

高职高专

现代信息技术系列教材

微型计算机原理 学习辅导与习题解析

洪小叶 刘新元 编



人民邮电出版社
POSTS & TELECOM PRESS

高 职 高 专 现 代 信 息 技 术 系 列 教 材

微 型 计 算 机 原 理 学 习 辅 导 与 习 题 解 析

洪小叶 刘新元 编

人 民 邮 电 出 版 社

图书在版编目(CIP)数据

微型计算机原理学习辅导与习题解析 / 洪小叶, 刘新元编. —北京: 人民邮电出版社, 2005.2
(高职高专现代信息技术系列教材)

ISBN 7-115-12989-4

I. 微... II. ①洪...②刘... III. 微型计算机—理论—高等学校: 技术学校—教学参考资料
IV. TP360.1

中国版本图书馆 CIP 数据核字 (2005) 第 007886 号

内 容 提 要

本书是《微型计算机原理》的配套学习辅导书, 内容包括每章的基本概念及知识点, 重点、难点解析和习题、答案与解析。本书对每章内容在归纳基本概念、知识点的基础上着重围绕其中的重点、难点作大量的示例分析, 并在每章的后面针对相关内容给出了多种类型的课外习题并给出答案及解析。全书共分 10 章, 分别介绍微型计算机的基础知识、80x86 微处理器、80x86 汇编语言指令系统及源程序的基本语法、汇编语言程序设计、存储器系统、输入输出接口技术的基本概念、中断系统与定时计数、微型计算机系统的串行/并行通信技术、数模与模数转换接口技术和总线技术与接口标准。

本书既可配合《微型计算机原理》使用, 也可作为学习微型计算机原理的辅导资料单独使用。本书适用于高职高专院校计算机及相关专业, 也可作为中等职业学校相关专业的教材。

高职高专现代信息技术系列教材

微型计算机原理学习辅导与习题解析

- ◆ 编 洪小叶 刘新元
责任编辑 赵慧君
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
读者热线: 010-67129259
北京朝阳展望印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 11.5
字数: 278 千字 2005 年 2 月第 1 版
印数: 1—4 000 册 2005 年 2 月北京第 1 次印刷

ISBN 7-115-12989-4/TP · 4388

定价: 16.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

编者的话

在信息技术产业深入发展、广泛应用的今天,围绕着高等职业教育的蓬勃发展,培养具有扎实的计算机软硬件基础知识、掌握微型计算机组成原理与开发应用专业技能的高素质应用型人才,已成为当务之急,我们依据多年电子信息及计算机技术教学、科研工作的积累编写了本书。

本书内容共分10章。每章的编写构架均为两部分:第一部分,基本概念知识点及示例分析;第二部分,课外习题与答案解析。具体内容为:第1章涉及计算机层次结构、特点、分类及主要技术指标,计算机中所用数据的进位计数制及相关概念。第2章涉及80x86微处理器的基本外部、内部结构,工作过程与原理,微处理器内部可编程序寄存器组及存储器组织管理。第3章80x86汇编语言指令系统与寻址方式、汇编语言源程序的基本语法及上机过程。第4章计算机程序设计的基本结构以及用汇编语言程序设计实现的基本算法。第5章通用PC机的存储体系结构、半导体存储器的分类及主要性能参数、使用半导体存储芯片构建存储体系及与CPU系统的连接。第6章微型计算机输入输出(I/O)系统的基本原理组成与基本功能,CPU系统与接口之间的I/O传送控制方式。第7章中断系统的相关基本概念,定时计数技术的相关基本概念。第8章并行传输与串行通信的基本概念,DMA方式的相关基本知识。第9章数模与模数转换接口概念及常用电路、数模与模数转换器与CPU系统的接口设计。第10章总线技术概述,常见总线及接口标准概要。第7、8、9章为刘新元老师编写,其他章节为洪小叶老师编写。

在今天,信息技术产业新技术、新产品层出不穷不断更新,但从产品到教学毕竟还有一段漫长的路。我们知道,上述内容在实际应用中有的已被新的产品与技术取代了,但技术的发展有其承袭性、连续性,其基本概念、基本方法在一段时期仍然起主导作用;同时,与上述内容相对应的教学实验环境是很成熟的,有利于学生的教学实践。基于这些考虑,在编写此书时我们注重内容的基本性和应用性。一般来说,基本概念是最简单的,但也是最关键的。所以我们力求对基本概念的叙述准确简捷,并以示例辅之。鉴于本书涉及的专业面具有很强的应用性、实践性,在编写示例与习题时我们力求贴近实际应用,注重分析问题、解决问题。我们以本书能给读者有所帮助为最大安慰。

