分功能既能由硬件完成，也能由软件完成，从逻辑上讲，两者是等效的。通常用硬件实现执行速度快、成本高、修改困难，而软件刚好相反。两者之间没有固定的界限。

4. 如何理解软、硬件之间的等价性？

答：计算机的大部分功能既能由硬件完成，也能由软件完成，从逻辑上讲，两者是等效的。通常用硬件实现执行速度快、成本高、修改困难，而软件刚好相反。两者之间没有固定的界限。

5. 何谓绿色计算机？对它有哪些要求？

答：绿色计算机是指不会对人类及其生存环境造成不良影响的计算机。对它的主要要求是：

(1) 节能：要求计算机的耗电量要低。

(2) 低污染：在生产、包装过程中尽量使用无毒、可再生材料，打印机噪声要降到最小，电磁辐射要符合环保标准。

(3) 易回收：生产、运输和使用等各环节所用的材料应容易销毁或回收。

(4) 符合人体工程学的要求。

## 第2章 数字电路基础和计算机中的逻辑部件

本章将给出主教材《计算机组成原理》第2章的习题及补充题的解析。

### 2.1 习题解析

#### 一. 判断题

1. 触发器是一种时序电路，它是构成时序逻辑电路的基础。

答：正确。

2. 组合逻辑电路的特点是它的输出状态仅与当时的输入状态有关，而与过去的输入状态无关。

答：正确。

3. 译码器是一种组合逻辑电路，而计数器是一种时序逻辑电路。

答：正确。

4. 移位寄存器除了能把存在其中的数据一位一位地向左或向右移动外，还可用它来判断最左边的位是0还是1。

答：正确。

5. 环形计数器不是用来计数的，而是用于发出顺序控制信号的。

答：正确。

6. J-K 触发器是组成计数器的理想记忆元件。

答：正确。

7. 计数器除了能对输入脉冲进行计数，还能作为分频器用。

答：正确。

8. PLA 中的“与”阵列和“或”阵列都是可编程的。

答：正确。

9. 在 PAL 中，“与”阵列是可编程的，而“或”阵列是固定连接的。

答：正确。

10. PROM 的“与”阵列是不可编程的，“或”阵列是可编程的。

答：正确。

## 二. 选择题

1. 4 输入端“与非”门的某一输入端的输入为“0”，那么它的输出值是\_\_\_\_\_。

- A. “0”
- B. 取决于其他输入端的输入
- C. “1”

解：答案为 C。

2. CPU 中有若干寄存器，其中存放存储器中数据和指令地址的寄存器是\_\_\_\_\_，暂存数据和指令的寄存器是\_\_\_\_\_，存放 CPU 将要执行的下一条指令地址的寄存器是\_\_\_\_\_。

- A. 地址寄存器
- B. 程序计数器
- C. 数据寄存器
- D. 指令寄存器

解：答案依次为 A, C, B。

3. 状态寄存器中的各个状态标志位是依据\_\_\_\_\_来置位的。

- A. 算术逻辑部件上次的运算结果
- B. CPU 将要执行的指令
- C. CPU 已执行的指令
- D. 累加器中的数据

解：答案为 A。

4. 触发器的状态由时钟脉冲的上升沿时 D（数据）端的状态决定，而与触发器的原状态无关，这种触发器是\_\_\_\_\_。

- A. J-K 触发器
- B. R-S 基本触发器
- C. D 型触发器
- D. R-S 同步触发器

解：答案为 C。

5. 在总线中起数据暂存和缓冲作用的器件是\_\_\_\_\_。

- A. 总线传输器
- B. 三态门
- C. 总线控制器
- D. 总线缓冲器

解：答案为 D。

6. 为协调计算机各部件的工作，需要\_\_\_\_\_来提供统一的时钟。

- A. 总线缓冲器
- B. 时钟发生器
- C. 总线控制器
- D. 操作命令发生器

解：答案为 B。

7. 逻辑表达式  $(A+B) \cdot (\bar{A}+B)$  可简化为 ( )

- (1) A
- (2) B
- (3)  $\bar{A}$
- (4)  $\bar{B}$
- (5) 都不是

解：答案为 (2) B。把逻辑表达式展开并用公式进行化简。

8. 正逻辑下的“或非”门是负逻辑下的\_\_\_\_\_。

- A. “与”门
- B. “或”门
- C. “与非”门
- D. “异或”门

解：答案为 C。

## 三. 填空题

1. 可编程逻辑器件的 PLD 实际上是\_\_\_\_\_两级结构的器件。

解：答案为“与”和“或”。

2. 计算机中常用的时序电路有\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_等。

解：答案为触发器、计数器和寄存器。

3. \_\_\_\_\_是构成时序电路的基础。

**解：**答案为触发器。

4. 设“异或”门的二个输入端为 A 和 B，输出为 Y，当 A 接 1 时，输出为\_\_\_\_，当 A 接 0 时，输出为\_\_\_\_。

**解：**答案为 B，B。

5. 计算机中常用的组合逻辑电路有\_\_\_\_\_、\_\_\_\_\_等。

**解：**答案为译码器、数据选择器。

6. 计算机中常用\_\_\_\_\_或\_\_\_\_\_来驱动总线。

**解：**答案集电极开路门、三态门。。

7. 当时钟脉冲 CP 到来时，各触发器的输入端的数据可被锁定至输出端以备输出的寄存器是\_\_\_\_\_。

**解：**答案为并行寄存器。

8. \_\_\_\_\_寄存器常用于乘、除法运算中。

**解：**答案为移位寄存器。因为乘法需要对部分积和乘数进行右移操作，除法要对被除数和余数进行左移操作。

#### 四. 综合题

1. 设 A, B, C 是三个二进制数码，用逻辑表达式写出  $A=B=C$  的判断条件。(提示：先画出真值表)。

**解：**设判断函数为 F，则 F 与 A、B、C 的关系用以下的真值表表示。

ABC	F	ABC	F
000	1	100	0
001	0	101	0
010	0	110	0
011	0	111	1

由此可得到  $F=A \cdot B \cdot C + \overline{A} \cdot \overline{B} \cdot \overline{C}$

2. 请画出下列函数的真值表：

(1)  $F_1(A, B, C) = A \cdot B + \overline{B} \cdot C$

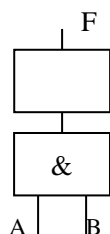
(2)  $F_2(A, B, C) = A + \overline{B} + C$

**解：**函数的真值表如下：

ABC	F1	F2	ABC	F1	F2
000	0	1	100	0	1
001	1	1	101	1	1
010	0	0	110	1	1
011	0	1	111	1	1

3. 试用三种基本门（与、或、非门）组成下列逻辑电路

(1) 异或门 (2) 同或门 (3) 与非门 (4) 或非门



**解：**要先写出这四种电路的逻辑表达式，再用三种基本电路搭接。以“与非”门为例，“与非”门的逻辑表达式是  $F = \overline{A \cdot B}$ ，故电路连接如左图。其他三种情况的做法类似，略。

4. 利用布尔代数的基本定律证明下列等式

(1)  $A+B \cdot C = (A+C) \cdot (A+B)$

(2)  $(A+B+C) \cdot A = 0$

**解：**证明

(1) 等式右边  $= (A+C) \cdot (A+B) = A \cdot A + A \cdot B + C \cdot A + C \cdot B$   
 $= A + A \cdot B + C \cdot A + C \cdot B$   
 $= A \cdot (1+B) + C \cdot A + C \cdot B$   
 $= A \cdot (1+C) + C \cdot B$   
 $= A + C \cdot B$   
 $= \text{等式左边}$

因此,  $A+C \cdot B = (A+C) \cdot (A+B)$

(2) 用摩尔定理, 等式左边展开为  $A \cdot B \cdot C \cdot A = A \cdot A \cdot B \cdot C = 0 = \text{右边}$

### 第3章 信息表示与运算基础

#### 习题解析

##### 一、判断题

1. 在数字计算机中所以采用二进制是因为二进制的运算最简单。

**答：**正确。

2. 在所有的进位计数制中, 整数部分最低位的权都是 1。

**答：**正确。

3. 某 R 进位计数制, 其左边一位的权是其相邻的右边一位的权的 R 倍。

**答：**正确。

4. 计算机表示的数发生溢出的根本原因是计算机的字长有限。

**答：**错误。

5. 表示定点数时, 若要求数值 0 在计算机中唯一地表示为全 0, 应采用补码。

**答：**正确。

6. 浮点数的取值范围由阶码的位数决定, 而精度由尾数的位数决定。

**答：**正确。

7. CRC 校验码的生成和检验大多采用软件实现。

**答：**正确。

8. 若浮点数的尾数用补码表示, 那么规格化的浮点数是指尾数数值位的最高位是 0 (正数) 或是 1 (负数)。

**答：**正确。

9. 在实际应用中, 奇偶校验多采用奇校验, 这是因为奇校验中不存在全“0”代码, 在某些场合下更便于判别。

**答：**正确。

10. 显示图形时要经过复杂的数学计算, 因此占用的时间要比位图图像的时间长。

**答：**正确。

##### 二、选择题

1. 下列各种数制的数中最小的数是\_\_\_\_\_。

A.  $(101001)_2$     B.  $(101001)_{BCD}$     C.  $(52)_8$     D.  $(233)_H$

**解：**答案为 B。

2. 下列各种数制的数中最大的数是\_\_\_\_\_。





10. 在大量数据传送中常用的且有效的检验法是\_\_\_\_\_。

A. CRC 码 B. 海明码 C. 偶校验码 D. 奇校验码

**解:** 答案为 A。

### 三、填空题

1. 计数制中使用的数据个数被称为\_\_\_\_\_。

**答:** 基。

2. 在用\_\_\_\_\_表示的机器数中, 零的编码是唯一的。

**答:** 补码。

3. 信息的数字化编码是指\_\_\_\_\_。

**答:** 是指用“0”或“1”的二进制编码, 并选用一定的组合规则来表示信息。

4. 一个定点数由\_\_\_\_\_和\_\_\_\_\_两部分组成。根据小数点位置不同, 定点数据有\_\_\_\_\_和\_\_\_\_\_两种表示方法。

**答:** 符号位, 数值域, 纯小数, 纯整数 (顺序可变)。

5. BCD 码中, 每一位十进制数字由\_\_\_\_\_位二进制数码组成, 用 ASCII 码表示一个字符通常需要\_\_\_\_\_位二进制数码。

**答:** 4, 7。

6. 移码常用来表示浮点数的\_\_\_\_\_部分, 移码和补码比较, 它们除\_\_\_\_\_外, 其他各位都\_\_\_\_\_。

**答:** 阶码, 符号位, 相同。

7. 码距的定义是\_\_\_\_\_。

**答:** 编码系统中任两个合法码之间的最少二进制位数的差异。

8. 8421 码用二进制求和时, 当和超过\_\_\_\_\_时, 需要做\_\_\_\_\_修正。

**答:** 9, 加 6 调整。

9. 有二进制数  $D_4D_3D_2D_1$ , 奇偶校验值用  $p$  表示, 则奇校验为\_\_\_\_\_, 偶校验为\_\_\_\_\_, 奇偶校验只能检测\_\_\_\_\_, 无法检测\_\_\_\_\_。

**答:**  $P=D_4 \oplus D_3 \oplus D_2 \oplus D_1$ ,  $\bar{P}=D_4 \oplus D_3 \oplus D_2 \oplus D_1$ , 奇数个错, 偶数个错。

10. 在浮点加减法运算中, 当运算结果的尾数的绝对值大于 1 时, 需要对结果进行\_\_\_\_\_, 其操作是\_\_\_\_\_。

**答:** 向右规格化, 尾数右移一位, 右边补一个 0, 阶码减 1, 直到尾数绝对值  $\geq 0.5$ 。

### 四、计算题

1. 用二进制数表示一个四位十进制的整数最少需要几位 (不含符号位)。

**解:**  $2^x=10^4$ ,  $N=4 \times 1/\log 2=14$  位。

2. 某机器字长 32 位, 定点表示, 其中 31 位表示尾数, 1 位是符号位, 问:

(1) 定点原码整数表示时, 最大正数是多少? 最小负数是多少?

(2) 定点原码小数表示时, 最大正数是多少? 最小负数是多少?。

**解:** (1) 定点原码整数表示时, 最大正数  $= (2^{31}-1)$ ; 最小负数  $= -(2^{31}-1)$

(2) 定点原码小数表示时, 最大正数  $= (1-2^{-31})$ ; 最小负数  $= -(1-2^{-31})$

3. 写出下列二进制数的原码、反码、补码和移码。

(1)  $\pm 1011$  (2)  $\pm 0.1101$  (3)  $\pm 0$

**解:** ①  $(+1011)_{原}=01011$   $(-1011)_{原}=11011$

$(+1011)_{反}=01011$   $(-1011)_{反}=10100$

$(+1011)_{补}=01011$   $(-1011)_{补}=10101$

$(+1011)_{\text{移}} = 11011$	$(-1011)_{\text{移}} = 00100$
② $(+0.1101)_{\text{原}} = 0.1101$	$(-0.1101)_{\text{原}} = 1.1101$
$(+0.1101)_{\text{反}} = 0.1101$	$(-0.1101)_{\text{反}} = 1.0010$
$(+0.1101)_{\text{补}} = 0.1101$	$(-0.1101)_{\text{补}} = 1.0011$
$(+0.1101)_{\text{移}} = 0.1101$	$(-0.1101)_{\text{移}} = 0.0011$
③ $(+0.0000)_{\text{原}} = 00000$	$(-00000)_{\text{原}} = 10000$
$(+0.0000)_{\text{反}} = 00000$	$(-00000)_{\text{反}} = 11111$
$(+0.0000)_{\text{补}} = 00000$	$(-00000)_{\text{补}} = 00000$
$(+0.0000)_{\text{移}} = 10000$	$(-00000)_{\text{移}} = 10000$

4. 某机器字长 16 位, 浮点表示时, 其中含 1 位阶符、5 位阶码、1 位尾符、9 位尾数, 请写出它能表示的最大浮点数和最小浮点数。

**解:** 最大浮点数  $= 2^{+21} \times (1 - 2^{-9})$

最小浮点数  $= -2^{+31} \times (1 - 2^{-9})$ 。

5. 字符“F”的 ASCII 码为 46H, 请写出它的奇校验码和偶校验码 (假定校验位加在最高位)。

**解:** 字符“F”的 ASCII 码为 46H, 奇校验码为 10110110 (B6H), 偶校验码为 00110110 (36H)。

6. 已知被校验的数据为 101101, 求其海明校验码。

提示: 先决定校验位的位数  $r=4$ , 然后根据编码规则决定海明校验位的位置和数据位的位置, 最后用偶校验法求出校验位的值。答案应为 1011100100。

**解:** (1) 先定校验位的位数, 当  $r=4$  时, 共有 16 种状态, 大于  $k+r+1=11$ , 故选用  $r=4$ , 校验位至少取 4 位。

(2) 决定校验位的位置: 按海明码生成法规定, 海明校验位第  $i$  位应放在  $2^{i-1}$  的海明位置上。

(3) 决定数据位的位置: 数据位应由低到高依次插空放在其他海明位上。即  $D_6 D_5 P_4 D_4 D_3 P_3 D_1 P_2 P_1$ 。

(4) 决定被校验数据位由哪几位校验位进行校验: 按海明码生成法规定, 每个数据位由多个校验位进行校验, 但被校验数据的海明位号要等于校验该位数据的各位校验位的海明位号之和。

(5) 决定各个校验位之值: 按海明码生成法规定, 用偶校验法生成校验位, 校验位之值为各被校验位数据之和。

$$P_1 = D_1 \oplus D_2 \oplus D_4 \oplus D_5$$

$$P_2 = D_1 \oplus D_3 \oplus D_4 \oplus D_6$$

$$P_3 = D_2 \oplus D_3 \oplus D_4$$

$$P_4 = D_5 \oplus D_6$$

在本例中, 被校数据为 101101, 即  $D_1=1, D_2=0, D_3=1, D_4=1, D_5=0, D_6=1$ , 故

$$P_1 = D_1 \oplus D_2 \oplus D_4 \oplus D_5 = 0$$

$$P_2 = D_1 \oplus D_3 \oplus D_4 \oplus D_6 = 0$$

$$P_3 = D_2 \oplus D_3 \oplus D_4 = 1$$

$$P_4 = D_5 \oplus D_6 = 1$$

最后得到被校验数据 101101 的海明码为 1011100100。

7. 已知被检信息为 1010, 选择的生成多项式是  $G(X)$  为  $X^3+X+1$ , 求 CRC 校验码, 并求循环余数, 说明其校验原理。

**解:** (1) 生成多项式为  $K+1$  位的  $X^3+X+1$ , 即  $G(X) = 1011$ 。

(2) 确定校验位的位数为  $K=3$ 。

(3) 在有效数据位后面添 3 ( $K=3$ ) 个 0, 然后用它与  $G(X)$  进行模 2 除法运算, 所得余数即为所求的校验位。

运算过程如下:

$$\begin{array}{r} 1001 \\ 1011 \overline{) 1010100} \\ \underline{1011} \phantom{00} \\ 1000 \phantom{00} \\ \underline{1001} \phantom{00} \\ 011 \end{array}$$

余数为 011, 所以被检数据的 CRC 校验码为 1010011。

(4) 求循环余数: 在上述余数 011 的基础上添 0 继续进行模 2 除法。余数循环如下:  
 $011 \rightarrow 110 \rightarrow 111 \rightarrow 101 \rightarrow 001 \rightarrow 010 \rightarrow 100 \rightarrow 011$ 。除法过程略。

8. 将二进制数 1011010 转换成 8421 码。

**解:** 先把二进制数转换成十进制数,  $(1011011)_2 = 91 = (10010001)_{8421}$ 。

## 五、简答题

1. 试比较定点带符号数在计算机内的四种表示方法。

**答:** 带符号数在计算机内部的表示方法有原码、反码、补码和移码。

原码表示方法简单易懂, 实现乘、除运算简单, 但用它实现加、减运算比较复杂。

补码的特点是加、减法运算规则简单, 正负数的处理方法一致。

反码通常只用来计算补码, 由于用反码运算不方便, 在计算机中没得到实际应用。

移码由于保持了数据原有的大小顺序, 便于进行比较操作, 常用于浮点数中的阶码, 使用比较方便。

2. 试述浮点数规格化的目的和方法。

**答:** 浮点的规格化是为了使浮点数尾数的最高数值位为有效数位。当尾数用补码表示时, 若符号位与小数点后的第一位不相等, 则被定义为已规格化的数, 否则便是非规格化数。通过规格化, 可以保证运算数据的精度。

通常, 采用向左规格化 (简称左规), 即尾数每左移一位, 阶码减 1, 直至规格化完成。

3. 在检错码中, 奇偶校验法能否定位发生错误的信息位? 是否具有纠错功能?