在本书编写过程中,我们得到了北京大学谢柏青教授的悉心指导、许多同行老师的热心帮助,中央民族大学物理与工程学院00级、01级的部分同学参与了示例、习题的整理调试工作,在此向他们表示衷心的感谢。在编写过程中我们广泛参考查阅了相关教材资料,对此我们尽可能的列入了书后的“参考文献”中,万一有所遗漏,敬请作者原谅。

由于水平有限,书中难免存在不当之处,真诚欢迎读者批评、指正。

编者

2004年10月

目 录

第1章 微型计算机原理的基础知识	1
1.1 基本概念及知识点	1
1.2 重点、难点解析	3
1.3 习题、答案与解析	7
1.3.1 习题	7
1.3.2 答案与解析	10
第2章 80x86 微处理器	13
2.1 基本概念及知识点	13
2.2 重点、难点解析	17
2.3 习题、答案与解析	19
2.3.1 习题	19
2.3.2 答案与解析	21
第3章 80x86 汇编语言指令系统及源程序的基本语法	24
3.1 基本概念及知识点	24
3.2 重点、难点解析	37
3.3 习题、答案与解析	53
3.3.1 习题	53
3.3.2 答案与解析	60
第4章 汇编语言程序设计	66
4.1 基本概念及知识点	66
4.2 重点、难点解析	67
4.3 习题、答案与解析	89
4.3.1 习题	89
4.3.2 答案与解析	90
第5章 存储器系统	99
5.1 基本概念及知识点	99
5.2 重点、难点解析	102
5.3 习题、答案与解析	107
5.3.1 习题	107
5.3.2 答案与解析	109

第 6 章 输入输出接口技术	111
6.1 基本概念及知识点	111
6.2 重点、难点解析	114
6.3 习题、答案与解析	121
6.3.1 习题	121
6.3.2 答案与解析	124
第 7 章 中断系统与定时计数	128
7.1 基本概念及知识点	128
7.2 重点、难点解析	129
7.3 习题、答案与解析	135
7.3.1 习题	135
7.3.2 答案与解析	136
第 8 章 微型计算机系统的串行/并行通信技术	140
8.1 基本概念及知识点	140
8.2 重点、难点解析	141
8.3 习题、答案与解析	146
8.3.1 习题	146
8.3.2 答案与解析	147
第 9 章 数模与模数转换接口技术	152
9.1 基本概念及知识点	152
9.2 重点、难点解析	153
9.3 习题、答案与解析	164
9.3.1 习题	164
9.3.2 答案与解析	165
第 10 章 总线技术与接口标准	167
10.1 基本概念及知识点	167
10.2 习题、答案与解析	173
10.2.1 习题	173
10.2.2 答案与解析	175
参考文献	177

第 1 章 微型计算机原理的基础知识

本章主要内容:

- (1) 微型计算机的层次结构;
- (2) 微型计算机的特点、分类及主要技术指标;
- (3) 计算机中常用数据的进位计数制;
- (4) 常用进位计数制之间的转换;
- (5) 计算机中常用的基本数值编码 (BCD 码)、非数值数据编码 (ASCII)。

1.1 基本概念及知识点

1. 微型计算机原理结构

现在的微型计算机原理结构是冯·诺依曼结构, 如图 1.1 所示。图中主要部件介绍如下。

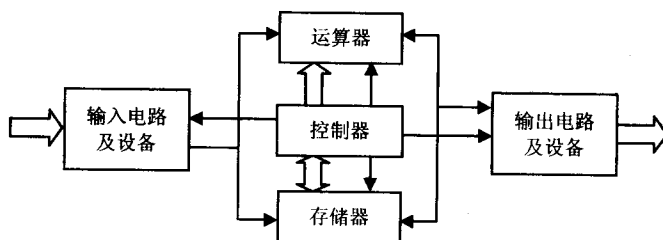


图 1.1

① 运算器: 计算机的运算场所。主要功能是实现算术运算和逻辑运算。

② 控制器: 按照当前的计算机指令代码, 向计算机各个部件发控制命令信号, 完成某种操作。

③ 存储器: 计算机的记忆部件, 存放数据和程序指令代码。

④ 输入、输出电路及设备: 传递计算机与外部信息的通道。

2. 微型计算机的层次结构

微型计算机的层次结构如图 1.2 所示。

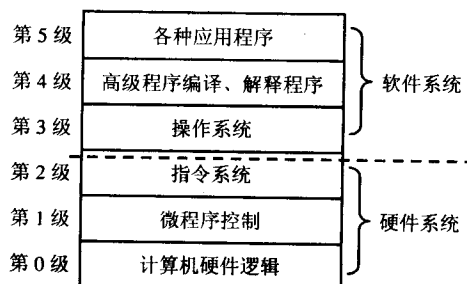


图 1.2

3. 微型计算机的主要技术指标

微型计算机的主要技术指标如下。

① 主频：CPU 的工作时钟频率，反映了 CPU 运行速度的快慢。

② 字长：CPU 内部运算器能同时处理二进制数据的位数，也是 CPU 内部数据线的位数，决定了 CPU 一次可以处理的二进制代码的能力。

③ 寻址空间与内存容量：CPU 系统地址线的位数决定了该 CPU 能够寻址存储空间的最大范围，CPU 能够寻址存储空间的最大范围为 2^n ， n 是 CPU 系统地址线的位数。