**答:** (1) 在检错码中, 奇偶校验法不能定位发生错误的信息位。

(2) 奇偶校验没有纠错能力。4. 简述循环冗余码 (CRC) 的纠错原理。

4. 简述循环冗余码 (CRC) 的纠错原理。

**答:** CRC 码是一种纠错能力较强的校验码。在进行校验时, 先将被检数据码的多项式用生成多项式  $G(X)$  来除, 若余数为 0, 说明数据正确; 若余数不为 0, 则说明被检数据有错。只要正确选择多项式  $G(X)$ , 余数与 CRC 码出错位位置的对应关系是一定的, 由此可以用余数作为判断出错位置的依据而纠正出错的数据位。

## 第 4 章 存储系统

### 4.1 习题解析

#### 一. 判断题

1. 计算机的主存是由 RAM 和 ROM 两种半导体存储器组成的。

**答:** 正确。

2. CPU 可以直接访问主存, 而不能直接访问辅存。

答：正确。

3. 外（辅）存比主存的存储容量大、存取速度快。

答：错误。

4. 动态 RAM 和静态 RAM 都是易失性半导体存储器。

答：正确。

5. Cache 的功能全部由硬件实现。

答：正确。

6. 引入虚拟存储器的目的是为了加快辅存的存取速度。

答：错误。

7. 多体交叉存储器主要是为了解决扩充容量的问题。

答：错误。多体交叉存储器主要是为了提高存取速度，增加带宽。

8. Cache 和虚拟存储器的存储管理策略都利用了程序的局部性原理。

答：正确。

9. 多级存储体系由 Cache、主存和辅存构成。

答：正确。

10. 在虚拟存储器中，当程序正在执行时，由编译器完成地址映射。

答：错误（由操作系统完成）。

## 二. 选择题

1. 主（内）存用来存放\_\_\_\_\_。

A. 程序    B. 数据    C. 微程序    D. 程序和数据

解：答案为 D。

2. 下列存储器中，速度最慢的是\_\_\_\_\_。

A. 半导体存储器    B. 光盘存储器    C. 磁带存储器    D. 硬盘存储器

解：答案为 C。

3. 某一 SRAM 芯片，容量为  $16K \times 1$  位，则其地址线有\_\_\_\_\_。

A. 14 根    B.  $16K$  根    C. 16 根    D. 32 根

解：答案为 A。

4. 下列部件（设备）中，存取速度最快的是\_\_\_\_\_。

A. 光盘存储器    B. CPU 的寄存器    C. 软盘存储器    D. 硬盘存储器

解：答案为 B。

5. 在主存和 CPU 之间增加 Cache 的目的是\_\_\_\_\_。

A. 扩大主存的容量  
B. 增加 CPU 中通用寄存器的数量  
C. 解决 CPU 和主存之间的速度匹配  
D. 代替 CPU 中的寄存器工作

解：答案为 C。

6. 计算机的存储器采用分级存储体系的目的是\_\_\_\_\_。

A. 便于读写数据    B. 减小机箱的体积  
C. 便于系统升级    D. 解决存储容量、价格与存取速度间的矛盾

解：答案为 D。

7. 相联存储器是按\_\_\_\_\_进行寻址的存储器。

A. 地址指定方式    B. 堆栈存取方式  
C. 内容指定方式    D. 地址指定与堆栈存取方式结合

解：答案为 C。

8. 某 SRAM 芯片, 其容量为  $1K \times 8$  位, 加上电源端和接地端后, 该芯片的引出线的最少数目应为\_\_\_\_\_。

- A. 23      B. 25      C. 50      D. 20

**解:** 答案为 D。

9. 常用的虚拟存储器由\_\_\_\_\_两级存储器组成, 其中辅存是大容量的磁表面存储器。

- A. 主存—辅存      B. 快存—主存  
C. 快存—辅存      D. 通用寄存器—主存

**解:** 答案为 A。

10. 在 Cache 的地址映射中, 若主存中的任意一块均可映射到 Cache 内的任意一块的位置上, 则这种方法称为\_\_\_\_\_。

- A. 全相联映射      B. 直接映射      C. 组相联映射      D. 混合映射

**解:** 答案为 A。

### 三. 填空题

1. 对存储器的要求是\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_，为解决这三者的矛盾, 计算机、采用\_\_\_\_\_体系结构。

**答:** 速度快、容量大、成本低, 分级存储体系。

2. CPU 能直接访问\_\_\_\_\_和\_\_\_\_\_，但不能访问\_\_\_\_\_和\_\_\_\_\_。

**答:** 主存、CACHE、外存、I/O 设备。

3. Cache 的映射方式有\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_三种。其中\_\_\_\_\_方式, 适度地兼顾了前两者的优点又尽量避免其缺点, 比较理想。

**答:** 直接映象、全相联映象、组相联映象, 组相联映象。

4. 广泛使用的\_\_\_\_\_和\_\_\_\_\_都是半导体\_\_\_\_\_存储器。前者的速度比后者快, \_\_\_\_\_不如后者高。它们的共同缺点是断电后\_\_\_\_\_保存信息。

**答:** 依次为 SRAM、DRAM、随机读写、集成度、不能。

5. 闪速存储器能提供高性能、低功耗、高可靠性以及\_\_\_\_\_能力, 为现有的\_\_\_\_\_体系结构带来巨大变化, 因此作为\_\_\_\_\_用于便携式电脑中。

**答:** 瞬时启动、存储器、固态硬盘。

6. 相联存储器不按地址而是按\_\_\_\_\_访问的存储器, 在 Cache 中用来存放\_\_\_\_\_, 在虚拟存储器中用来存放\_\_\_\_\_。

**答:** 内容, 行地址表, 段表、页表和快表。

7. 虚拟存储器指的是\_\_\_\_\_层次, 它给用户提供了一个比实际\_\_\_\_\_空间大得多的\_\_\_\_\_。

**答:** 主-辅存, 主存, 虚拟地址。

8. 磁盘的地址格式由\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_四部分组成。

**答:** 台号、柱面号(磁道号)、盘面号(磁头号)、扇区号。

9. 温彻斯特硬盘的特点是\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_。

**答:** 固定盘片、活动磁头和密封结构。

10. 一个完整的磁盘存储器由三部分组成, 其中\_\_\_\_\_又称磁盘机或磁盘子系统, 是独立于主机的一个完整的设备, \_\_\_\_\_是磁盘机与主机的接口部件, \_\_\_\_\_用于保存信息。

**答:** 驱动器、控制器、盘片。

### 四. 计算题

1. 设有一个具有 24 位地址和 8 位字长的存储器，求：

- (1) 该存储器能存储多少字节的信息？
- (2) 若存储器由  $4\text{M} \times 1$  位的 RAM 芯片组成，需要多少片？
- (3) 需要哪种译码器实现芯片选择？

**解：**(1) 存储单元数为  $2^{24} = 16\text{M} = 16777216$ ，故能存储 16M 字节的信息。

(2) 由于存储容量为 16MB (8 位字长)，每 4M 字节需要 4 片 (位并联方式)，故需芯片数为  $16/4 \times 8 = 32$  片。

(3) 若用 32 片组成一个 16M (8 位字长)，地址总线的低 22 位可直接连到芯片的 A0-A21 管脚，而地址总线的高 2 位 (A22, A23) 需要通过 2:4 线译码器进行芯片选择。存储器组成方案为位并联和地址串联相结合的方式。

2. 某计算机系统的内存由 Cache 和主存构成，Cache 的存取周期为 45ns，主存的存取周期为 200ns。已知在一段给定的时间内，CPU 共访问内存 4500 次，其中 340 次访问主存，求：

- (1) Cache 的命中率是多少？
- (2) CPU 访问内存的平均访问时间是多少？
- (3) Cache-主存系统的效率是多少？

**解：**(1) 命中率  $H = (4500 - 340) / 4500 = 0.92$ 。

(2) CPU 访存的平均时间  $T = 0.92 \times 45 + (1 - 0.92) \times 200 = 57.4\text{ns}$

(3) cache-主存系统的效率  $e = 45/57.4 = 78\%$

3. 下图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
3	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：**(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，求得主存实地址码为 80324。

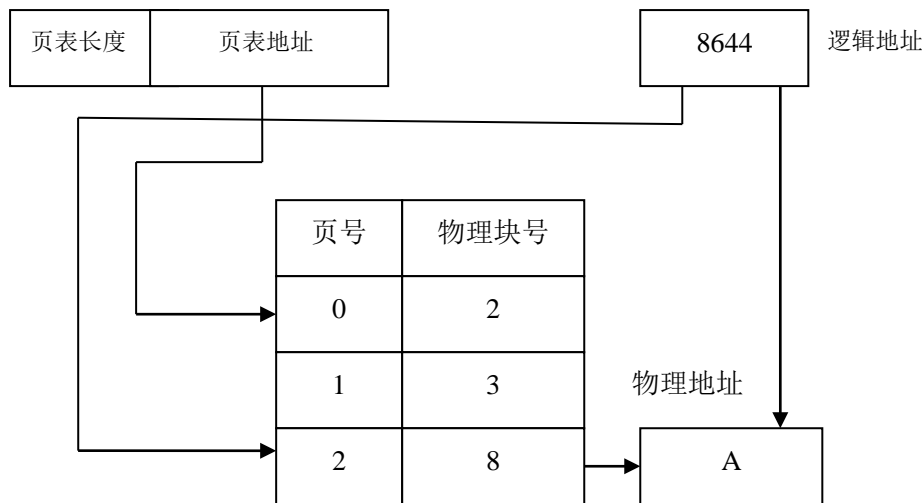
(2) 同理，主存实地址码  $= 96000 + 0128 = 96128$ 。

(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。

4. 页式存储器的逻辑地址由页号和页内地址两部分组成，若页面大小为 4KB，地址转换过程如图所示，图中逻辑地址 8644 用十进制表示，经页表转换后，该逻辑地址的物理地

址（十进制）是多少？

提示：把逻辑地址转换成二进制来做。



**解：**第一步，已知页面大小为 4KB，故可算出页内地址为 12 位 ( $2^{12}=4096$ )。

第二步：把逻辑地址 8644 转换成二进制地址 10000111000100，其中高 2 位为页面号。

第三步：查页表，2 号页面的物理块号为 8，由于逻辑地址和物理地址的页内地址部分是相同的，即可把页号与页内地址拼接，得到物理地址为 100000011100100。

第四步：把 100000011100100 转换成十进制数为 33220。

5. 设某磁盘存储器的平均找道时间为  $t_s$ ，转速每分钟  $r$  转，每磁道容量为  $N$  个字，每信息块为  $n$  个字，试推导磁盘存储器读写一块信息所需总时间  $t_B$  的公式。

**解：**  $t_B = t_s + 60/2r + 60n/rN$ 。

6. 一般，CD-ROM 光盘的外缘有 5mm 宽的范围不使用，故标准的播放时间为 60 分钟。请计算模式 1 和模式 2 情况下光盘的存储容量是多少？

提示：光盘的模式 1 用于存放计算机的程序和数据，共有 270000 个扇区，每个扇区 2048 个字节，而模式 2 用于存放声音、图像等多媒体数据，共有 270000 个扇区，每个扇区 2336 个字节。

**解：**模式 1——容量  $C = 270000 \times 2048 / 1024 / 1024 = 527\text{MB}$

模式二——容量  $C = 270000 \times 2336 / 1024 / 1024 = 601\text{MB}$

7. 某磁盘组有 4 个盘片，5 个记录面，每个记录面的内磁道直径为 22cm，外磁道直接为 33cm，最大位密度为 1600b/cm，道密度为 80 道/cm，转速 7200 转速/分。

(1) 磁盘的总存储容量（非格式化）？

(2) 最大数据传输率是多少？

**解：**(1) 总容量 = 每面容量  $\times$  记录面数

每面容量 = 某一磁道容量  $\times$  磁道数

某磁道容量 = 磁道长  $\times$  本道位密度

所以，最内圈磁道的容量 =  $1600 \times 22 \times 3.14 = 110528$  字节/道

磁道数 = 存储器域长  $\times$  道密度 =  $(33 - 22) / 2 \times 80 = 253616000$  字节

(2) 最大数据传输率 = 转速  $\times$  某磁道容量 =  $7200 / 60 \times 110528 = 13263360$  字节/秒

8. 假设某计算机的 cache 采用直接映象，容量为 4096。若 CPU 依次从主存单元 0, 1, ..., 99 和 4096, 4097, ..., 4195 交替取指令，循环执行 10 次，试求此时的命中率？



**解：**此时 CPU 的命中率为 0。

## 第 5 章 中央处理器

### 习题解析

#### 一. 判断题

1. 一个指令周期由若干个机器周期组成。

**解：**答案为正确。

2. 非访内指令不需从内存中取操作数，也不需将目的操作数存放到内存，因此这类指令的执行不需地址寄存器参与。

**解：**答案为错误。

3. 组合逻辑控制器比微程序控制器的速度快。

**解：**答案为正确。

4. 流水线中的相关问题是指在一段程序的相邻指令之间存在某种信赖关系，这种关系影响指令的执行。

**解：**答案为正确。

5. 微程序控制控制方式与硬布线控制方式相比，最大的优点是提高了指令的执行速度。

**解：**答案为正确。

6. 微程序控制器中的控制存储器可用 PROM、EPROM 或闪存实现。

**解：**答案为正确。

7. 指令周期是指从 CPU 从主存取出一条指令开始到执行这条指令完成所需的时间。

**解：**答案为正确。

8. 控制存储器是用来存放微程序的存储器，它比主存储器速度快。

**解：**答案为正确。

9. 机器的主频最快，机器的速度就最快。

**解：**答案为正确。

10. 80X86 的数据传送指令 MOV，不能实现两个内存操作数的传送。

**解：**答案为正确。

#### 二. 选择题

1. 指令系统中采用不同寻址方式的目的主要是\_\_\_\_\_。

- A. 实现程序控制和快速查找存储器地址
- B. 可以直接访问主存和外存
- C. 缩短指令长度，扩大寻址空间，提高编程灵活性
- D. 降低指令译码难度

**解：**答案为 C。

2. CPU 组成中不包括\_\_\_\_\_。

- |          |          |
|----------|----------|
| A. 指令寄存器 | B. 地址寄存器 |
| C. 指令译码器 | D. 地址译码器 |

**解：**答案为 D。

3. 程序计数器 PC 在\_\_\_\_\_ 中。

- |        |           |
|--------|-----------|
| A. 运算器 | B. 控制器    |
| C. 存储器 | D. I/O 接口 |

**解：**答案为 B。

4. 计算机主频的周期是指\_\_\_\_\_。

- |         |         |
|---------|---------|
| A. 指令周期 | B. 时钟周期 |
|---------|---------|

- 解：答案为 B。

- 解：答案为 B。

- 解：答案为 A。

- 解：答案为 C。

- 解：答案为 C。

- 解：答案为 D。

- 解：答案为 C。

- 解：答案为 C。

- 解：答案为 C。

- 可供选择的答案:

- ① A. 顺序 B. 重叠 C. 循环 D. 并行  
② A. 为 0 B. 为 1 个周期 C. 相等 D. 不等  
③ A. 3 个阶段执行时间之和 B. 3 个阶段执行时间的平均值  
C. 3 个阶段执行时间的最小值 D. 3 个阶段执行时间的最大值

14. 微指令格式分成水平型和垂直型, 前者的位数\_\_\_\_\_, 用它编写的微程序\_\_\_\_\_。

A. 较少                                      B. 较多

C. 较长                                      D. 较短

15. 异步控制常作为\_\_\_\_\_的主要控制方式。

- A. 单总线计算机结构计算机中访问主存和外部设备时
- B. 微型机的 CPU 控制中
- C. 组合逻辑的 CPU 控制中
- D. 微程序控制器中

16. 与微指令的执行周期对应的是\_\_\_\_\_。

A. 指令周期                                      B. 机器周期

C. 节拍周期                                     D. 时钟周期

12. 在同一微周期中\_\_\_\_\_的微命令叫互斥的微命令; 在同一微周期中\_\_\_\_\_的微命令叫相

容的微命令。显然，\_\_\_\_\_不能放在一起译码。

**答：**不可能同时出现，可以同时出现，相容的微命令。

13. 在微程序控制器中，时序信号比较简单，一般采用\_\_\_\_\_。

**答：**同步控制。

14. 保存当前栈顶地址的寄存器叫\_\_\_\_\_。

**答：**栈顶指针 SP。

15. 实现下面各功能有用哪些寄存器？

(1)表示运算结果是零的是\_\_\_\_\_。

(2)表示运算结果溢出的是\_\_\_\_\_。

(3)表示循环计数的是\_\_\_\_\_。

(4)做 8 位乘除法时用来保存被乘数和被除数的是\_\_\_\_\_。

(5)暂时存放参加 ALU 中运算的操作数和结果的是\_\_\_\_\_。

**答：**(1)状态寄存器中的 ZF。

(2)状态寄存器中的 OF。

(3)CX。

(4)AL/AX。

(5)累加器。

#### 四. 综合题

1. 在 8086 中，对于物理地址 2014CH 来说，如果段起始地址为 20000H，则偏移量应为多少？

**解：**14CH。

2. 在 8086 中 SP 的初值为 2000H，AX=3000H，BX=5000H。试问：

(1) 执行指令 PUSH AX 后，SP=?

(2) 再执行指令 PUSH BX 及 POP AX 后，SP=?, BX=? 请画出堆栈变化示意图。

**解：**(1)  $SP=2000H-2=1FFE H$

(2) 执行 PUSH BX 和 POP AX 后， $SP=1FFE H-2+2=1FFE H$ ，BX=5000H，指针变化图略。

3. 指出下列 8086 指令中，源操作数和目的操作的寻址方式。

(1) PUSH AX (2) XCHG BX, [BP+SI]

(3) MOV CX, 03F5H (4) LDS SI, [BX]

(5) LEA BX, [BX+SI] (6) MOV AX, [BX+SI+0123H]

(7) MOV CX, ES: [BX][SI] (8) MOV [SI], AX

(9) XCHG AX, [2000H]

**解：**(1)源是寄存器直接寻址。目的是寄存器间接寻址。

(2)源是变址/基址寻址，目的是寄存器直接寻址。

(3)源是立即数寻址，目的是寄存器直接寻址。

(4)源是寄存器间接寻址。目的是寄存器直接寻址。

(5)源是变址/基址寻址，目的是寄存器直接寻址。

(6)源是变址/基址加偏移量寻址，目的是寄存器直接寻址。

(7)源是跨段的变址/基址寻址，目的是寄存器直接寻址。

(8)源是寄存器直接寻址。目的是寄存器间接寻址。

(9)源是存储器直接寻址。目的是寄存器直接寻址。

4. 请按下面的要求写出相应的 8086 汇编指令序列。

(1) 将 1234H 送入 DS 中

(2) 将 5678H 与 AX 中的数相加，结果放在 AX 中。

- (3) 将 DATAX 和 DATAY 相加，其和放在 DATAY 中。
- (4) 将 AX 中的高 4 位变为全 0。
- (5) 将 AX 中的低 2 位变为全 1。

**解：**(1)MOV AX, 1234H ; MOV DS, AX

(2)ADD AX, 5678H

(3)MOV AX, DATAX

ADD DATAY, AX

(4)AND AX, 0FFFFH

(5)OR AX, 0003H

5. 若 BX=0379H，下面 8086 指令执行后，BX=?

(1) XOR BX, 0FF00H

(2) AND BX, 0FF00H

(3) OR BX, 0FF00H

**解：**(1)根据“异或”操作的特点，与“0”做“异或”操作，操作数不变；与“1”做“异或”操作，即使操作数变反，此题仅使 BH 变反，所以结果为 BX=1111110001111001=FC79H。

(2)根据“与”操作的特点，对“0”做“与”操作，操作数为 0；对“1”做“与”操作，即操作数不变。此题仅使 BL 清 0，BH 不变，所以结果为 BX=0000001100000000=0300H。

(3)根据“或”操作的特点，对“0”做“或”操作，操作数不变；对“1”做“或”操作，即操作数为 1。此题仅使 BH(即高 8 位)置 1，BL(低 8 位)不变，所以结果为 BX=0000001110000110=0386H。

6. 若 (BX)=5555H，试写出执行完下面的指令序列后 BX 中的内容。

MOV CL, 5

SHR BX, CL

**解：**第一条指令把数 5 传送到 CL (CL 是指定用于存放移位或循环次数的寄存器) 中，第 2 条是逻辑右移指令，使 BX 逻辑右移 5 次 (由 CL 给出移位次数)，结果 BX=02AAH。

7. 试用 8086 移位和加法指令完成将 AX 中的内容乘以 10 的操作。

**解：**算法为  $AX \times 10 = AX \times 2 + AX \times 8$ ，指令序列如下：

SHL AX, 1 ; AX×2

MOV BX, AX ; AX×2 保存到 BX

SHL AX, 1 ; AX×4

SHL AX, 1 ; AX×8

ADD AX, BX ; AX×10

8. 根据以下要求写出相应的 8086 汇编语言指令。

(1) 把 BX 寄存器和 DX 寄存器的内容相加，结果存入 DX 寄存器中。

(2) 用寄存器 BX 和 SI 的基址变址寻址方式把存储器中的一个字节与 AL 寄存器的内容相加，并把结果送到 AL 寄存器中。

(3) 用寄存器 BX 和偏移量 0BD2H 的寄存器相对寻址方式把存储器中的一个字和 CX 相加并把结果送回存储器中。

(4) 用偏移量为 0524H 的直接寻址方式把存储器中的一个字与数 2A59H 相加，并把结果送回该存储单元中。

(5) 把数 0B5H 与 AL 相加，并把结果送回 AL 中。

**解：**(1)ADD DX, BX

(2)ADD AL, [BX+SI]

(3)ADD [BX+0BD2H], CX

(4)ADD [0524H], 2A59H

(5)ADD AL, 0B5H

9. 已知 8086 汇编程序段如下:

MOV AX, 1234H

MOV CL, 4

ROL AX, CL

DEC AX

MOV CX, 4

MUL CX

INT 20H

试问:

(1) 每条指令执行完后, AX 寄存器的内容是什么?

(2) 每条指令执行完后, 进位、溢出和零标志的值是什么?

(3) 程序结束时, AX 和 DX 的内容是什么?

**解:** (1) MOV AX, 1234H      AX=1234H

MOV CL, 4      AX=1234H

ROL AX, CL      AX=2341H

DEC AX      AX=2340H

MOV CX, 4      AX=2340H

MUL CX      AX=8D00H

INT 20H

第 2、5、7 条指令对 AX 没有操作, 故不影响 AX 的值。第 1 条指令把立即数 1234H 送到 AX 中, AX 的值就是 1234H, 第 3 条指令把 AX 的内容循环移位 4 次, AX 值为 2341H, 第 4 条指令把 AX 减 1, AX 的值为 2340H, 第 6 条指令把 AX 的值乘以 4, AX 的值变为 8D00H。

(2) 传送指令不影响标志位, 第 3 条指令影响进位 (CF)、溢出 (OF) 标志, CF=0, OF=0; 第 4 条指令不影响 CF 标志, 该指令执行后, CF 维持原状, OF=0, 零标志 (ZF) 也为 0; 第 6 条指令影响各标志位, 由于 DX=0, 因此 CF=0, OF=0, ZF=0。

(3) 程序结束时, AX 的值为 8D00H, DX 的值为 0000H。

10. 有一主频为 25 MHz 的微处理器, 平均每条指令的执行时间为两个机器周期, 每个机器周期由两个时钟脉冲组成。

(1) 假定存储器为“0 等待”, 请计算机器速度 (每秒钟执行的机器指令条数)。

(2) 假如存储器速度较慢, 每两个机器周期中有一个是访问存储器周期, 需插入两个时钟的等待时间, 请计算机器速度。

**解:** (1) 存储器“0 等待”是假设在访问存储器时, 存储周期=机器周期, 此时  
机器周期=主频周期×2 (一个机器周期由两个时钟脉冲组成)

$$=2/25\text{MHz}=0.08\mu\text{S}$$

$$\text{指令周期}=2\times\text{机器周期}=0.16\mu\text{S}$$

$$\text{机器平均速度}=1/0.16=6.25\text{MIPS (百万条指令/秒)}$$

(2) 若每两个机器周期中有一个是访问存储器周期, 则需插入两个时钟的等待时间。

$$\text{指令周期}=0.16\mu\text{S}+0.08\mu\text{S}=0.24\mu\text{S}$$

$$\text{机器平均速度}=1/0.24\approx4.2\text{MIPS (百万条指令/秒)}$$

## 习题解析

### 一. 判断题

1. 随着半导体集成电路的发展, 外部设备在计算机系统硬件的价格中所占的比重将越来越低。

**解:** 答案为正确。

2. 灰度级是指黑白显示器中所显示像素点的亮暗差别, 在彩色显示器中则表现为颜色数的不同, 灰度级越高, 图像层次越逼真清晰。

**解:** 答案为正确。

3. 在计算机处理的图形和图像两种对象中, 一般来讲图像的数据量比图形大。

**解:** 答案为正确。

4. 顾名思义, 外部设备位于主机箱的外部。

**解:** 答案为错误。

5. 图形比图像更适合表现类似于照片和绘画之类有真实感的画面。

**解:** 答案为错误。

6. 从控制光标原理上讲, 三种鼠标器基本相同, 都是把鼠标器的移动距离和方向变为脉冲信号送给计算机, 计算机再把脉冲信号转换成显示器光标的坐标位置, 从而达到指示位置的目的。

**解:** 答案为正确。

7. 感光鼓是激光打印机中电子照相系统的核心, 鼓面上涂有一层具有光敏特性的半导体感光材料, 通常用硒, 故有硒鼓之称。

**解:** 答案为正确。

8. 点阵针式打印机点阵的点越多, 打印质量越高。

**解:** 答案为正确。

9. 非击打式打印机速度快、噪声低、打印质量高, 但价格较贵。

**解:** 答案为正确。

10. LCD 显示器没有背景光源也能工作。

**解:** 答案为错误。

### 二、选择题

1. 使主机从外部获取信息的设备称为\_\_\_\_\_。

A. 外部存储器 B. 外部设备 C. 输入设备 D. 输出设备

**解:** 答案为 C

2. 在显示器的技术指标中, 数据  $640 \times 480$ ,  $1024 \times 768$  等表示\_\_\_\_\_。

A. 显示器屏幕的大小 B. 显示器显示字符的最大行数和列数  
C. 显示器的颜色指标 D. 显示器的分辨率

**解:** 答案为 D

3. 有一个 CRT 的分辨率是  $1024 \times 768$  像素, 颜色数为 256 色, 则刷新存储器的容量是\_\_\_\_\_。

A. 768KB B. 512KB C. 256KB D. 2MB

**解:** 答案为 A

4. 黑白显示器具有\_\_\_\_\_ 级灰度。

A. 1 B. 2 C. 4 D. 16

**解:** 答案为 B

5. 为看到稳定图像, 显示器刷新频率应大于\_\_\_\_\_。

- A. 15 帧/S      B. 30 帧/S      C. 50 帧/S      D. 100 帧/S

**解：**答案为 B

6. 图形显示器可显示\_\_\_\_\_，图像显示器可显示\_\_\_\_\_。

- A. 有亮暗层次的照片      B. 没有亮暗层次的线条图  
C. 立体图形      D. 有亮暗层次的线条图

**解：**答案为 B, A

7. 显示器的灰度级是指\_\_\_\_\_。

- A. 显示器的亮度      B. 显示字符的清晰程度  
C. 显示器中光点亮暗的层次级别      D. 显示器上能显示的光点数

**解：**答案为 C

8. 计算机外部设备是指\_\_\_\_\_。

- A. 输入输出设备      B. 外存设备  
C. 远程通信设备      D. 除 CPU 和内存以外的其他设备

**解：**答案为 D

9. CRT 的颜色数为真彩色，则它的颜色位是\_\_\_\_\_。

- A. 24 位      B. 8 位      C. 16 位      D. 32 位

**解：**答案为 A

10. D/A 转换器是\_\_\_\_\_。

- A. 把数字量转为模拟量，并把转化结果输入到计算机。  
B. 把模拟量转为数字量，并把转化结果输入到计算机。  
C. 把计算机输出的模拟量转换成数字量。  
D. 把计算机输出的数字量转换成模拟量。

**解：**答案为 D

11. 双向打印机的特点是\_\_\_\_\_。

- A. 左右双向同时打印。  
B. 既可选择从左向右打印，也可选择从右向左打印。  
C. 先从左向右打印，再从右向左打印。  
D. 具有两个打印头。

**解：**答案为 C

12. 显示器的控制逻辑的存储逻辑一般都在\_\_\_\_\_A\_\_\_\_\_中。终端是由\_\_\_\_\_B\_\_\_\_\_组成的一套独立的\_\_\_\_\_C\_\_\_\_\_设备，它能完成显示控制与存储、键盘管理及通信控制等功能，还可完成简单的编辑操作。

A: ①主机内部    ②显示器内部    ③主机接口板中

B: ①显示器和控制逻辑    ②键盘与显示器    ③键盘与控制逻辑

C: ①输入    ②输出    ③输入输出

**解：**答案为 A: ③; B: ②; C: ③

13. 几种打印机的特点可归纳如下: 串行点阵式打印机是按\_\_\_\_\_A\_\_\_\_\_打印的, 打印速度\_\_\_\_\_B\_\_\_\_\_; 喷墨打印机是按\_\_\_\_\_C\_\_\_\_\_，速度\_\_\_\_\_D\_\_\_\_\_；激光打印机是按\_\_\_\_\_E\_\_\_\_\_打印的, 速度\_\_\_\_\_F\_\_\_\_\_。行式点阵打印机是按\_\_\_\_\_G\_\_\_\_\_打印的, 速度是\_\_\_\_\_H\_\_\_\_\_。所有打印机的打印都受到打印字符点阵的控制。打印字符的点阵信息在点阵式打印中控制打印针\_\_\_\_\_I\_\_\_\_\_，在激光打印机中控制激光束\_\_\_\_\_J\_\_\_\_\_。

A, C, E, G: ①字符    ②行    ③页

B, D, F, H: ①最快    ②最慢    ③较快    ④中等

I, J: ①运动方向    ②有无    ③是否动作



**解：**答案为 A: ①; B: ②; C: ①; D: ④; E: ③; F: ③; G: ②; H: ③; I: ③; J: ②

14. 一级汉字有 3755 个, 如每个汉字字模采用  $24 \times 24$  点阵, 并存放在主存中, 则约占 A 字节。若将汉字显示在荧光屏上, 共 24 行, 每行 80 个字, 为保证一帧信息, 约需 B 字节的存储空间。汉字在输入时采用 C, 在存储时采用 D, 打印或显示时用 E。存储一个汉字一般可用        字节, 有时也用 G 字节。

A, B: ①30K ②60K ③90K ④270K

C, D, E: ①ASCII 码 ②字形码 ③机内码 ④点阵 ⑤拼音码 ⑥区位码

F, G: ①1 个 ②2 个 ③3 个 ④32 个 ⑤16 个

**解：**答案为 A: ④; B: ②; C: ②, ⑤, ⑥; D③; E④; F②; G③。

### 三、填空题

1. 著名的 Amdahl 定律告诉我们, 计算机的性能主要由                                  决定。

**答：**计算机的性能主要由系统中最慢的部分 (称为系统瓶颈) 决定。

2. 计算机的主机包括       ,       ,       。

**答：**运算器, 控制器, 主存储器。

3. 计算机的外部设备包括       ,       ,       。

**答：**输入设备, 输出设备, 外存储器。

4. 从人们使用的角度, 可把外部设备大致分成以下三大类, 即       ,       ,        等。

**答：**人-机交互设备, 计算机信息的驻在设备, 机-机通信设备。

5. 鼠标器按其内部结构的不同可分为                 、                、                 三大类。

**答：**机械式、光机式和光电式三大类。

6. 显示设备按使用的显示器件分, 常用的有       ,       ,       。

**答：**CRT 显示器, LCD 显示器, 等离子显示器。

7. PDP 是指                                  的平板显示器件的总称。

**答：**是指所有利用气体放电而发光的平板显示器件的总称。

8. 按打印原理分类, 打印机可分成        和        两类。

**答：**击打式和非击打式两大类。

9. 激光打印机的打印过程一般需经历以下六个步骤, 即充电、扫描曝光、      ,       ,       ,        等。

**答：**显影、转印、定影和清除残像。

10. 激光打印机主要由                 、                、                 和接口控制器等几部分组成。

**答：**激光扫描系统、电子照相系统、字符发生器。

11. 打印机有两种工作模式, 即和                  和                 。

**答：**文本模式 (字符模式) 和图形模式。

12. 虚拟环境产生器实质上是一个包括                                  的高性能计算机系统。

**答：**是一个包括虚拟世界数据库的高性能计算机系统。

13. 语音输入的实质是                 , 就是让计算机能正确识别、准确理解                 。

**答：**语音识别, 就是让计算机能正确识别、准确理解自然语言。

14. 终端与显示器是两个不同的概念, 终端的结构比显示器复杂, 它能完成                 , 及通信控制等, 还可完成                 。

**答：**它能完成显示控制与存储、键盘管理及通信控制等, 还可完成简单的编辑操作。

15. 为了能用普通电话线进行计算机通信, 应当把要发送的数字信号        成音频信号, 送到目的地后再        成数字信号。完成这一功能的设备称为调制解调器 MODEM。

答：调制，解调。

#### 四、计算题

1. 显示器的灰度级反映每个光点的亮暗级别，用二进制数据表示，若用 4 位、8 位、16 位二进制数，分别可表示多少级灰度或多少种颜色？

解： $2^4=16$ ， $2^8=256$ ， $2^{16}=65536$ 。

2. 现有一显示器的分辨率是  $1024 \times 1024$ ，灰度级为 64K，求其刷新存储器的容量。

解： $1024 \times 1024 \times 2B = 2MB$ 。

3. 显示器的刷新频率是 50 帧/S，若分辨率为  $640 \times 480$ ，灰度级为 256，求其刷新存储器的容量和读出速度。

解：(1) 刷新存储器容量为  $640 \times 480 \times 256 \text{bit} = 307200 \times 1B = 300KB$

(2)  $307.2KB \times 50 = 15MB/S$

4. 某光栅扫描显示器的分辨率是  $1024 \times 768$ ，帧频为 50 帧/S（逐行扫描），垂直回扫和水平回扫时间忽略不计，求显示器的行频是多少？若考虑以 32 个像素为一个存取单位，则求每一像素允许的读出时间是多少？

解：行频为  $768 \text{ 行} \times 50/S = 38,400 \text{ 行/S}$

每个像素允许读出时间小于  $1 \div (38400 \times 1024) \approx 0.0254 \mu S$ ；

若考虑以 32 个像素为单位存取，其读出时间也需小于  $0.0254 \times 32 = 0.8 \mu S$ 。

### 第 7 章 输入输出数据传送控制方式

#### 习题解析

##### 一、判断题

1. 所有的数据传送方式都必须由 CPU 控制实现。

答：错误。

2. 屏蔽所有的中断源，即为关中断。

答：错误。

3. 一旦中断请求出现，CPU 立即停止当前指令的执行，转去受理中断请求。

答：错误。

4. CPU 响应中断时，暂停运行当前程序，自动转移到中断服务程序。

答：正确。

5. 中断方式一般适合于随机出现的服务。

答：正确。

6. DMA 设备的中断级别比其他外设高，否则可能引起数据丢失。

答：正确。

7. CPU 在响应中断后可立即响应更高优先级的中断请求（不考虑中断优先级的动态分配）。

答：错误。

8. DMA 控制器和 CPU 可同时使用总线。

答：错误。

9. DMA 是主存与外设之间交换数据的方式，也可用于主存与主存之间的数据交换。

答：错误。

10. 为保证中断服务程序执行完毕以后，能正确返回到被中断的断点继续执行程序，必须进行现场保存操作。

答：正确。

## 二、选择题

1. 主机、外设不能并行工作的方式\_\_\_\_\_。

- A. 程序查询方式      B. 中断方式      C. 通道方式

**解：**答案为 A

2. 在单独（独立）编址下，下面的说法是\_\_\_\_\_ 是对的。

- A. 一个具体地址只能对应输入输出设备  
B. 一个具体地址只能对应内存单元  
C. 一个具体地址既可对应输入输出设备，也可对应内存单元  
D. 只对应内存单元或只对应 I/O 设备

**解：**答案为 C

3. 在关中断状态，不可响应的中断是\_\_\_\_\_。

- A. 硬件中断              B. 软件中断  
C. 可屏蔽中断          D. 不可屏蔽中断

**解：**答案为 C

4. 禁止中断的功能可由\_\_\_\_\_ 来完成。

- A. 中断触发器          B. 中断允许触发器  
C. 中断屏蔽触发器      D. 中断禁止触发器

**解：**答案为 B

5. 在微机系统中，主机与高速硬盘进行数据交换一般用\_\_\_\_\_ 方式。

- A. 程序中断控制      B. DMA  
C. 程序直接控制      D. 通道方式

**解：**答案为 B

6. 常用于大型计算机的控制方式是\_\_\_\_\_。

- A. 程序中断控制      B. DMA  
C. 程序直接控制      D. 通道方式

**解：**答案为 D

7. 有关中断的论述不正确的是\_\_\_\_\_。

- A. CPU 和 I/O 设备可并行工作，但设备间不可并行工作  
B. 可实现多道程序、分时操作、实时操作  
C. 对硬盘采用中断可能引起数据丢失  
D. 计算机的中断源可来自主机，也可来自外设

**解：**答案为 A

8. 中断系统是由\_\_\_\_\_ 实现的

- A. 仅用硬件              B. 仅用软件  
C. 软、硬件结合      D. 以上都不对

**解：**答案为 D

9. DMA 数据的传送是以\_\_\_\_\_ 为单位进行的。

- A. 字节                  B. 字  
C. 数据块              D. 位

**解：**答案为 C

10. DMA 是在\_\_\_\_\_ 之间建立的直接数据通路。

- A. CPU 与外设          B. 主存与外设  
C. 外设与外设          D. CPU 与主存

**解：**答案为 B

11. 数组多路通道数据的传送是以\_\_\_\_\_ 为单位进行的。

- A. 字节
- B. 字
- C. 数据块
- D. 位

**解：**答案为 C

12. 通道是特殊的处理器，它有自己的\_\_\_\_\_，故并行工作能力较强。

- A. 运算器
- B. 存储器
- C. 指令和程序
- D. 以上均有

**解：**答案为 C

13. 下列 I/O 控制方式中，主要由程序实现的是\_\_\_\_\_。

- A. PPU（外围处理机）
- B. 中断方式
- C. DMA 方式
- D. 通道方式

**解：**答案为 B

14. 产生中断的条件是\_\_\_\_\_。

- A. 一条指令执行结束
- B. 机器内部发生故障
- C. 一次 I/O 操作开始
- D. 一次 DMA 操作开始

**解：**答案为 B，C，D

15. 在微机系统中，外设通过\_\_\_\_\_ 与主板的系统总线相连接。

- A. 适配器
- B. 设备控制器
- C. 计数器
- D. 寄存器

**解：**答案为 A

16. 对于低速输入输出设备，应当选用的通道是\_\_\_\_\_。

- A. 数组多路通道
- B. 字节多路通道
- C. 选择通道
- D. DMA 专用通道

**解：**答案为 B

### 三、填空题

1. 实现输入输出数据传送方式分成三种：\_\_\_\_\_、\_\_\_\_\_ 和程序控制方式。

**答：**DMA 方式、通道方式。

2. 输入输出设备寻址方式有\_\_\_\_\_ 和\_\_\_\_\_。

**答：**统一编址、独立编址。

3. CPU 响应中断时最先完成的两个步骤是\_\_\_\_\_ 和\_\_\_\_\_。

**答：**关中断、保存现场信息。

4. 内部中断是由\_\_\_\_\_ 引起的，如运算溢出等。

**答：**CPU 的某种内部因素。

5. 外部中断是由\_\_\_\_\_ 引起的，如输入输出设备产生的中断。

**答：**主机外部的中断信号。

6. DMA 的含义是\_\_\_\_\_，用于解决\_\_\_\_\_。

**答：**直接存储器访问，数据块传送。

7. DMA 数据传送过程可分为\_\_\_\_\_、数据块传送和\_\_\_\_\_ 三个阶段。

**答：**传输前预处理，传送后处理。

8. 基本 DMA 控制器主要由\_\_\_\_\_、\_\_\_\_\_、数据寄存器、控制逻辑、标志寄存器及地址译码与同步电路组成。

**答：**地址计数器、字计数器。

9. 在中断服务中, 开中断的目的是允许\_\_\_\_\_。

答: 多级中断。

10. 一个中断向量对应一个\_\_\_\_\_。

答: 中断服务程序的入口地址。

11. 接口收到中断响应信号 INTA 后, 将\_\_\_\_\_ 传送给 CPU。

答: 中断向量。

12. 中断屏蔽的作用有两个, 即\_\_\_\_\_ 和\_\_\_\_\_。

答: 改变中断处理的优先级别, 屏蔽一些不允许产生的中断。

13. 串行接口之所以需要串、并数据的转换电路, 是因为\_\_\_\_\_。

答: 计算机内部数据是并行传送的。

14. CPU 响应中断时, 必须先保护当前程序的断点状态, 然后才能执行中断服务程序, 这里的断点状态是指\_\_\_\_\_。

答: 返回地址和状态寄存器的内容。

15. 通道是一个特殊功能的\_\_\_\_\_, 它有自己的\_\_\_\_\_ 专门负责数据输入输出的传送控制, CPU 只负责\_\_\_\_\_ 的功能。

答: 处理器, 指令和程序, 数据处理。

16. CPU 对外设的控制方式按 CPU 的介入程度, 从小到大为\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_。

答: 通道方式, DMA 方式, 中断方式, 程序方式。

#### 四、计算题

1. 若输入输出系统采用字节多路通道控制方式, 共有 8 个子通道, 各子通道每次传送一个字节, 已知整个通道最大传送速率为 1200B/S, 求每个子通道的最大传输速率是多少? 若是数组多路通道, 求每个子通道的最大传输速率是多少?

解: 字节多路通道: 每个子通道的最大传输速率是  $1200\text{B/S} \div 8 = 150\text{B/S}$

数组多路通道: 每个子通道的最大传输速率是 1200B/S。

2. 某字节多路通道共有 6 个子通道, 若通道最大传送速率为 1500B/S, 求每个子通道的最大传输速率是多少?

解:  $1500\text{B/S} \div 6 = 250\text{B/S}$ 。

3. 在 8086CPU 中, 1 号中断向量的地址是从 0004~0007H, 其中地址 0004~0005 存放 200H, 地址 0006~0007 存放 2000H, 试问:

(1) 1 号中断的中断服务程序的入口地址是多少?

(2) 4 号中断的向量地址是多少?

解: (1) 入口地址为 2000H: 0200H

(2)  $0 + 4 \times 4 = 16 = 10\text{H}$

4. 在 8255A 并行接口中, 下图是其端口 C 的置 1/置 0 控制字, 请读下列程序, 它实现什么功能?

0	D <sub>6</sub>	D <sub>5</sub>	D <sub>4</sub>	D <sub>3</sub>	D <sub>2</sub>	D <sub>1</sub>	D <sub>0</sub>
---	----------------	----------------	----------------	----------------	----------------	----------------	----------------

↖ 不用 ↗

位选择

000: C 口位 0

001: C 口 1 位

:

↑

1: 置位

0: 复位

MOV AL, 0DH

OUT 63H, AL

MOV AL, 0CH

OUT 63H, AL

**解：**程序完成先使 PC6=1，再使 PC6=0。

5. 用异步方式传送 ASCII 码，数据格式为：数据位 8 位、奇校验位 1 位、停止位 1 位。当波特率为 4800b/s 时，每个字符传送的速率是多少？每个数据位的时间长度是多少？数据位的传送速率又是多少？

**解：**每个字符包含 10 位，因此字符传送速率为： $4800 \div 10 = 480$  字符/s

每个数据位长度  $T = 1 \div 4800 \approx 0.208\text{ms}$

数据位传输速率为  $8 \times 480 = 3840$  位/秒。

6. 假定某外设向 CPU 传送信息最高频率为 40K 次/秒，而相应中断处理程序的执行时间为  $40 \mu\text{S}$ ，问该外设能否用中断方式工作？

**解：**外设传送一个数据的时间  $= 1/40 \times 10^3 = 25 \mu\text{S}$ ，所以请求中断的周期为  $25 \mu\text{S}$ ，而相应中断处理程序的执行时间为  $40 \mu\text{S}$ ，这样会丢失数据，故不能采用中断方式。

## 第 8 章 总线系统

本章将给出主教材《计算机组成原理》第 8 章的习题及补充题的解析。仅供读者参考。

### 8. 1 主教材习题解析

#### 一、判断题

1. 组成总线不仅要有传输信息的传输线，还应有实现总线传输控制的器件，即总线缓冲器和总线控制器。

**答：**正确。

2. 大多数微机总线由地址总线、数据总线和控制总线组成，因此，它们是三总线结构的。

**答：**错误。

3. 在计算机总线中，地址信息、数据信息和控制信息不能同时出现。

**答：**正确。

4. 三态缓冲门可组成运算器的数据总线，它的输出电平有逻辑“1”或逻辑“0”、高阻抗三种状态。

**答：**正确。

5. 计算机使用总线结构的主要优点是便于积木化，同时减少了信息传输线的数目。

**答：**正确。

6. 任何类型的计算机都包含地址总线、数据总线和控制总线。

**答：**正确。

7. 地址线的条数反映了微处理器的寻址能力。

**答：**正确。

8. 地址总线的特点是可双向传输，控制总线的特点是双向传输。

**答：**错误。

9. USB 全速版的最大传输速率比 USB 高速版大。

**答：**错误。

10. 总线的发展与 CPU 的发展休戚相关，CPU 的主频提高后，总线的数据传输如果不

随之提高，必将妨碍整机性能的提高。

答：正确。

## 二. 选择题

1. CPU 的芯片中的总线属于\_\_\_\_\_。

解：答案为 A

- A. 内部
- B. 局部
- C. 系统
- D. 板级

2. 下面所列的\_\_\_\_\_ 不属于系统总线接口的功能。

- A. 数据缓冲
- B. 数据转换
- C. 状态设置
- D. 完成算术和逻辑运算

解：答案为 D

3. 信息只用一条传输线，且采用脉冲传输的方式称为\_\_\_\_\_。

- A. 串行传输
- B. 并行传输
- C. 并串行传输
- D. 分时传输

解：答案为 A

4. 在\_\_\_\_\_ 的计算机系统中，外设可以主存储器单元统一编址。

- A. 单总线
- B. 双总线
- C. 三总线
- D. 以上三种都可以

解：答案为 A

5. 系统总线中地址线的功能是\_\_\_\_\_。

- A. 用于选择主存单元地址
- B. 用于选择进行信息传输的设备
- C. 用于选择外存地址
- D. 用于指定主存和 I/O 设备接口电路的地址

解：答案为 D

6. 同步通信比异步通信具有较高的传输频率，这是因为\_\_\_\_\_。

- A. 同步通信不需要应答信号。
- B. 同步通信方式的总线长度较短。
- C. 同步通信一个公共时钟信号进行同步。
- D. 同步通信中各部件存取时间比较短。

解：答案为 C

7. 异步控制常用于\_\_\_\_\_ 作为其主要控制方式。

- A. 在单总线结构计算机中访问主存与外设时
- B. 微机的 CPU 控制中
- C. 组合逻辑控制的 CPU 控制中
- D. 微程序控制器中

解：答案为 A

8. 在 3 种集中式总线控制中，\_\_\_\_\_方式响应时间最快，\_\_\_\_\_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

解：答案为 A

9. 在计数器定时查询方式下，若从 0 开始计数，则\_\_\_\_\_。

- A. 用于选择主存单元地址
- B. 用于选择进行信息传输的设备
- C. 用于选择外存地址
- D. 用于指定主存和 I/O 设备接口电路的地址

10. CPU 的控制总线提供\_\_\_\_\_。

- 解：答案为 D

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

12. 在独立方式下, 若有几个设备, 则\_\_\_\_\_。

- 解：答案为 A

A. 有几条总线请求信号  
B. 公用一条总线请求信号  
C. 有  $n-1$  条总线请求信号  
D. 无法确定

### 三. 填空题

答：总线。

答：芯片。

答：集中、分布。

答：没有、一个公共。

答：握手（就绪/应答）。

答：总线仲裁，总线协议。

答: 数据传输速率。

答：复用。



9. 总线协议是指\_\_\_\_\_。

答：实现总线数据传输的定时规则。

10. 总线设备与总线的连接界面是\_\_\_\_\_。

答：总线接口。

11. 总线的基本特性包括\_\_\_\_\_、\_\_\_\_\_和电气特性。

答：物理特性，功能特性。

12. 总线控制主要解决\_\_\_\_\_问题。集中式仲裁有\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_。

答：总线控制权，链式查询方式，计数器定时查询方式，独立请求方式。

13. USB 接口通过使用\_\_\_\_\_，理论上可使一台 PC 机连接的外设多达\_\_\_\_\_台。

答：集线器（HUB），127 台。

14. 在单总线、双总线、三总线三种系统中，从信息流传送效率的角度看，\_\_\_\_\_的工作效率最低，从吞吐量来看，\_\_\_\_\_最强。

答：单总线，双总线。

15. AGP 总线不同于 PCI 总线，它是提供\_\_\_\_\_专用的，它在\_\_\_\_\_与系统内存之间提供了一条直接访问的途径。

答：图形加速卡，图形卡。

16. 根据总线传输的信息内容的不同，总线可分为\_\_\_\_\_、\_\_\_\_\_和\_\_\_\_\_。任何类型的计算机总线都包括这三种总线。

答：数据，地址，控制总线。

#### 四. 计算题

1. 在一个 16 位的总线中，若时钟频率为 100MHz，总线数据周期为 5 个时钟周期传输一个字。试计算总线的数据传输率。

解：时钟频率为 100MHz，所以

$$5 \text{ 个时钟周期} = 5 \times 100 \mu\text{s} = 0.05 \mu\text{s}$$

$$\text{数据传输率} = 16\text{bit}/0.05 \mu\text{s} = 40 \times 10^6 \text{ 字节/秒}$$

2. (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：(1)设带宽用  $D_r$  表示，总线时钟周期用  $T=1/f$  表示，一个总线周期传送的数据量用  $D$  表示，根据定义可得

$$D_r = D/T = D \times f = 4\text{B} \times 33 \times 10^6/\text{S} = 132\text{MB/S}$$

$$(2) \text{因为 } 64 \text{ 位} = 8\text{B}, \text{ 所以 } D_r = D/T = D \times f = 8\text{B} \times 66 \times 10^6/\text{S} = 528\text{MB/S}$$

(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。

3. 在异步串行传输系统中，若每秒可传输 20 个数据帧，一个数据帧包含 1 个起始位、7 个数据位、一个奇校验位和 1 个结束位。试计算其波特率和比特率。

解：波特率 =  $(1+7+1+1) \times 20 = 200\text{b/s}$ ，比特率 =  $20 \times 7 = 140\text{b/s}$ 。

#### 第 9 章 计算机系统结构的的发展

##### 习题解析

## 一. 选择题

1. 下面描述的流水 CPU 基本概念中, 不正确的表达是\_\_\_\_\_。

- A. 流水 CPU 是以空间并行性为原理构造的处理器
- B. 流水 CPU 一定是 RISC 机器
- C. 流水 CPU 一定是多媒体 CPU
- D. 流水 CPU 是以时间并行性为原理构造的处理器

**解:** 答案为 A, B, C。

2. 流水 CPU 是由一系列叫做“段”的处理部件构成的, 和具备 m 个并行部件的 CPU 相比, 一个 m 段流水 CPU\_\_\_\_\_。

- A. 具备同等水平的吞吐能力
- B. 不具备同等水平的吞吐能力
- C. 吞吐能力小于前者的吞吐能力
- D. 吞吐能力大于前者的吞吐能力

**解:** 答案为 A。

3. 在高速计算机中, 广泛采用流水线技术。例如, 可以将指令执行分成取指令、分析指令和执行指令 3 个阶段, 不同指令的不同阶段可以\_\_\_\_\_①\_\_\_\_\_执行; 各阶段的执行时间最好\_\_\_\_\_②\_\_\_\_\_; 否则在流水线运行时, 每个阶段的执行时间应取\_\_\_\_\_③\_\_\_\_\_。

可供选择的答案:

- ① A.顺序      B.重叠      C.循环      D.并行
- ② A.为 0      B.为 1 个周期      C.相等      D.不等
- ③ A. 3 个阶段执行时间之和
- B. 3 个阶段执行时间的平均值
- C. 3 个阶段执行时间的最小值
- D. 3 个阶段执行时间的最大值

**解:** 答案为(1) D (2) C (3) D。

4. 光学计算机的特点是\_\_\_\_\_。(多选)

- A. 比电子计算机快
- B. 比电子计算机功耗大
- C. 光传输失真大
- D. 光器件开关速度比电子器件快
- E. 不同波长的光波会发生干涉
- F. 光器件带宽比电子器件大

**解:** 答案为 A、D、F。

5. CPU 内使用流水线技术后, 下列可能的说法是\_\_\_\_\_。(多选)

- A. 取指令与执行指令同步进行
- B. 取指令与执行指令异步进行
- C. 正在执行的指令与流水线中的指令冲突
- D. 流水线内的指令无效
- E. 流水线取指令与执行的指令有关
- F. 流水线取指令与执行的指令无关

**解:** 答案为 A、C、D、E、F。

## 二. 填空题

1. 并行处理技术已成为计算机发展的主流, 它有三种形式: \_\_\_\_\_并行, \_\_\_\_\_并行和\_\_\_\_\_并行。

**答:** 时间并行, 空间并行, 时间+空间并行。

2. Neumann 计算机属于\_\_\_\_\_驱动方式, 数据流计算机属于\_\_\_\_\_驱动方式, 归约计算机属于\_\_\_\_\_驱动方式。

**答:** 控制驱动, 数据驱动, 需求驱动。

3. 光学计算机是利用\_\_\_\_\_实现的计算机。

答：光技术和光器件。

4. 生物计算机是利用\_\_\_\_\_而研制开发的一种新型计算机。

答：生物系统固有的信息处理机理。

5. 量子计算机是基于\_\_\_\_\_的信息处理方式，按照一定的体系结构，采用\_\_\_\_\_所构成的计算机。

答：基于量子力学理论和量子器件的信息处理方式，采用量子器件。

6. 按并行等级技术分类，流水线可分为\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_三类。

答：指令流水线、算术流水线和处理机流水线。

7. 流水 CPU 中的主要问题是\_\_\_\_\_相关，\_\_\_\_\_相关和\_\_\_\_\_相关。

答：资源、数据、控制。

8. 现代计算机的发展受半导体材料限制，人们正转向\_\_\_\_\_、\_\_\_\_\_、\_\_\_\_\_的研制，软件方面也开展\_\_\_\_\_的研究。

答：生物计算机、光计算机、量子计算机、人工智能。

### 三. 计算题

1. 现有 4 级流水线，分别完成取指令、指令译码并取数、运算、送结果四步操作，若完成各步操作的时间依次为 100ns、100ns、80ns、50ns，请求：

(1) 流水线的操作周期。

(2) 若相邻两条指令发生数据相关，且在硬件上不采取措施，那么第二条指令要推迟的时间。

(3) 若对硬件进行改进，那么第二条指令至少要推迟的时间。

解：(1) 流水线的操作时间应按各步操作的最大时间来考虑，应为 100ns。

(2) 若相邻两条指令发生数据相关，且在硬件上不采取措施，应停顿第 2 条指令的执行，直到前面的指令结果已经产生，因此要推迟 2 个时钟周期的时间。

(3) 若对硬件进行改进，如采用专用的通路技术，那么第 2 条指令的执行将不会被推迟。

2. 在 4 段流水线浮点加法器中，若每段所需的时间为： $T_1=60\text{ns}$ ， $T_2=50\text{ns}$ ， $T_3=90\text{ns}$ ， $T_4=80\text{ns}$ 。

求：(1) 流水线加法器的加速比。

(2) 若每段的时间都是 75ns（包括缓冲时间），求加速比。

解：(1) 加法器的流水线时钟周期至少为  $T=90\text{ns}$ 。若采用同样的逻辑电路（不使用流水线），则浮点加法所需的时间是：

$$T_1 + T_2 + T_3 + T_4 = 60 + 50 + 90 + 80 = 280\text{ns}$$

因此，加速比为  $280/90 \approx 3.1$

(2) 若每个过程段的时间都是 75ns，则加速比 =  $(75 \times 4) / 75 = 4$

3. 假设一条指令按取指、分析和执行三步解释，每步相应的时间分别为  $t_{\text{取}}$ 、 $t_{\text{分}}$ 、 $t_{\text{执}}$ ，分别计算下列几种情况下执行完 100 条指令所需的时间：

(1) 顺序方式。

(2) 仅(K+1)取指与 K 执行重叠。

(3) 仅(K+2)取指，(K+1)分析、K 执行重叠。

若  $t_{\text{取}}=t_{\text{分}}=2$ ， $t_{\text{执}}=1$ ，计算上述结果。

解：若  $t_{\text{取}}=t_{\text{分}}=2$ ， $t_{\text{执}}=1$

(1) 顺序方式： $T=100 \times 5=500\text{ns}$

(2) 一次重叠方式： $T=6+99 \times (2+2)=402\text{ns}$

(3) 2 次重叠方式： $T=99 \times 2+2+2+2+2=204\text{ns}$

# 计算机组成原理(05 年新版)复习资料

## 第一章概论

CPU：中央处理器，是计算机的核心部件，由运算器和控制器构成。

运算器：计算机中完成运算功能的部件，由 ALU 和寄存器构成。

总线：计算机中连接功能单元的公共线路，是一束信号线的集合。

主机：由 CPU、存储器与 I/O 接口合在一起构成的处理系统称为主机。

接口：是主机与外设之间传递数据与控制信息的电路，是主机与外设的桥梁。

汇编语言：采用文字方式（助记符）表示的程序设计语言。

字长：一个数据字包含的位数，一般为 8 位、16 位、32 位和 64 位等。

兼容：计算机部件的通用性。

运算器的功能：完成算术逻辑运算，由 ALU 和若干寄存器组成。其中 ALU 负责执行各种数据运算操作，寄存器用于暂时存放参与运算的数据以及保存运算状态。

控制器的功能：从内存中取出指令，对其进行译码，产生相应的时序控制信号，控制其它器件工作。

## 第二章数据编码和数据运算

数据：定点数据、浮点数据、图形数据、文字数据。

原码：用一个符号位表示数据的正负，0 代表正号，1 代表负号，其余的代码表示数据的绝对值。

补码：用最高位表示符号，其余各位代码给出数值按 2 取模的结果。

阶码：浮点数编码中，表示小数点的位置的代码。

上溢：指数据的绝对值太大，以至大于数据编码所能表示的数据范围。

海明距离：在信息编码中，两个合法代码对应位上编码不同的位数。

冯诺依曼舍入法：浮点数据的一种舍入方法，在截去多余位时，将剩下数据的最低位置 1。

桶形移位器：一种移位电路，具有移 2 位、移 4 位和移 8 位等功能。

规格化数：浮点数编码中，为使浮点数具有唯一的表示方式所作的规定，规定尾数部分用纯小数形式给出，而且尾数的绝对值应大于  $1/R$ ，即小数点后的第一位不为零。

机器零：浮点数编码中，阶码和尾数为全 0 时代表的 0 值。

为什么用二进制：容易用数据电路表示，数据运算和存储方式简单，是高效的数据表示方式。

如何区分 ASCII 代码和汉字编码：ASCII 代码是 7 位的代码，在存储时可以在它前面增加一位形成 8 位的代码，增加的位用 0 表示是 ASCII 码，1 表示是汉字编码。

采用双符号位检测溢出的方法：在运算时，两个符号位同时参加运算，结果中如果两个符号位不同，则表示产生了溢出。若符号为 01，则表示运算结果大于允许取值范围的最大正数，称为正溢出；若符号位为 10，则表示运算结果是负数，其值小于允许取值范围的最小负数，称为负溢出。两个符号位的最高位仍为正确的符号。

十进制加法器的原理：它是利用二进制定点运算器，将两个 BCD 码相加后判断是否大于 9，若是，则做十进制调整（即加 6），否则直接输出。

浮点数规格化的目的和方法：目的是使浮点数尾数的最高数值位为有效数位。当尾数用补码表示时符号位与小数点后第一位不相等则已为规格化数据，否则是非规格化数据。通过规格化，可以保证运算数据的精度。规格化中通常采用向左规格化，即尾数每左移一位，阶码减 1，直至规格化完成。

## 第三章存储系统

SRAM：静态半导体存储器，可随机读写，其存储的数据表示为晶体三极管构成的双稳态电路的电平，存储数据稳定，不需刷新。

DRAM：动态半导体存储器，它利用电容存储电荷的特性来存储数据，可以提高存储器芯片的存储容量，但必须不断地刷新每个存储单元中存储的信息。

快闪存储器：它属于电可擦写可编程只读存储器。其存储单元结构与 EEPROM 类似，只是存储单元的氧化层较薄，且有更好的电可擦性能。

多体交叉存储器：由多个相互独立、容量相同的存储体构成，每个存储体都有各自的读写电路、地址寄存器和数据寄存器，各自以等同的方式与 CPU 传递信息。CPU 可以在一个存储周期内连续访问多个存储体。

相联存储器：一种按内容访问的存储器，每个存储单元有匹配电路，可用于 cache 中查找数据。

固件：固化在硬件中（如写入 ROM）的固定不变的常用软件。

EDO DRAM：增强数据输出动态随机访问存储器，采用快速页面访问模式，并增加了一个数据锁存器以提高数据传输速率。存储芯片由存储体、读写电路、地址译码和控制电路等组成。存储芯片并联的目的是为了位扩展，串联的目的是为了字节单元扩展。

访问存储器的速度指标：访问时间、访问周期、带宽。

访问局部性规律：程序对存储空间的 90% 的访问局限于存储空间的 10% 的区域中，而另外 10% 的访问则分布在存储空间的其余 90% 的区域中。

地址映射：采用虚拟存储技术执行程序时，必须把逻辑地址映射到主存储器的物理地址空间上，

称为地址映射。

存储器周期：从一次启动存储器操作到操作完成后可启动下一次操作的时间。

带宽：存储器在连续访问时的数据吞吐速率。

访问时间：从启动访问存储器的操作到操作完成的时间。

命中：访问主存的数据或代码存在于 cache 中的情形称为命中。

主存与 BACHE 之间的映象方式：有直接映象、全相联印象、组相联印象三种。直接映象是指主存储

器中的每个块只能映象到 CACHE 中唯一一个指定块的地址映象方式。全相联映象是指每个主存块

都能够映象到任一 CACHE 块的地址映象方式。组相联印象是直接映象和全相联映象两种方式的结

合，它将存储空间分成若干组，在组间直接映象方式，而在组内使用全相联印象方式。

CACHE 的替换策略：随机法是用一个随机数产生器产生一个随机的替换块号；先进先出法是替换最

早调入的存储单元；近期最少用法替换近期最少使用的存储。

CACHE 的更新策略：写操作 CACHE 命中时，CACHE 更新策略有两种：①写直达法：将内容同时写入 CACHE 和主存。②写回法：将内容只写入 CACHE，当 CACHE 数据被替换出去才写回主存。写操作 CACHE 不命中时，更新策略有两种：①按写分配法：当 CACHE 不命中时将该地址对应的块从主存调入 CACHE。②当 CACHE 不命中时将该地址对应的块不从主存调入 CACHE。

虚拟存储器的管理方式：页式虚拟存储器是把虚拟存储空间和实际存储空间等分成固定容量的页，各虚拟页可装入中不同的实际页面位置；段式虚拟存储器是将主存按段分配，段长度不固定，由 OS 为程序分配各段；段页式是前两种的结合，它将存储空间按逻辑模块分段再分成若干页

通过段表和页表进行访存。

页表的作用：是反映逻辑页号和物理页号的对应关系，用于实现虚实地址的变换。页表由与逻辑页相同数量的表单元构成，每个单元包含有装入位和物理页号。装入位表示相应的逻辑页是否在主存中，若在则物理页号表示在哪一个物理页中。

提高存储器工作速度：①芯片技术：快速页式动态存储器（FPM DRAM）、同步型动态存储器 SDRAM、相联存储器；②结构技术：增加存储器的数据宽度、采用多体交叉存储器。

#### 第四章指令系统

指令：是计算机硬件能够识别并直接执行的操作命令，又称机器指令。

操作码：指令中指定操作类型的代码。

地址码：指令中指定操作数的字段。

助记符：用一些比较容易记忆的文字符号来表示指令中的代码和操作数。

堆栈：是一种按 LIFO 顺序进行访问的存储区。

寻址方式：蕴含寻址、立即数寻址、寄存器寻址、直接寻址、寄存器间接寻址、存储器间接寻址、相寻址、变址和基址寻址、复合寻址方式。

指令系统：计算机中各种指令的集合，或指令集。

指令类型：数据传送、算术运算、逻辑运算、程序流控、IO 操作、堆栈操作、捷符串处理、系统指令。

RISC 的特点：优点：RISC 技术简化了指令系统，以寄存器—寄存器方式工作，采用流水方式总从而在一个时钟周期内执行完毕，使用较多的寄存器以减少访存，绝大部分指令采用组合电路实现。缺点：指令功能简单使得程序代码较长，占用了较多的存储器空间。

转子指令执行过程：①将下一条指令的地址（PC 的值）存放在一个临时存储位置，以便于子程序返回时取出这个地址，继续执行下一条指令；②将子程序的起始地址装入 PC 中，这样取指令时将读取子程序中的指令。子程序的最后一条指令一般是一条返回指令，它将存放在临时存储位置的指令地址取出，放回 PC，这样程序就返回原程序了。

基址与变址的区别：基址寻址用于程序定位，一般由硬件或 OS 完成。而变址寻址是面向用户的，用于对一级数据进行访问等。

相对寻址的特点：操作数的地址是程序计数器 PC 的值加上偏移量形成的，是一种特殊的变址寻址方式，偏移量用补码表示，可正可负。相对寻址可用较短的地址码访问内存。

#### 第五章控制器

CPU 的基本功能：指令控制；操作控制；数据运算；异常处理和中断处理。

控制器的功能：从内存中取指令，并计算下一条指令在内存中的地址；对指令进行译码，产生相

应的操作控制信号；控制指令执行的步骤和数据流动的方向。

CPU 寄存器类型：指令寄存器 IR；程序计数器 PC；数据寄存器 DR；地址寄存器 AR；状态寄存器 SR。

硬件连线方式：是用一个时序电路产生时间控制信号，采用组合逻辑电路实现各种控制功能。

微程序方式：采用存储逻辑实现，即用一个存储在 ROM 中的微程序产生控制信号。

指令周期：从一条指令的启动到下一条指令的启动的间隔时间。

机器周期：是指令执行中每一步操作所需要的时间。

时钟周期：是计算机主频的周期。

单总线运算指令周期的执行过程：①送指令地址；②计算下一条指令的地址；③读入指令；④把一个源操作数寄存器中的数据送到运算器中；⑤把另一个源操作数寄存器中的数据送到运算器；⑥把结果送到指令指定的寄存器。

条件转移指令的三种安排方法：条件码方法；条件寄存器方法；比较与转移方法。

转移指令的执行过程：①取指令；②指令译码；③计算地址。

组合电路的硬连线控制器的构成：时钟源、环形脉冲发生器、控制信号的编码/译码逻辑电路。

微程序控制器的构成：控制存储器、微指令寄存器  $\mu IR$ 、微地址寄存器  $\mu AR$ 、地址转移逻辑等。

微程序：存储在控制存储器中的完成指令功能的程序，由微指令组成。

微指令：控制器存储的控制代码，分为操作控制部分和顺序控制部分。

微地址：微指令在控制存储器中的存储地址。

相容性微操作：在同时或同一个 CPU 周期内可以并行执行的微操作。

相斥性微操作：不能在同时或不能在同一个 CPU 周期内并行执行的微操作。

微指令的编码方法：直接表示法；编码表示法；混合表示法。

产生后继微地址的方式：计数器方式；断定方式；结合方式。

水平型微指令：一次能定义并执行多个并行操作控制信号的微指令。

垂直型微指令：采用微操作码编码法，由微操作码规定微指令的功能。包括微操作码、源部件号、目标部件号、下一微指令地址。

指令传真：是一种获得机器软件兼容的方法，是使得已有的软件能够在新型的计算机中继续运行。

指令模拟：在一种计算机上用软件来解释另一种计算机的指令。

微程序控制器的特点：与硬连线控制器比较，具有规整性，可维护性的优点，是一种用软件设计方法来设计硬件的技术，可实现复杂指令的操作控制，且极其灵活性，可方便地添加和修改指令。

## 第六章系统总线

总线：一组可由多个部件分时共享的信息传输线。

总线的物理特性包括：物理连接方式、连线的类型、连线的数量、接插件的形状尺寸、引脚线的排列方式。

总线的功能特性包括：功能层次、资源类型、信息传递类型、信息传递方式和控制方式。

总线的电气特性包括：信号传递方向、信号的时序特征、电平特征。

总线电平信号的表示方式：单端方式，用一条信号线和一条公共接地线来传递信号。高电平表示 1，低电平表示 0。差分方式采用两条信号线互补传输信号，即两条线都传输信号，但两条线上的信号是电平相反的。差分方式具有抗干扰能力强的特点，可提高数据传输速度或增加总线的长度。

传输方式：串行、并行、复合、消息。

串行传输：数据的传输在一条线路上按位进行。

并行传输：所有的数据位同时传输。

复合传输：是一种总线复用的传输方式，它使不同的信号在同一条信号线上传输。

消息传输：是将总线需要传送的数据信息、地址信息和控制信息等组成一个固定的数据结构以猝发方式进行传输。

码元：携带数据信息的信号单元。

波特率：每秒钟通过信道传输的码元数。

比特率：每秒钟通过信道传输的信息量。

猝发式数据传输：在一个总线周期传输存储地址连续的多个数据字。

总线协议：规定了实现总线数据传输的定时规则。

同步通信：所有的设备都从一个公共的时钟信号中获得定时信息。

异步通信：使用一个在 CPU 和设备之间的“握手”信号，去除了公共的时钟信号，从而使

得操作变成异步的。非互锁、半互锁、全互锁。

主设备：获得总线控制权的设备。

从设备：被主设备访问的设备。

总线事务：从请求总线到完成总线使用的操作序列。

总线访问延迟：是主设备为获得总线控制权而等待的时间。

总线周期：是主设备占用总线的时间。

总线裁决方式：决定总线由哪个设备进行控制的方式。

集中式控制将总线的控制功能用一个专门的部件实现，这个部件可以位于连接在总线的某个设备上。链式查询、计数器定时查询、独立请求方式。

菊花链方式：各申请总线的设备合用一条总线作为请求信号线，而总线控制设备的响应信号线则串接在各设备间。

独立请求方式：集中式总线裁决方式之一，每一个设备都有一个独立的总线请求信送到总线控制器，控制器也给各设备分别发送一个总线响应信号。

计数器定时查询方式：集中式总线裁决方式之一，设备要求使用总线时通过一条公用请求线发出，总线控制器按计数器的值对各设备进行查询。

总线接口：是总线设备与总线的连接界面。

系统总线接口：是 CPU、内存、外围设备之间的连接的逻辑部件。

系统总线接口的基本功能：控制；数据缓存；状态设置；数据转换；整理；程序中断。

串行数据位格式：起始位、数据位、检验位、结束位、空闲位。

指定位时间：指定一个二进制位在传输线上占用的时间长度，它是由同步脉冲来体现的。

通用异步接收/发送器 UART：由接收器、发送器、控制电路、状态寄存器组成。

并行总线接口包括：系统时钟驱动器、电压控制模块、总线主设备、总线从设备、总线请求部件、中断处理部件、总线仲裁器、中断部件。

UART 中出错标志：1 奇偶错误，表示在接收时收到的每一个字符的 1 的个数不符合要求。2 帧错误，

表示接收到的字符格式不符合规定。3 溢出错误，表示缓冲寄存器已经满了不能接收新的数据，到达的数据将丢失。

表示接收到的字符格式不符合规定。3 溢出错误，表示缓冲寄存器已经满了不能接收新的数据，到达的数据将丢失。

达的数据将丢失。

UART 工作过程：接收时，由串行输入端来的串行数据先进行移位寄存器，然后并行地输入到缓冲

寄存器，从而将并行数据由数据总线传输到 CPU；发送时，CPU 的数据先送到发送器的缓冲寄存

器，然后送到移位寄存器，由串行输出端一位一位地串行输出。在工作时，接收器部分始终监视

着串行输入端线，当发现一个起始位时，就开始了一个新字符的接收过程。

提高总线速度的措施：从物理层次：1 增加总线宽度；2 增加传输的数据长度；3 缩短总线长度；4 降低信号电平；5 采用差分信号；6 采用多条总线。从逻辑层次：1 简化总线传输协议；2 采用总线复用技术；3 采用消息传输协议。

PCI 总线的层次：局部总线、PCI 总线、IO 总线。

PCI 定义了三个物理地址空间：内存地址空间、IO 地址空间、配置地址空间。

## 第 7 章外围设备

磁盘访问时间：指从发出读写命令，磁头从某一起始位置移动到新的记录位置，到开始从盘片读出或写入信息所花的时间。= 寻道时间 + 旋转延迟 + 控制延迟 + 数据传输时间。



寻道时间：是将磁头定位到所要求的磁道上所需的时间。

旋转延迟：是找道完成后到磁道上需要访问的信息到达磁头的时间。

平均旋转延迟：是磁盘旋转半周的时间，也称磁盘的寻址时间。

数据传输时间取决于读扇区数据时间和传输数据时间，等于两者的最大值。

磁盘数据传输率=转速/秒 \* 每道容量

## 第 8 章输入输出系统

外设寻址方法：统一编址法、单独编址法。

统一编址法：将 IO 设备中的控制寄存器、数据寄存器、状态寄存器和内存单元联合在一起编排地址。

单独编址法：采用专门的控制信号进行 IO 操作，内存的地址空间和 IO 设备的地址空间是分开的，需要使用专门的 IO 指令。

外设的定时方式：异步应答方式、同步定时方式。

数据传送方式：程序控制方式（程序查询和中断方式）、DMA 方式、通道方式。

程序查询方式的操作过程：输入：1CPU 把一个地址值放在地址总线上，选择某一输入设备；2CPU 等待输入设备的数据成为有效；3CPU 从数据总线输入数据，放在寄存器中。输出：1CPU 把一个地址值放在地址总线上，选择某一输出设备；2CPU 把数据放在数据总线上；3 输出设备认为数据有效，将数据取走。

中断请求信号：由外设发出的信号。

中断：是一种在发生了一个外部的事件时调用相应的处理程序的过程，这个过程中包括了程序的返回。

中断响应过程：1 关中断；2 保存现场信息；3 识别发出中断的设备，判别中断条件，以确定中断服务程序的入口地址；4 执行中断服务程序；5 从中断服务程序返回；6 开中断，继续执行原程序。

中断向量：由发出中断请求的设备通过输入输出总线主动向 CPU 发出一个识别代码。

中断裁决机制：轮询、菊花链、独立请求。

单重中断方式：CPU 在处理一个中断时禁止其它中断的方式。

多重中断方式：CPU 在处理一个中断期间允许被其它中断打断。

多重中断系统中，CPU 响应的步骤：1 关中断；2 保存现场信息；3 判别中断条件，确定中断源；4 开中断；5 执行中断服务程序；6 关中断；7 恢复现场信息，包括 PC 的值；8 开中断，继续执行原程序。

中断方式的接口控制器功能：能向 CPU 发出中断请求信号；能实现 CPU 对中断请求允许或禁止的控制；能使中断请求参加优先级排队；能发出中断向量提供引导 CPU 在响应中断请求后转入相应服务程序的地址。

内部中断：是由 CPU 的某种内部因素引起的，称为自陷。

外部中断：是由中断信号引起的中断。

软件中断：由自陷指令引起的中断，用于调用 OS 服务程序。

中断源分析：是根据不同的中断请求生成不同的中断向量或中断程序入口地址，使中断的处理交由系统中断服务软件进行进一步的处理。

DMA 方式：直接依靠硬件实现主存与外设之间的数据直接传输，传输过程本身不需 CPU 程序干预。

DMA 传输过程：传输前预处理、数据传输、传送后处理。

DMA 预处理：由 CPU 执行几条 IO 指令，测试设备状态，向 DMA 控制器的设备地址寄存器中送入设备号并启动设备，同时向内存地址计数器中送入起始地址，向字节计数器中送入交换的数据个数。

DMA 后处理：传输完成后向 CPU 发出中断请求，一旦 DMA 的中断请求得到响应，CPU 停止主程序的执行，转去执行中断服务程序，做一些 DMA 的结束工作，包括校验送入传输的数据、继续用 DMA 传输或停止 DMA 工作。

DMA 数据传输的基本操作：1 从外围设备发出 IO 请求；2 CPU 响应请求，DMA 控制器从 CPU 接管总线控制；3 由 DMA 控制器对内存寻址，并执行数据传输操作；DMA 控制器向 CPU 报告操作的结束。

DMA 访问裁决方法：CPU 等待 DMA 的操作、DMA 乘存储器空闲时访问、CPU 与 DMA 交替访问存储器。

DMA 组成：地址计数器、字计数器、数据寄存器、控制逻辑、标志寄存器。

CPU 启动 DMA 的步骤：1 测试设备状态；2 写存储器地址计数器；3 写字数计数器；4 启动 DMA 控制逻辑。

通道：是一个具有输入输出处理器控制的 IO 部件。

选择通道：它与设备之间的传输一直维持到设备请求的传输完成为止，然后为其它外围设备传输数据。数据宽度是可变的，通道中包含一个保存 IO 数据传输所需的参数寄存器。

数组多路通道：以数组为单元在若干高速传输操作之间进行交叉复用。

字节多路通道：用于连接多个慢速的和中速的设备，这些设备的数据传送以字节为单位。

字节交叉模式、猝发模式。

字节多路通道与数组多路通道的区别：首先数组多路通道允许多个设备同时工作，但只允许一个设备进行传输型操作，而其它设备进行控制型操作；字节多路通道不仅允许多个设备操作，而且允许它们同时进行传输型操作。其次，数组多路通道与设备之间的数据传送的基本单位是数据块，通道必须为一个设备传送完一个数据块以后才能为别的设备传送数据，而字节多路通道与设备之间的数据传送基本单位是字节，各设备之间的数据传送是以字节为单位交替进行的。

通道的功能：1 接受 CPU 的 IO 操作指令，按指令要求控制外围设备；2 从内存中读取通道程序，并执行，即向设备控制器发送各种命令；3 组织和控制数据在内存与外设之间的传送操作；4 读取外设的状态信息，形成整个通道的状态信息，提供给 CPU 或保存在内存中；5 向 CPU 发出 IO 操作中中断请求，将外围设备的中断请求和通道本身的中断请求按次序报告 CPU。

DMA 控制方式的特点：优点是传输速度快，适合于成批数据传送，只在传送开始和结束时需要 CPU 参与，数据传送过程中无须 CPU 干预，提高了 CPU 的效率。

# 课程设计

## 1 关于设计

### 1.1 设计目的及要求

通过一个完整的 8 位指令系统结构 (ISA) 的设计和实现, 加深对计算机组成原理课程内容的理解, 建立起整机系统的概念, 掌握计算机设计的基本方法, 培养科学的工作作风和分析、解决实际问题的能力。

综合运用计算机组成原理、数字逻辑和汇编语言等相关课程的知识, 理解和熟悉计算机系统的组成原理, 掌握计算机主要功能部件的工作原理和设计方法, 掌握指令系统结构设计的一般方法, 掌握并运用微程序设计 (Microprogramming) 思想, 在设计过程中能够发现、分析和解决各种问题, 自行设计自己的指令系统结构 (ISA)。

### 1.2 设计内容

基于 TDN-CM++ 计算机组成原理实验教学系统, 设计和实现一个 8 位指令系统结构 (ISA), 通过调试和运行, 使设计的计算机系统能够完成指定的功能。

设计过程中要求考虑到以下各方面的问题:

- (1) 指令系统风格 (寄存器-寄存器, 寄存器-存储器, 存储器-存储器);
- (2) 数据类型 (无符号数, 有符号数, 整型, 浮点型);
- (3) 存储器划分 (指令, 数据);
- (4) 寻址方式 (立即数寻址, 寄存器寻址, 直接寻址等);
- (5) 指令格式 (单字节, 双字节, 多字节);
- (6) 指令功能类别 (算术 / 逻辑运算, 存储器访问, 寄存器操作, 程序流控制, 输入 / 输出);
- (7) 依据 CPI 值对指令系统进行性能分析。

### 1.3 仪器

TDN-CM++ 计算机组成原理实验教学系统一台, 排线

## 2 设计分析

### 2.1 时序分析

TDN-CM++计算机组成原理实验教学系统时序功能部件：

实验所用的时序电路原理如下图 2.1 可产生四个等间隔的时序信号 TS1~TS4，其中  $\phi$  为时钟信号，由实验台左上方的方波信号源提供，可产生频率及脉宽可调的方波信号。在本实验中，TS1~TS4 具体的控制为：

TS1：控制输入伪指令并校验

TS2：控制微指令寄存器及地址寄存器的写入

TS3：控制指令寄存器和地址寄存器的写入、W/R 信号的变化

具体有：LDAR、LDIR

TS4：控制指令译码电路及数据通路中的信号

具体有：LDDR1、LDDR2、LDR0、LDR1、LDR2、LDPC、299B

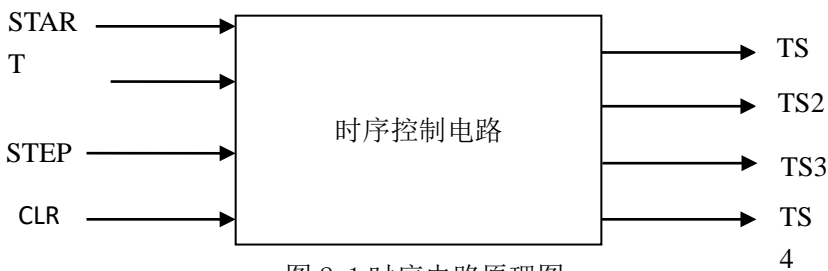


图 2.1 时序电路原理图

### 2.2 微指令格式：

微指令字长共 24 位，其控制顺序表如表 2-1：

表 2-1 微指令格式

24	23	22	21	20	19	18	17	16	15 14 13	12 11 10	9 8 7	6	5	4	3	2	1
S3	S2	S1	S0	M	Cn	WE	A9	A8	A	B	C	uA5	uA0	uA0	uA0	uA0	uA0

其中 uA5~uA0 为 6 位的后续微地址，A、B、C 为三个译码字段，分别有三个控制位译码出多位（如表 2-2）。字段 A9、A8 控制 I/O 译码电路，对 Y0、Y1、Y2 进行选择输出；字段 WE 为控制对主存 W/R 的开关；字段 24~19 控制运算器的控制端，通过改变 S3~CN 来决定对数据是否进行运算或者进行什么运算。

其中 C 字段中的 P(1)~P(4) 是四个测试字段。其功能是根据机器指令及相应微代码进行译码，使微程序转入相应的微地址入口，从而实现微程序的顺序、分

支、循环运行。AR 为算术运算是否影响进位及判零标志控制位，为零有效。

A9、A8 控制如表 2-3：

表 2-3 A9、A8 的控制

A9	A8	选中
0	0	Y0
0	1	Y1
1	0	Y2
1	1	Y3

其中 Y1、Y0、Y1、Y2、Y3 都为低电平有效。

表 2-2

A 字段

15	14	13	选择
0	0	0	
0	0	1	LDRi
0	1	0	LDDR1
0	1	1	LDDR2
1	0	0	LDIR
1	0	1	LOAD
1	1	0	LDAR

B 字段

12	11	10	选择
0	0	0	
0	0	1	RS-B
0	1	0	RD-B
0	1	1	RI-B
1	0	0	299-B
1	0	1	ALU-B
1	1	0	PC-B

C 字段

9	8	7	选择
0	0	0	
0	0	1	P(1)
0	1	0	P(2)
0	1	1	P(3)
1	0	0	P(4)
1	0	1	AR
1	1	0	LDPC

2.3 指令译码电路

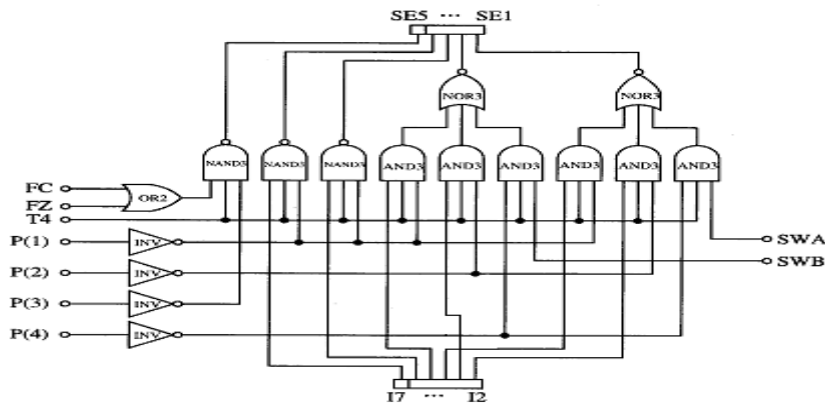


图 2.2 指令译码电路图

图 2.2 为指令译码电路，图中 P(1)~P(4) 是测试字段，低电平有效，I7~I2 为指令寄存器的第 7~2 为输出，SE5~SE1 为为控制单元微地址锁存器的强制端输出。

后续微地址的形成：

由此电路图可以写出以下逻辑方程：

$$SE5 = \overline{(FC + FZ)} * \overline{T4} * P(3)$$

$$SE4 = \overline{I7} * \overline{T4} * P(1)$$

$$SE3 = \overline{I6} * \overline{T4} * P(1)$$

$$SE2 = \overline{I5} * \overline{T4} * P(1) + \overline{I3} * \overline{T4} * P(2) + \overline{T4} * P(4) * SWB$$

$$SE1 = \overline{I4} * \overline{T4} * P(1) + \overline{I2} * \overline{T4} * P(2) + \overline{T4} * P(4) * SWA$$

由微控制器实验原理图可以看出后续微地址 uA5~uA0 分别 SE5~SE0 的非，结合上述逻辑方程可以得出 I7~I2 对应 uA3~ uA0 由此可以强制微指令的后续微地址，从而强制了下一要执行哪一条指令。

## 2.4 寄存器组控制电路（如图 2.3）：

图中 RS-B、RD-B、RI-B 分别为源寄存器选通信号、目的寄存器选通信号及变址寄存器选通信号，其功能是根据机器指令来进行三个工作寄存器 R0、R1 及 R2 的选通译码，I0~I3 为指令寄存器的第 0~3 位，LDRi 为打入工作寄存器的译码器使能控制位。当 I1、I0 为 00 时，选通 LDR0，当 I1、I0 为 01 时，选通 LDR1，当 I1、I0 为 10 时，选通 LDR2。当 I3、I2 为 00 时，选通 R1-B，当 I3、I2 为 01 时，选通 R1-B，当 I3、I2 为 10 时，选通 R2-B。

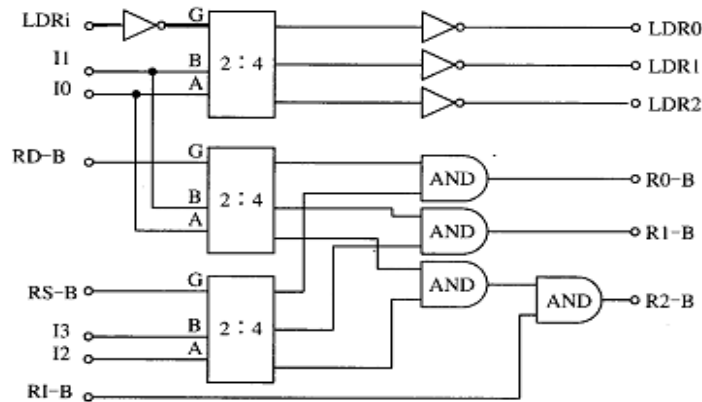


图 2.3 寄存器译码电路图

## 3 初步设计

3.1 指令格式

本实验中共设计了 13 条机器指令：IN（输入）、ADDi（加立即数）、STA（存数）、OUT（输出）、JMP（无条件跳转）、SUB（算术减）、AND（逻辑与）、LW（取数）、RS（逻辑右移）、MOVER（寄存器—寄存器的移动）、MOVEM（存储器—存储器的移动）、STOP（停止）（如表 3-1）。

表 3-1 指令格式

指令类别		助记符	字节	机器指令码	说明
输入输出		IN (R0)	单	00000000	[INPUT] → R0
		OUT (R0 addr)	双	00110000	R0 → addr
				addr	
运 算 类 指 令	立即数加	ADDi (R0 addr]	双	00010000	R0 + [imme] → R0
				imme	
	算术减	SUB (R0 addr)	双	01010001	R0 - [ADDR] → R1
				addr	
	逻辑与	AND (R0 addr)	双	01100010	R0. [addr] → R0
				addr	
	逻辑移位 (右)	RS (R0)	单	10010000	R0 >> 1 → R0
	一元非	NOT (R0 addr)	双	10000000	$\overline{[addr]} \rightarrow R0$
				Addr	
	存/取数据指令	STA (Ri addr)	双	00100000	addr → Ri
				addr	
		LW (R0 addr)	双	01110000	[RAM] → R0
				addr	

跳转类指令		JMP(addr)	双	01000000	无条件跳转到 addr
				addr	
数据 传送 类指 令	寄存器- 寄存器	MOVER(R0 R1)	单	10100001	R0→R1
	存储器- 存储器	MOVEM(addr1 addr2)	三	10110000	Addr1→addr2
				addr1	
				addr2	
停止类指令		STOP	单	11000000	STOP

3.2 控制台流程

图 3.1 中一个矩形方框表示一条微指令，方框中的内容为该指令执行的微操作，右上角的数字是该调指令的微地址，用八进制表示，向下的箭头指出了下一条要执行的指令。本控制台以 P(4)为测试字，根据条件使微指令产生分支。

其中：

KRD 为存储器读操作：拨动总清开关 CLR 后，控制台开关 SWB、SWA 为“00”时，按 START 微动开关，可对 RAM 进行连续手动读操作。

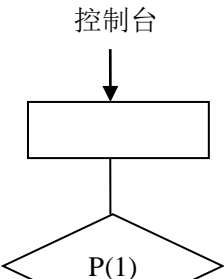
KWE 为存储器写操作：拨动总清开关 CLR 后，控制台开关 SWB、SWA 为“01”时，按 START 微动开关，可对 RAM 进行连续手动写操作。

RP 为启动程序：拨动总清开关 CLR 后，控制台开关 SWB、SWA 为“11”时，按 START 微动开关，即可转入到第 01 号“取址”微指令，启动程序运行。

表 2-2 为 SWB、SWA 状态设置表，通过此表可以看出相应的控制台指令：

表 2-2 SWB、SWA 状态设置表

SWB	SWA	控制台指令
0	0	读内存（KRD）
0	1	写内存（KWE）
1	1	启动程序（RP）





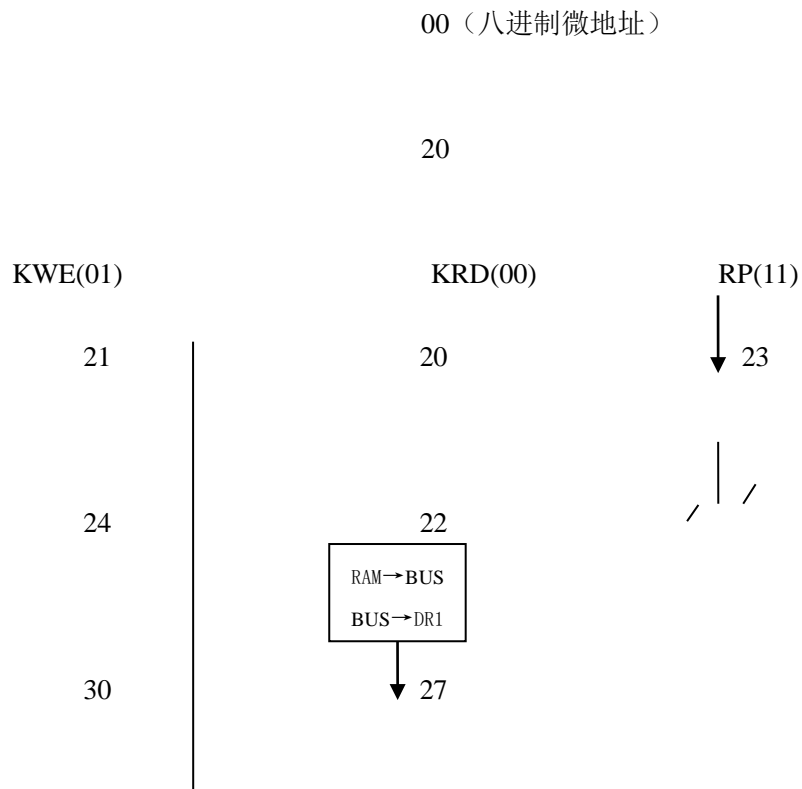


图 3.1 控制台

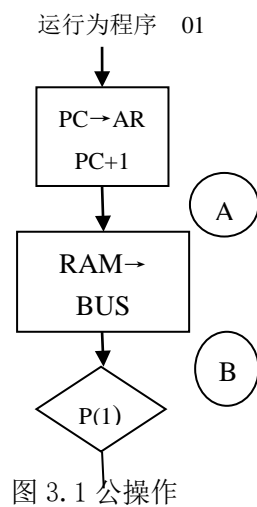
### 3.3 微程序分流程

#### 3.3.1 公操作{如图 3.1):

公操作是每条指令都必须经过的，然后经过测试字段 **P**（1）跳到要执行的指令。

结合表 2-2 并由表 3-2 可以看出 **A** 字段选中 LDAR, **B** 字段选中 PC-B, **C** 字段选中 LDPC, 从而实现了 PC+1。

表 3-1 A



S3	0
S2	0
S1	0
S0	0
M	0
CN	0
WE	0
A9	1
A8	1
A	110
B	110
C	110

表 3-2 B

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	1 1	100	000	001

结合表 2-2 并由表 3-2 可以看出 A 字段选中 LDIR, B 字段无操作, C 字段选中 P(1), 从而经过译码跳到要执行的指令。

3.3.2 输入指令 IN (如图 3.2)

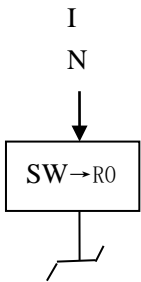


图 3.2 IN

表 3-3IN

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	0 0	001	000	001

根据后续微地址的形成方法 (2.3 中已叙述) 可以由公操作跳到 IN, 综合表 2-2 由表 3-3 可以看出 A 字段选中了 LDRi, B、C 字段都为空, A9, A8 选中 Y0, 从而打开开关 SW-B, 将数据打到 R0。

3.3.3 加立即数 ADDi 指令 (如图 3.3)

表 3-4 ADDi (1)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	1 1	110	110	110

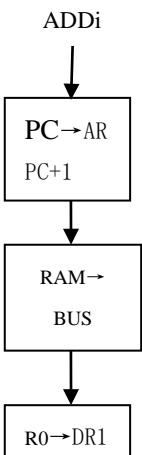


图 3.3 ADDi

根据后续微地址的形成方法（2.3 中已叙述）可以由公操作跳到加立即数指令 ADDi，综合表 2-2 由表 3-4 可以看出 A 字段选中了 LDAR, B 字段选中 PC-B, C 字段选中 LDPC, 从而实现了 PC+1。

表 3-5 ADDi (2)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	0 1	011	000	000

综合表 2-2 由表 3-5 可以看出 A 字段选中了 LDDR2, B、C 字段为空, A9、A8 为 01，选中 Y1，将 MEM 的 CE 端打开，将 RAM 中的数据通过总线打入 DR2.

表 3-6 ADDi (3)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	1 1	010	001	000

综合表 2-2 由表 3-6 可以看出 A 字段选中了 LDDR1，打开锁存器 DR1，B 字段选中了 RD-B，A9、A8 为 11，选中 Y3，将 R0 寄存器中的数据送至 DR1 。

表 3-7 ADDi (4)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
1 0 0 1 0 1	0	1 1	001	101	000

由 S3~S0 决定做加运算，综合表 2-2 由表 3-7 可以看出 A 字段选中了

LDRi, R0, B

字段选中 ALU-B，执行运算，并把结果打回 R0。

3.3.4 取数据指令 STA（如图 3.4）

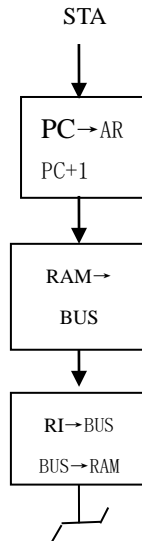


图 3.4 STA

表 3-8 STA(1)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	0	0	0	1	1	110	110	110

根据后续微地址的形成方法（2.3 中已叙述）可以由公操作跳到存数据指令 STA，综合表 2-2 由表 3-8 可以看出 A 字段选中了 LDAR, B 字段选中 PC-B, C 字段选中 LDPC, 从而实现了 PC+1。

表 3-9 STA(2)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	0	0	0	0	1	110	000	000

综合表 2-2 由表 3-9 可以看出 A 字段选中了 LDAR , B、C 字段为空, A9、A8 为 01，选中 Y1，将 MEM 的 CE 端打开，将 RAM 中的数据通过总线打入 AR。

表 3-10 STA(3)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	0	0	1	0	1	000	001	000

综合表 2-2 由表 3-10 可以看出 A 、C 字段为空，B 字段选中 RS-B，A9, A8 为 01，选中 Y1，打开将 RAM 的 CE 端，WE=1 时执行写操作，将 R0 寄存器中的内

### 3.3.5 输出数据指令 OUT (如图 3.5)

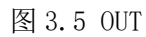


表 3-11 OUT(1)

根据后续微地址的形成方法（2.3 中已叙述）可以由公操作跳到输出指令 OUT, 综合表 2-2 由表 3-11 可以看出 A 字段选中了 LDAR, B 字段选中 PC-B, C 字段选中 LDPC, 从而实现了 PC+1。

表 3-12 OUT(2)

综合表 2-2 由表 3-12 可以看出 A 字段选中了 LDAR ,B、C 字段为空,A9、A8 为 01,选中 Y1,将 MEM 的 CE 端打开,将 RAM 中的数据通过总线打入 AR。

表 3-13 OUT (3)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
----	----	----	----	---	----	----	----	----	---	---	---

0	0	0	0	0	0	0	0	0	1	010	000	000
---	---	---	---	---	---	---	---	---	---	-----	-----	-----

综合表 2-2 由表 3-13 可以看出 A 字段选中了 LDDR1,B、C 字段为空,A9、A8 为 01,选中 Y1,将 MEM 的 CE 端打开,将 RAM 中的数据通过总线打入 DR1.

表 3-14 OUT (4)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	0	1	1	1	0	000	101	000

综合表 2-2 由表 3-13 可以看出 A、C 字段为空, B 字段选中了 ALU-B, WE=1 进行写操作,A9 A8 为 10,选中 Y2, 打开 LED-B,将数据显示出来。

### 3.3.6 无条件跳转指令 JMP (如图 3.6)

根据后续微地址的形成方法 (2.3 图 2.3 所述) 可以由公操作跳到无条件跳转指令 JMP,综合表 2-2 由表 3-15 可以看出 A 字段选中了 LDAR,B 字段选中 PC-B,C 字段选中 LDPC,从而实现了 PC+1。

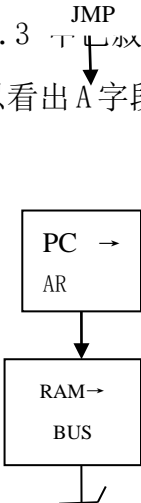


图 3.6 JMP

表 3-15 JMP (1)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	0	0	0	1	1	110	110	110

表 3-16 JMP (2)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	0	0	0	0	1	101	000	110

综合表 2-2 由表 3-16 可以看出 A 字段选中了 LOAD ,B 字段为空,C 字段选中 LDPC, A9、A8 为 01,选中 Y1,将 MEM 的 CE 端打开,将 RAM 中的数据通过总线打向指向的 PC。

### 3.3.7 算术减指令 SUB (如图 3.7)

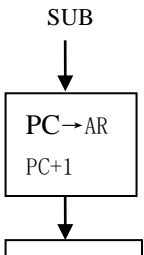


图 3.7 SUB

表 3-17 SUB (1)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	1 1	110	110	110

根据后续微地址的形成方法（2.3 中已叙述）可以由公操作跳到算术减指令 SUB，综合表 2-2 由表 3-17 可以看出 A 字段选中了 LDAR, B 字段选中 PC-B, C 字段选中 LDPC, 从而实现了 PC+1。

表 3-18 SUB(2)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	0 1	110	000	000

综合表 2-2 由表 3-18 可以看出 A 字段选中了 LDAR , B、C 字段为空, A9、A8 为 01，选中 Y1，将 MEM 的 CE 端打开，将 RAM 中的数据通过总线打入 AR。

表 3-19 SUB (3)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	0 1	011	000	000

综合表 2-2 由表 3-19 可以看出 A 字段选中了 LDDR2, B、C 字段为空, A9、A8 为 01，选中 Y1，将 MEM 的 CE 端打开，将 RAM 中的数据通过总线打入 DR2。

表 3-20 SUB(4)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C

0	0	0	0	0	0	0	0	1	1	010	001	000
---	---	---	---	---	---	---	---	---	---	-----	-----	-----

综合表 2-2 由表 3-20 可以看出 A 字段选中了 LDDR1，打开锁存器 DR1，B 字段选中了 RD-B，

A9、A8 为 11，选中 Y3，将 R0 寄存器中的数据送至 DR1 。

表 3-21 SUB(5)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	1	1	0	0	0	0	1	1	001	101	000

由 S3~S0 决定做算术减运算，综合表 2-2 由表 3-21 可以看出 A 字段选中了 LDRi，打开 R0，B 字段选中 ALU-B，执行运算，并把结果打回 R1。

### 3.3.8 逻辑与 AND(如图 3.8)

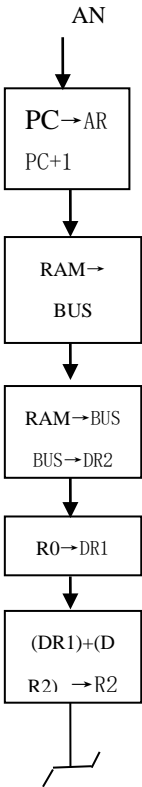


图 3.8 AND

表 3-22 AND（1）

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	0	0	0	1	1	110	110	110

根据后续微地址的形成方法（2.3 中已叙述）可以由公操作跳到逻辑与运



算 AND，综合表 2-2 由表 3-22 可以看出 A 字段选中了 LDAR,B 字段选中 PC-B,C 字段选中 LDPC,从而实现了 PC+1。

表 3-23 AND(2)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	0 1	110	000	000

综合表 2-2 由表 3-23 可以看出 A 字段选中了 LDAR ,B、C 字段为空,A9、A8 为 01，选中 Y1，将 MEM 的 CE 端打开，将 RAM 中的数据通过总线打入 AR。

表 3-24 AND（3）

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	0 1	011	000	000

综合表 2-2 由表 3-24 可以看出 A 字段选中了 LDDR2,B、C 字段为空,A9、A8 为 01，选中 Y1，将 MEM 的 CE 端打开，将 RAM 中的数据通过总线打入 DR2。

表 3-25 AND(4)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	1 1	010	001	000

综合表 2-2 由表 3-25 可以看出 A 字段选中了 LDDR1，打开锁存器 DR1，B 字  
段选中

了 RD-B，A9、A8 为 11，选中 Y3，将 R0 寄存器中的数据送至 DR1 。

表 3-26 AND(5)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
1 0 1 1 1 0	0	1 1	001	101	000

由 S3~S0 决定做逻辑与运算，综合表 2-2 由表 3-26 可以看出 A 字段选中了 LDRi, 打开 R0,B 字段选中 ALU-B，执行运算，并把结果打回 R2。

3.3.9 取数据指令 LW（如图 3.9）

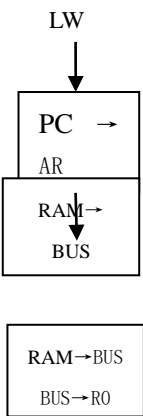




图 3.9 LW

表 3-27 LW (1)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	1 1	110	110	110

根据后续微地址的形成方法（2.3 中已叙述）可以由公操作跳到取数据指令 LW，综合表 2-2 由表 3-27 可以看出 A 字段选中了 LDAR,B 字段选中 PC-B,C 字段选中 LDPC,从而实现了 PC+1。

表 3-28 LW(2)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	0 1	110	000	000

综合表 2-2 由表 3-28 可以看出 A 字段选中了 LDAR ,B、C 字段为空,A9、A8 为 01，选中 Y1，将 MEM 的 CE 端打开，将 RAM 中的数据通过总线打入 AR。

表 3-29 LW(3)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	0 1	001	000	000

综合表 2-2 由表 3-28 可以看出 A 字段选中了 LDRi ,B、C 字段为空,A9、A8 为 01，选中 Y1，将 MEM 的 CE 端打开，将 RAM 中的数据通过总线打入 R0。

3.3.10 非指令 NOT(如图 3.10)

表 3-30 NOT(1)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C

0	0	0	0	0	0	0	0	1	1	110	110	110
---	---	---	---	---	---	---	---	---	---	-----	-----	-----

根据后续微地址的形成方法（2.3 中已叙述）可以由公操作跳到输出指令 OUT，综合表 2-2 由表 3-30 可以看出 A 字段选中了 LDAR,B 字段选中 PC-B,C 字段选中 LDPC,从而实现了 PC+1。

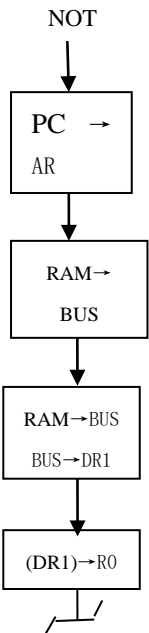


图 3.10 NOT

表 3-31 NOT (2)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	0	0	0	0	1	110	000	000

综合表 2-2 由表 3-31 可以看出 A 字段选中了 LDAR ,B、C 字段为空,A9、A8 为 01，选中 Y1，将 MEM 的 CE 端打开，将 RAM 中的数据通过总线打入 AR。

表 3-32 NOT (3)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	0	0	0	0	1	010	000	000

综合表 2-2 由表 3-32 可以看出 A 字段选中了 LDDR1,B、C 字段为空,A9、A8 为 01，选中 Y1，将 MEM 的 CE 端打开，将 RAM 中的数据通过总线打入 DR1。

表 3-33 NOT (4)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	1	0	0	1	1	001	101	000

由 S3~S0 决定做逻辑非运算，综合表 2-2 由表 3-33 可以看出 A 字段选中了 LDRi, 打开 R0, B 字段选中 ALU-B, 执行运算，并把结果打回 R0。

3.3.11 逻辑右移指令 RS (如图 3.11)

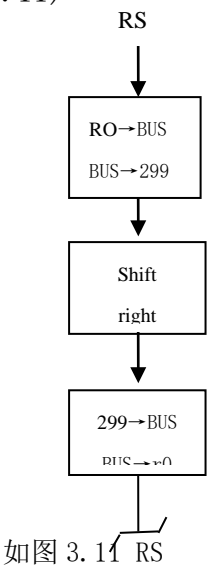


表 3-34 RS(1)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	1	1	0	0	0	1	1	000	001	000

根据后续微地址的形成方法（2.3 中已叙述）可以由公操作跳到逻辑右移指令 RS，综合表 2-2 由表 3-34 可以看出 B 字段选中了 RS-B，A、C 字段为空，将数据从 R0 取出通过总线到移位寄存器 299。

表 3-35 RS(2)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	1	0	0	0	0	1	1	000	000	000

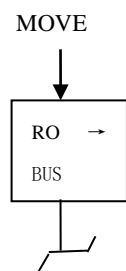
由表 3-35 可以看出 A、B、C 字段为空，299 为低电平有效，当 S1、S0 为 10，M=0 时，为循环右移。

表 3-36 RS(3)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	0	0	0	1	1	001	100	000

综合表 2-2 由表 3-36 可以看出 A 字段选中了 LDRi，B 字段选中 299-B，C 字段为空，将数据从移位寄存器取出，通过总线打到 R0。

3.3.12 寄存器-寄存器的移动指令 MOVER（如图 3.12）



如图 3.12 MOVER

表 3-37 MOVER(1)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	1 1	001	001	000

根据后续微地址的形成方法（2.3 中已叙述）可以由公操作跳到寄存器-寄存器的移动指令 MOVER，综合表 2-2 由表 3-37 可以看出 A 字段选中 LDRI，B 字段选中了 RS-B，C 字段为空，将 R0 的数据取出，打到 R1。

3.3.13 存储器-存储器指令 MOVEM-MOVEM(如图 3.13)

表 3-38 MOVEM(1)

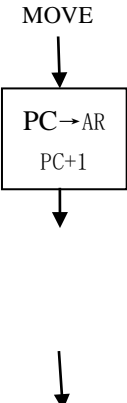
S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	1 1	110	110	110

根据后续微地址的形成方法（2.3 中已叙述）可以由公操作跳到输出指令 OUT，综合表 2-2 由表 3-38 可以看出 A 字段选中了 LDAR，B 字段选中 PC-B，C 字段选中 LDPC，从而实现了 PC+1。

表 3-39 MOVEM(2)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	0 1	110	000	000

综合表 2-2 由表 3-39 可以看出 A 字段选中了 LDAR，B、C 字段为空，A9、A8 为 01，选中 Y1，将 MEM 的 CE 端打开，将 MEM 中的数据通过总线打入 AR。



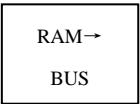


图 3.13 （MOVEM）

表 3-40 MOVEM（3）

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	0 1	010	000	000

综合表 2-2 由表 3-40 可以看出 A 字段选中了 LDDR1,B、C 字段为空,A9、A8 为 01,选中 Y1,将 MEM 的 CE 端打开,将 RAM 中的数据通过总线打入 DR1.

表 3-41 MOVEM(1)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C
0 0 0 0 0 0	0	1 1	110	110	110

根据后续微地址的形成方法（2.3 中已叙述）可以由公操作跳到输出指令 OUT,综合表 2-2 由表 3-41 可以看出 A 字段选中了 LDAR,B 字段选中 PC-B,C 字段选中 LDPC,从而实现了 PC+1。

表 3-42 MOVEM(2)

S3 S2 S1 S0 M CN	WE	A9 A8	A	B	C

0	0	0	0	0	0	0	0	0	1	110	000	000
---	---	---	---	---	---	---	---	---	---	-----	-----	-----

综合表 2-2 由表 3-42 可以看出 A 字段选中了 LDAR ,B、C 字段为空, A9、A8 为 01，选中 Y1，将 MEM 的 CE 端打开，将 RAM 中的数据通过总线打入 AR。

表 3-43 STA(3)

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	0	1	1	0	1	000	101	000

综合表 2-2 由表 3-43 可以看出 A 、C 字段为空，B 字段选中 ALU-B， A9、A8 为 01，选中 Y1，打开将 RAM 的 CE 端，WE=1 时执行写操作，将 DR1 中的数据通过总线写入 RAM。

### 3.3.14 停止指令 STOP（图 3.14）

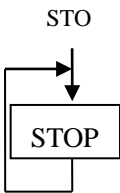


图 3.14 STOP

表 3-44 STOP

S3	S2	S1	S0	M	CN	WE	A9	A8	A	B	C
0	0	0	0	0	0	0	0	1	000	000	000

根据后续微地址的形成方法（2.3 中已叙述）可以由公操作跳到停止指令 STOP, 停止所有指令的执行。

## 4 详细设计

### 4.1 主存分配

指令区：00H-2FH

数据区：30H-3EH

## 4.2 微程序流程总图

本次设计共有 13 条指令，流程总图如图 4.1。

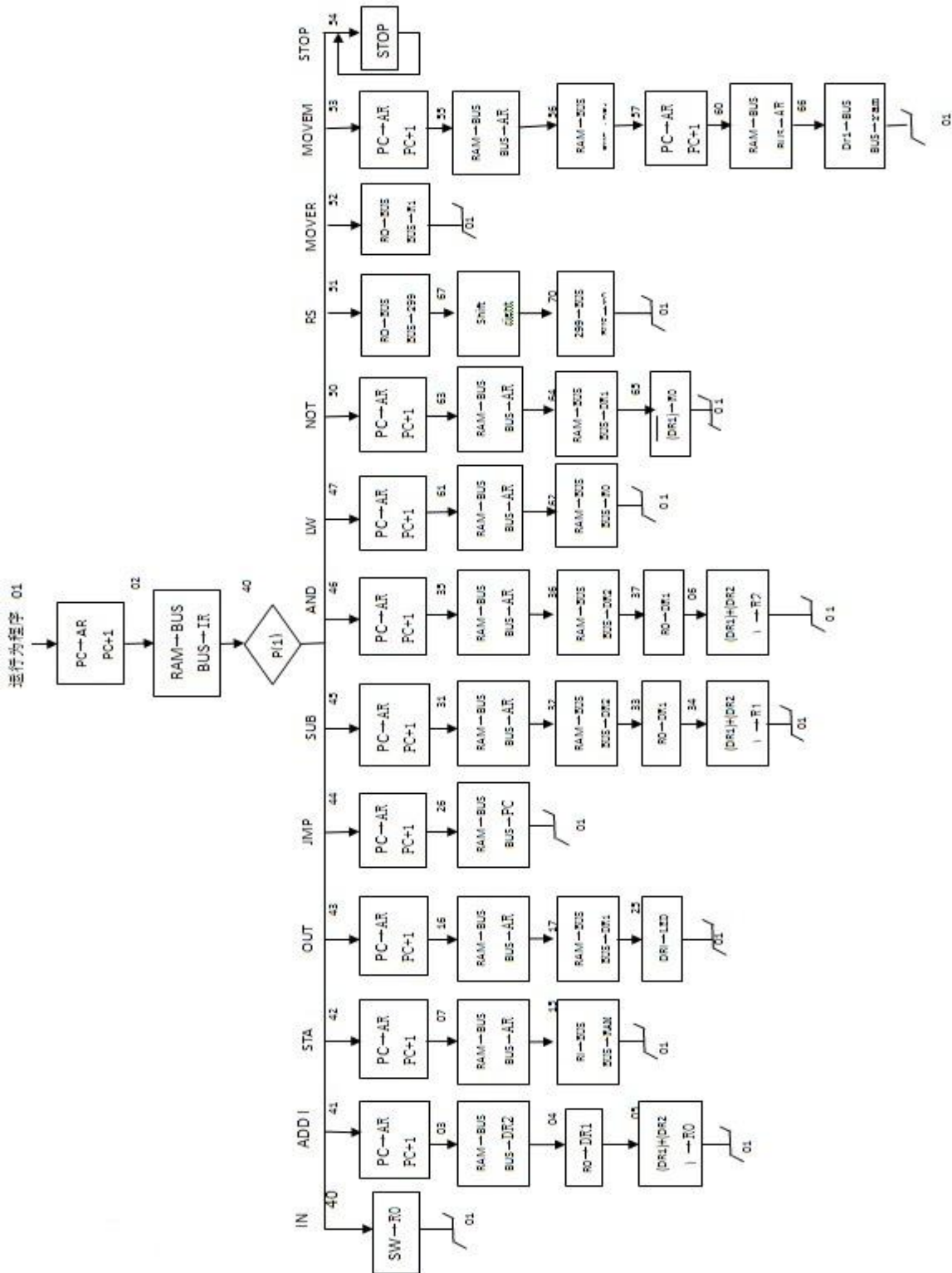


图 4.1 微程序流程总图



表 5-1 二进制代码表

微地址	S3 S2 S1 S0 M CN WE A9 A8	A	B	C	UA5...UA0
0 0	0 0 0 0 0 0 0 1 1	0 0 0	0 0 0	1 0 0	0 1 0 0 0 0
0 1	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	0 0 0 0 1 0
0 2	0 0 0 0 0 0 0 0 1	1 0 0	0 0 0	0 0 1	1 0 0 0 0 0
0 3	0 0 0 0 0 0 0 0 1	0 1 1	0 0 0	0 0 0	0 0 0 1 0 0
0 4	0 0 0 0 0 0 0 1 1	0 1 0	0 0 1	0 0 0	0 0 0 1 0 1
0 5	1 0 0 1 0 1 0 1 1	0 0 1	1 0 1	0 0 0	0 0 0 0 0 1
0 6	1 0 1 1 1 0 0 1 1	0 0 1	1 0 1	0 0 0	0 0 0 0 0 1
0 7	0 0 0 0 0 0 0 0 1	1 1 0	0 0 0	0 0 0	0 0 1 1 0 1
1 5	0 0 0 0 0 0 1 0 1	0 0 0	0 0 1	0 0 0	0 0 0 0 0 1
1 6	0 0 0 0 0 0 0 0 1	1 1 0	0 0 0	0 0 0	0 0 1 1 1 1
1 7	0 0 0 0 0 0 0 0 1	0 1 0	0 0 0	0 0 0	0 1 0 1 0 1
2 0	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	0 1 0 0 1 0
2 1	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	0 1 0 1 0 0
2 2	0 0 0 0 0 0 0 0 1	0 1 0	0 0 0	0 0 0	0 1 0 1 1 1
2 3	0 0 0 0 0 0 0 1 1	0 0 0	0 0 0	0 0 0	0 0 0 0 0 1
2 4	0 0 0 0 0 0 0 0 0	0 1 0	0 0 0	0 0 0	0 1 1 0 0 0
2 5	0 0 0 0 0 1 1 1 0	0 0 0	1 0 1	0 0 0	0 0 0 0 0 1
2 6	0 0 0 0 0 0 0 0 1	1 0 1	0 0 0	1 1 0	0 0 0 0 0 1
2 7	0 0 0 0 0 1 1 1 0	0 0 0	1 0 1	0 0 0	0 1 0 0 0 0
3 0	0 0 0 0 0 1 1 0 1	0 0 0	1 0 1	0 0 0	0 1 0 0 0 1
3 1	0 0 0 0 0 0 0 0 1	1 1 0	0 0 0	0 0 0	0 1 1 0 1 0
3 2	0 0 0 0 0 0 0 0 1	0 1 1	0 0 0	0 0 0	0 1 1 0 1 1
3 3	0 0 0 0 0 0 0 1 1	0 1 0	0 0 1	0 0 0	0 1 1 1 0 0
3 4	0 1 1 0 0 0 0 1 1	0 0 1	1 0 1	0 0 0	0 0 0 0 0 1
3 5	0 0 0 0 0 0 0 0 1	1 1 0	0 0 0	0 0 0	0 1 1 1 1 0
3 6	0 0 0 0 0 0 0 0 1	0 1 1	0 0 0	0 0 0	0 1 1 1 1 1
3 7	0 0 0 0 0 0 0 1 1	0 1 0	0 0 1	0 0 0	0 0 0 1 1 0
4 0	0 0 0 0 0 0 0 0 0	0 0 1	0 0 0	0 0 0	0 0 0 0 0 1

4 1	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	0 0 0 0 1 1
4 2	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	0 0 0 1 1 1
4 3	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	0 0 1 1 1 0
4 4	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	0 1 0 1 1 0
4 5	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	0 1 1 0 0 1
4 6	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	0 1 1 1 0 1
4 7	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	1 1 0 0 0 1
5 0	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	1 1 0 0 1 1
5 1	0 0 1 1 0 0 0 1 1	0 0 0	0 0 1	0 0 0	1 1 0 1 1 1
5 2	0 0 0 0 0 0 0 1 1	0 0 1	0 0 1	0 0 0	0 0 0 0 0 1
5 3	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	1 0 1 1 0 1
5 4	0 0 0 0 0 0 0 1 1	0 0 0	0 0 0	0 0 0	1 0 1 1 0 0
5 5	0 0 0 0 0 0 0 0 1	1 1 0	0 0 0	0 0 0	1 0 1 1 1 0
5 6	0 0 0 0 0 0 0 0 1	0 1 0	0 0 0	0 0 0	1 0 1 1 1 1
5 7	0 0 0 0 0 0 0 1 1	1 1 0	1 1 0	1 1 0	1 1 0 0 0 0
6 0	0 0 0 0 0 0 0 0 1	1 1 0	0 0 0	0 0 0	1 1 0 1 1 0
6 1	0 0 0 0 0 0 0 0 1	1 1 0	0 0 0	0 0 0	1 1 0 0 1 0
6 2	0 0 0 0 0 0 0 0 1	0 0 1	0 0 0	0 0 0	0 0 0 0 0 1
6 3	0 0 0 0 0 0 0 0 1	1 1 0	0 0 0	0 0 0	1 1 0 1 0 0
6 4	0 0 0 0 0 0 0 0 1	0 1 0	0 0 0	0 0 0	1 1 0 1 0 1
6 5	0 0 0 0 1 0 0 1 1	0 0 1	1 0 1	0 0 0	0 0 0 0 0 1
6 6	0 0 0 0 0 1 1 0 1	0 0 0	1 0 1	0 0 0	0 0 0 0 0 1
6 7	0 0 1 0 0 0 0 1 1	0 0 0	0 0 0	0 0 0	1 1 1 0 0 0
7 0	0 0 0 0 0 0 0 1 1	0 0 1	1 0 0	0 0 0	0 0 0 0 0 1

5.4 机器指令（表 5-2）

表 5-2 机器指令

地址（十六进制）	内容（二进制）	助记符	说明
00	00000000	IN(R0)	INPUTDEVIECE→R0
01	00010000	ADDI[02H]	R0+[ ]→R0

02	00010001		
03	00100000	STA[37H]	R0→[37H]
04	00110111		
05	00110000	OUT[37H]	[37H]→LED
06	00110111		
07	01000000	JUMP	[09]→PC
08	00001001		
09	01010001	SUB[31H]	R0-[31H]→R1
0A	00110001		
0B	00100100	STA[38H]	R1→[38H]
0C	00111000		
0D	00110101	OUT[38H]	[38H]→LED
0E	00111000		
0F	01100010	AND[32H]	R0[32H]→R2
10	00110010		
11	00101000	STA[39]	R2→[39H]
12	00111001		
13	00111010	OUT[39H]	[39H]→LED
14	00111001		
15	01110000	LW[33H]	[39H]→R0
16	00110011		
17	00100000	STA[3AH]	R0→[3AH]
18	00111010		
19	00110000	OUT[3AH]	[3AH]→LED
1A	00111010		
1B	10000000	NOT[34H]	R0
1C	00110100		
1D	00100000	STA[3BH]	R0→[3BH]
1E	00111011		

1F	00110000	OUT[3BH]	[3BH]→LED
20	00111011		
21	10010000	RS	R0 shift right
22	00100000	STA[3CH]	R0→[3CH]
23	00111100		
24	00110101	OUT[3CH]	[3CH]→LED
25	00111100		
26	10100001	MOVER	R0→R1
27	00100100	STA[3DH]	R1→[3DH]
28	00111101		
29	00110101	OUT[3DH]	[3DH]→LED
2A	00111101		
2B	10110000	MOVEM[35-36H]	MOVEM1→MOVEM2
C	00110101		
D	00110110		
E	00110000	OUT[3EH]	[3EH]→LED
F	00111110		
30	11000000	STOP	
31	00000001		SUB 自定义数据
32	00000010		AND 自定义数据
33	00000011		LW 自定义数据
34	00000100		NOT 自定义数据
35	00000101		MOVEM 自定义数据
36			
37			ADDi 结果
38			SUB 结果
39			AND 结果
3A			LW 结果
3B			NOT 结果

3C			RS 结果
3D			MOVER 结果
3E			MOVEM 结果

表 5-2 机器指令表

### 5.5 系统性能分析

表 5-3 CPI 计算

	使用次数	单条指令 触发次数	总次数
IN	1	4	4
ADDi	1	6	6
STA	7	5	35
OUT	8	6	48
JMP	1	4	4
SUB	1	7	7
AND	1	7	7
LW	1	5	5
NOT	1	6	6
RS	1	5	5
MOVER	1	3	3
MOVEM	1	8	8
STOP	1	3	3
累计	26	69	141

静态 CPI=69/13=5.31

动态 CPI=141/26=5.42

控制存储器利用率=52/64=81.25%

## 心得体会

通过两周的课程设计，让我有了很大的收获。有学习方面的，也有思想方面的。

首先，学习方面，两周的课程设计让我对上学期所学的知识有了更加深刻的理解，同时使上学期做过的的算术逻辑运算实验、存储器实验、微程序控制器实验和基本模型机实验得到巩固，从而让自己的知识在质的方面得到了提升。本次课程设计，我们使用的是 TDN-CM++ 计算机组成原理实验教学系统，让我学会了如何正确的使用实验系统，懂得了它的基本的工作原理，比如说时序功能部件、指令译码电路、寄存器组控制、微地址形成部件的功能结构和工作机制等，同时还提高了我的动手能力，学会了怎么样把一个很复杂的实验引脚图用排线连接起来，把编好的微指令和机器指令输进去，并且能够正确地运行。

其次，就像老师在实验前说过的一样，这次课程设计将会是我们做过的收获最大的，的确，它使我们从思想上有的很大的转变。以前的课程设计，都是以应付老师、混个成绩的心态来做的，这次就不同了，刚开始大家都懒懒散散的，直到后来学会了如何进行一个团队工作，知道这不是自己一个人的事，对待事情的态度不同了，当然做出来的事情就也不同了。在这次试验中，大家喊得最多的就是烦了，因为一个小小的插线错误，就会导致整个线路的错误，使程序不能正常运行；或是一个小小的输入错误，把 0 输成 1 了，或者把 1 输成 0 了，都会产生很大的影响，不能调到期望的指令去，或是其他的错误，所以我们在做任何事情的时候都要认认真真的，不得有半点马虎。在实验中，我们遇到了很多很多的困难，有时候也想过放弃，要不要实现这条指令了，可是到最后还是要实现的，所以这次课程设计让我们学会了要勇敢地去克服困难，不能遇到阻碍就后退，否则就不会有大的收获。

这次的课程设计，让我受益匪浅，我会把从中得到的教训及经验运用到更多的地方去，让自己学到更多东西，收获更多，成长更多！

## 参考文献

- [1] Patterson, David A. & John L. Hennessy. 计算机组织与设计硬件/ 软件接口(英文版, 第二版) [M]. 北京:机械工业出版社, 1999.
- [2] 计算机组成原理实验指导书

# 计算机组成原理(名词解析)

## 第一章 概论

- 1、主机：主机中包含了除输入输出设备以外的所有电路部件，是一个能够独立工作的系统。
- 2、CPU：中央处理器，是计算机的核心部件，同运算器和控制器构成。
- 3、运算器：计算机中完成运算功能的部件，由 ALU 和寄存器构成。
- 4、ALU：算术逻辑运算单元，执行所有的算术运算和逻辑运算。
- 5、外围设备：计算机的输入输出设备，包括输入设备，输出设备和外存储设备。
- 6、数据：编码形式的各种信息，在计算机中作为程序的操作对象。
- 7、指令：是一种经过编码的操作命令，它指定需要进行的操作，支配计算机中的信息传递以及主机与输入输出设备之间的信息传递，是构成计算机软件的基本元素。
- 8、透明：在计算机中，从某个角度看不到的特性称该特性是透明的。
- 9、位：计算机中的一个二进制数据代码，计算机中数据的最小表示单位。
- 10、字：数据运算和存储的单位，其位数取决于具体的计算机。
- 11、字节：衡量数据量以及存储容量的基本单位。1 字节等于 8 位二进制信息。
- 12、字长：一个数据字中包含的位数，反应了计算机并行计算的能力。一般为 8 位、16 位、32 位或 64 位。
- 13、地址：给主存中不同的存储位置指定的一个二进制编号。
- 14、存储器：计算机中存储程序和数据部件，分为内存和外存。
- 15、总线：计算机中连接功能单元的公共线路，是一束信号线的集合，包括数据总线、地址总线和控制总线。
- 16、硬件：由物理元器件构成的系统，计算机硬件是一个能够执行指令的设备。
- 17、软件：由程序构成的系统，分为系统软件和应用软件。
- 18、兼容：计算机部件的通用性。
- 19、软件兼容：一个计算机系统上的软件能在另一个计算机系统上运行，并得到相同的结果，则称这两个计算机系统是软件兼容的。
- 20、程序：完成某种功能的指令序列。
- 21、寄存器：是运算器中若干个临时存放数据的部件，由触发器构成，用于存储最频繁使用的数据。
- 22、容量：是衡量容纳信息能力的指标。
- 23、主存：一般采用半导体存储器件实现，速度较高、成本高且当电源断开时存储器的内容会丢失。
- 24、辅存：一般通过输入输出部件连接到主存储器的外围设备，成本低，存储时间长。
- 25、操作系统：主要的系统软件，控制其它程序的运行，管理系统资源并且为用户提供操作界面。
- 26、汇编程序：将汇编语言程序翻译成机器语言程序的计算机软件。
- 27、汇编语言：采用文字方式（助记符）表示的程序设计语言，其中大部分指令和机器语言中的指令一一对应，但不能被计算机的硬件直接识别。
- 28、编译程序：将高级语言程序转换成机器语言程序的计算机软件。
- 29、解释程序：解释执行高级语言程序的计算机软件，解释并立即执行源程序的语句。
- 30、系统软件：计算机系统的一部分，进行命令解释、操作管理、系统维护、网络通信、软件开发和输入输出管理的软件，与具体的应用领域无关。
- 31、应用软件：完成应用功能的软件，专门为解决某个应用领域中的具体任务而编写。



32、指令流：在计算机的存储器与 CPU 之间形成的不断传递的指令序列。从存储器流向控制器。

33、数据流：在计算机的存储器与 CPU 之间形成的不断传递的数据序列。存在于运算器与存储器以及输入输出设备之间。

## 第二章 数据编码和数据运算

1、原码：带符号数据表示方法之一，一个符号位表示数据的正负，0 代表正号，1 代表负号，其余的代表数据的绝对值。

2、补码：带符号数据表示方法之一，正数的补码与原码相同，负数的补码是将二进制位按位取反后在最低位上加 1。

3、反码：带符号数据的表示方法之一，正数的反码与原码相同，负数的反码是将二进制位按位取反。

4、移码：带符号数据表示方法之一，符号位用 1 表示正，0 表示负，其余位与补码相同。

5、阶码：在浮点数据编码中，表示小数点的位置的代码。

6、尾数：在浮点数据编码中，表示数据有效值的代码。

7、基数：在浮点数据编码中，对阶码所代表的指数值的数据，在计算机中是一个常数，不用代码表示。

8、机器零：在浮点数据编码中，阶码和尾数都全为 0 时代表的 0 值。

9、上溢：指数的绝对值太大，以至大于数据编码所能表示的数据范围。

10、下溢：指数的绝对值太小，以至小于数据编码所能表示的数据范围。

11、规格化数：在浮点数据编码中，为使浮点数具有唯一的表示方式所作的规定，规定尾数部分用纯小数形式给出，而且尾数的绝对值应大于  $1/R$ ，即小数点后的第一位不为零。

12、Booth 算法：一种带符号数乘法，它采用相加和相减的操作计算补码数据的乘积。

13、海明距离：在信息编码中，两个合法代码对应位上编码不同的位数。

14、冯·诺依曼舍入法：浮点数据的一种舍入方法，在截去多余位时，将剩下数据的最低位置 1。

15、检错码：能够发现某些错误或具有自动纠错能力的数据编码。

16、纠错码：能够发现某些错误并且具有自动纠错能力的数据编码。

17、海明码：一种常见的纠错码，能检测出两位错误，并能纠正一位错误。

18、循环码：一种纠错码，其合法码字移动任意位后的结果仍然是一个合法码字。

## 第三章 存储系统

1、RAM：随机访问存储器，能够快速方便的访问地址中的内容，访问的速度与存储位置无关。

2、ROM：只读存储器，一种只能读取数据不能写入数据的存储器。

3、SRAM：静态随机访问存储器，采用双稳态电路存储信息。

4、DRAM：动态随机访问存储器，利用电容电荷存储信息。

5、EDO DRAM：增强数据输出动态随机访问存储，采用快速页面访问模式并增加了一个数据锁存器以提高数据传输速率。

6、PROM：可编程的 ROM，可以被用户编程一次。

7、EPROM：可擦写可编程的 ROM，可以被用户编程多次。靠紫外线激发浮置栅上的电荷

以达到擦除的目的。

8、 **EEPROM**：电可擦写可编程的 ROM，能够用电子的方法擦除其中的内容。

9、 **SDRAM**：同步型动态随机访问存储器，在系统时钟控制下进行数据的读写。

10、 **快闪存储器**：一种非挥发性存储器，与 **EEPROM** 类似，能够用电子的方法擦除其中的内容。

11、 **相联存储器**：一种按内容访问的存储器，每个存储单元有匹配电路，可用于在 cache 中查找数据。

12、 **多体交叉存储器**：由多个相互独立、容量相同的存储体构成的存储器，每个存储体独立工作，读写操作重叠进行。

13、 **访存局部性**：CPU 的一种特性，对存储空间的 90% 的访问局限于存储空间中的 10% 的区域中，而另外 10% 的访问则分布在 90% 的区域中。

14、 **直接映象**：cache 的一种地址映象方式，一个主存块只能映象到 cache 中的唯一一个指定块。

15、 **全相联映象**：cache 的一种地址映象方式，一个主存块可映象到任何 cache 块。

16、 **组相联映象**：cache 的一种地址映象方式，将存储空间分成若干组，各组之间用直接映象，组内各块之间用全相联映象。

17、 **全写法（写直达法）**：cache 命中时的一种更新策略，写操作时将数据既写入 cache 又写入主存，但块更替时不需要将调出的块写回主存。

18、 **写回法**：cache 命中时的一种更新策略，写 cache 时不写主存，而当 cache 数据被替换出去时才写回主存。

19、 **按写分配**：cache 不命中时的一种更新策略，写操作时把对应的数据块从主存调入 cache。

20、 **不按写分配**：cache 不命中时的一种更新策略，写操作时该地址的数据块不从主存调入 cache。

一般写回法采用按写分配法，写直达法则采用不按写分配法。

21、 **虚拟存储器**：在内存与外存间建立的层次体系，使得程序能够像访问主存储器一样访问外部存储器，主要用于解决计算机中主存储器的容量问题。

22、 **层次化存储体系**：把各种不同存储容量、不同访问速度、不同成本的存储器件按层次构成多层的存储器，并通过软硬件的管理将其组成统一的整体，使所存储的程序和数据按层次分布在各种存储器件中。

23、 **访问时间**：从启动访问存储器操作到操作完成的时间。

24、 **访问周期时间**：从一次访问存储的操作到操作完成后可启动下一次操作的时间。

25、 **带宽**：存储器在连续访问时的数据吞吐率。

26、 **段式管理**：一种虚拟存储器的管理方式，把虚拟存储空间分成段，段的长度可以任意设定，并可以放大或缩小。

27、 **页式管理**：一种虚拟存储器的管理方式，把虚拟存储空间和实际存储空间等分成固定容量的页，需要装入内存，各页可装入主存中不同的实际页面位置。

28、 **段页式管理**：一种虚拟存储器的管理方式，将存储空间逻辑模块分成段，每段又分成若干页。

29、 **块表**：主存-cache 地址映像机制，由查块表判定主存地址的存储单元是否在 cache 中以及在 cache 中的位置。

30、 **页表**：页式虚拟存储器管理用的地址映象表，其中包括每个页的主存页号、装入位和访问方式等。

31、 **段表**：段式虚拟存储器管理用的地址映象表，其中包括每个段的基地址、段长、装入位和访问方式等。

32、 固件：固化在硬件中的固定不变的常用软件。

#### 第四章 指令系统

- 1、 指令系统:计算机中各种指令的集合
- 2、 计算机指令:计算机硬件能识别并能直接执行操作的命令,描述一个基本操作.
- 3、 指令编码:将指令分成操作码和操作数地址码的几个字段来编码.
- 4、 指令格式:指定指令字段的个数,字段编码的位数和编码的方式.
- 5、 立即数:在指令中直接给出的操作数
- 6、 指令字长度:一个指令字所占有的位数.
- 7、 助记符:用容易记忆的符号来表示指令中的操作码和操作数.
- 8、 汇编语言:采用文字方式（助记符）表示的程序设计语言，其中大部分指令和机器语言中的指令一一对应，但是不能被计算机的硬件直接识别。
- 9、 伪指令:汇编语言程序所提供的装入内存中的位置信息,表示程序段和数据段开始信息及结束信息等.且不转换成 2 进制机器指令.
- 10、 小数端:将最低字节存到小地址位置.
- 11、 堆栈:数据的写入写出不需要地址,按先进后出的顺序读取数据的存储区.
- 12、 操作数寻址方式:指令中地址码的内容及编码方式.
- 13、 系统指令:改变计算机系统的工作状态的指令.
- 14、 特权指令:改变执行特权的指令,用于操作系统对系统资源的控制.
- 15、 自陷指令:特殊的处理程序,又叫中断指令.
- 16、 寻址方式：对指令的地址码进行编码，以得到操作数在存储器中的地址的方式。
- 17、 相对转移：一种形成转移目标地址的方式，转移指令的目标指令地址是由 PC 寄存器的值加上一个偏移量形成的。
- 18、 绝对转移：一种形成转移目标地址的方式，转移指令的目标指令地址是由有效地址直接指定，与 PC 寄存器的内容无关。
- 19、 无条件转移：一种转移指令类型，不管状态如何，一律进行转移操作。
- 20、 条件转移：一种转移指令类型，根据计算机中的状态决定是否转移。

#### 第五章 控制器

- 1、 指令周期：从一条指令的启动到下一条指令的启动的间隔时间。
- 2、 机器周期：指令执行中每一步操作所需的时间。
- 3、 指令仿真：通过改变微程序实现不同机器指令系统的方式，使得在一种计算机上可以运行另一种计算机上的指令代码。

- 4、指令模拟：在一种计算机上用软件来解释执行另一种计算机的指令。
- 5、硬连线逻辑：一种控制器逻辑，用一个时序电路产生时间控制信号，采用组合逻辑电路实现各种控制功能。
- 6、微程序：存储在控制存储中的完成指令功能的程序，由微指令组成。
- 7、微指令：控制器存储的控制代码，分为操作控制部分和顺序控制部分。
- 8、微操作：在微程序控制器中，执行部件接受微指令后所进行的操作。
- 9、微地址：微每时令在控制存储器中的存储地址。
- 10、水平型微指令：一次能定义并执行多个并行操作控制信号微指令。
- 11、型微指令：一种微指令类型，设置微操作码字段，采用微操作码编码法，由微操作码规定微指令的功能。
- 12、控制存储器：微程序型控制器中存储微指令的存储器，通常是 ROM。
- 13、控制存储器的容量取决于指令的数量，每条指令的微程序长度和微指令代码的利用率。

## 第六章 系统总线

- 1、猝发传输方式：在一个总线周期内传输存储地址连续的多个数据字的总线传输方式。
- 2、四边沿协议（全互锁）：全互锁的总线通信异步方式，就绪信号和应答信号的上升边沿和下降边沿都是触发边沿。
- 3、码元：信息传输通道中，携带数据信息的信号单元。
- 4、波特率：码元传输速率，每秒通过信道传输的码元数。（传的是信号）
- 5、比特率：信息位传输速率，每秒钟通过信道传输的有效信息量。（传的是信息）
- 6、UART：通用异步接收器/发送器，一种典型的集成电路异步串行接口电路。
- 7、主设备：获得总线控制权的设备
- 8、从设备：被主设备访问的设备。
- 9、总线事务：从总线的请求到完成总线的使用的操作序列。
- 10、总线协议：总线通信同步方式规则，规定实现总线数据传输的定时规则。
- 11、链式查询方式（菊花链方式）：各申请总线的设备合用一条总线作为请求信号线，布总线控制设备的响应信号线则串接在各设备间。
- 12、计数器定时查询方式：集中式总线裁决方式之一，设备要求使用总线时通过一条公用请求线发出，总线控制器按计数的值对各设备进行查询。
- 13、独立请求方式：集中式总线裁决方式之一，每一个设备都有一个独立的总线请求信号线送到总线控制器，控制器也给各设备分别发送一个总线响应信号。
- 14、系统总线：处理器总线，连接处理器和主存，是计算机系统的主干。
- 15、数据帧：串行数据传输的位格式，包括起始位，数据位，校验位，结束位和空闲位。
- 16、串行传输：是指数据的传输在一条线路上按位进行。（只需一条数据传输线，线路的成本低，适合于长距离的数据传输）

17、 并行传输：每个数据位都需要单独一条传输线，所有的数据位同时进行传输。（在采用并行传输方式的总线中，除了有传输数据的线路外，还可以具有传输地址和控制信号的线路，地址线用于选择存储单元和设备，控制线用于传递操作信号）

18、 复合传输：又称总线复用的传输方式，它使不同的信号在同一条信号线上传输，不同的信号在不同的时间片中轮流地身总线的同一条信号线上发出。（它与并串传输的区别在于分时地传输同一数据源的不同信息。）

19、 消息传输方式：总线的信息传输方式之一，将总线需要传送的数据信息、地址信息、和控制信息等组合成一个固定的数据结构以猝发方式进行传输。

## 第七章 输入输出系统

- 1、 统一编址：就是将外围设备中的控制寄存器、数据寄存器、状态寄存器和内存单元一起编排地址。
- 2、 单独编址：采用内存地址空间和外围设备的地址空间分开编排地址。
- 3、 中断：是一种在发生了一个外部的的事件时调用相应的处理程序的过程。
- 4、 多级中断：计算机在处理中断的过程中也能够禁止某些外部中断请求而允许另一些中断请求。
- 5、 中断屏蔽：在处理中断时阻止其他中断的过程。
- 6、 DMA：称为直接存储器访问，是指数据在主存与外设之间传输，不需要由 CPU 来对每个字的传输进行控制。
- 7、 现场保护：将程序计数器 PC 中以及有关的寄存器的内容存储起来，使中断结束后能恢复到原来的状态继续执行程序的过程。
- 8、 中断向量方式：是指发生中断请求的外围设备通过输入输出总线主动向 CPU 发出一个识别代码，CPU 根据这个代码就可以识别出请求中断的外围设备的中断方式。
- 9、 自陷：由 CPU 内部因素引起的中断（内部中断）。
- 10、 软件中断：由专门的指令引起的中断。
- 11、 通道：是一个具有输入输出处理器的输入输出接口。
- 12、 通道命令：通道自己专门用来处理输入输出事务的指令。

## 第八章 外围设备

- 1、 CRT：阴极射线管，显示器的一种。
- 2、 LCD：液晶显示器。
- 3、 CCD：电荷耦合器件，用于图像输入。
- 4、 MIDI：音乐器材数字化接口，规定电子乐器与计算机之间传递信息的方式。
- 5、 分辨率：衡量显示器显示清晰度的指标，以像素的个数为标志。
- 6、 灰度级：显示器所显示的像素点的亮度差别。
- 7、 归零制 RZ：一种磁盘信息记录方式，正脉冲表示 1，负脉冲表示 0，在记录下一个信息之前记录电流要恢复到零电流。
- 8、 不归零制 NRZ0：一种磁盘信息记录方式，磁头线圈上始终有电流，正向电流代表 1，反向电流代表 0。
- 9、 调相制 PM：一种磁盘信息记录方式，在一个磁化元的中间位置，利用电流相位的变化进行写 1 或写 0。
- 10、 调频制 FM：一种磁盘信息记录方式，写 1 电流的变化频率是写 0 电流频率的 2 倍。
- 11、 RLL 码：游程长度受限码，将原始数据序列变换成 0、1 游程长度受限的代码。
- 12、 寻道时间：在磁盘中，将磁头定位到所要求的磁道上所需要的时间。

- 13、 温彻斯特技术：硬盘中采用的一种技术，将磁头、盘片和音圈电机组合在一个密封的盒内，避免产生磁头与介质的磨损，并且采用接触式启停。
- 14、 气泡式喷墨打印机：一种非击打式打印设备，喷头通过电加热，使墨水在蒸气的作用下从喷头射到纸上。
- 15、 绘图机：计算机图形输出设备，主要用于工程图纸的输出。
- 16、 数字化仪：一种二维坐标的输入系统，主要用于输入工程图，包括一个游标和一个图形板。
- 17、 触摸屏：一种具有触摸式输入功能的显示屏或者附加显示屏上的输入设备，用于输入屏幕位置信息，通常与屏幕菜单配合使用。
- 18、 扫描仪：一种图像输入设备，主要用于各类计算机静态图像的输入。
- 19、 音频识别：一个对于音频信息提炼和压缩的过程，如将语音信号转化成文字信息以便于计算机的存储和处理。
- 20、 音频合成：使计算机能够朗读文本或者演奏出音乐的过程，如将文字信息转化成语音信息，或者将 MIDI 数据文件转化成音乐信号。
- 21、 音效处理：改进音频设备输出效果的过程，分为三种类型：混响和延时处理；声音的回放效果处理；环绕声的处理。
- 22、 **CD-ROM**：计算机中只读型光盘的主要标准。
- 23、 **WORM**：写一次读多次型光盘，可由用户一次性写入信息，写入后可以反复读取。