④ 总线标准：微型计算机主机内部的传输总线配置直接影响到微型计算机整体工作速度以及与外部设备交换信息的能力。

4. 计算机中常用的进位计数制

计算机中常用的进位计数制有二进制 (Binary)、八进制 (Octal)、十进制 (Decimal) 和十六进制 (Hexadecimal)。

5. 任意进位计数制数表示通式

任意进位计数制的任一数总可以表示为以下展开式：

$$N = \sum_{i=-m}^n a_i \times r^i = a_n \times r^n + a_{n-1} \times r^{n-1} + \cdots + a_1 \times r^1 + a_0 \times r^0 + \cdots + a_{-m} \times r^{-m}$$

其中， r 为进制基数、 r^i 为数据中第 i 位的权值、 a_i 为数据中第 i 位的系数 ($0 \sim r-1$)、 m 为小数点右边的位数、 n 为小数点左边的位数 (整数部分位数)。

6. 位权

对应某一进位计数制第 i 位上基数的 i 次幂叫做该计数制第 i 位的位权 (权值)。

7. 常用进位计数制之间的转换

二进制与八进制之间的转换：3 位二进制数转为 1 位八进制数。

二进制与十六进制之间的转换：4 位二进制数转为 1 位十六进制数。

其他进制 (二、八、十六进制) 转换为十进制：按权值展开。

十进制转换为其他进制 (二、八、十六进制)：乘除法 (整数除基取余、小数乘基取整)，降幂法。

8. 计算机基本的数据种类与性质

无符号数：组成数据的所有二进制数位都是数值位。无符号数的数据范围： $0 \leq x \leq 2^n - 1$ (n ：二进制数位)。

有符号数：组成数据的所有二进制数位中最高有效位作为符号位，其余数位是数值位。当最高有效位=0，表示此数为正数；反之，表示此数为负数。带符号数的数据范围： $-2^{(n-1)} \leq x \leq +2^{(n-1)} - 1$ (n ：二进制数位)。

9. 机器数

计算机把有符号数的符号位与数值位合起来作为一个数处理，这个数叫做机器数。机器数的编码形式有原码 $[X]_{\text{原}}$ 、反码 $[X]_{\text{反}}$ 和补码 $[X]_{\text{补}}$ 。

10. 定点数

无论是正数还是负数，计算机对整数或纯小数按定点数处理，对整数按数据的小数点位置固定在数据最低有效位的后面处理；对纯小数按数据的小数点位置固定在数据最高有效位

的前面处理。

11. 浮点数

对于数据中既有整数部分又有小数部分的实数计算机用浮点数表示, 即一个有效系数与一个基数的某次幂相乘的形式: $X = S \times b^J$

12. 计算机对有符号数的初级运算规则

设 X 、 Y 为有符号数, 则计算机对有符号数的初级运算规则如下。

$$X+Y: [X]_{补}+[Y]_{补}=[Z]_{补}$$

$$X-Y: [X]_{补}+[-Y]_{补}=[Z]_{补}$$

$$[[X]_{补}]_{补}=[X]_{原}$$

13. BCD码

BCD 码是计算机中表示数值数据的一种编码形式, 它用 4 位二进制编码表示 1 位十进制数。

14. ASCII

ASCII 是在计算机中用 7 位二进制编码表示西文字母、符号及基本控制符 (如清除屏幕、退格、回车、换行) 等的代码。

15. 位 (bit)

位是计算机中表示信息的最基本量化单位, 表示一个二进制数位。n 位二进制数码从全 0 到全 1 最多可以表示 2^n 个数。

16. 字节 (byte)

8 位二进制数位为一个字节。字节是计算机中最基本的表示信息长度的单位, 以此为基础, 在偏重硬件叙述的微型计算机原理教学内容中, 两个字节连起来叫做一个字 (word); 4 个字节连起来叫做一个双字 (double word)。

1.2 重点、难点解析

本章所涉及的内容, 前半部分为微型计算机的最基本概念与常识; 后半部分是计算机技术中的数制、码制问题。这些内容是理解掌握微型计算机工作原理及微型计算机应用的基础, 本书后面各章的内容均是在此基础上的展开与深入。其中, 数制、码制中各种进制之间的转换、计算机基本数据种类与性质等内容在教学中往往容易成为教师讲解和学生掌握的“瓶颈”, 下面用一些示例进行说明。

示例 1 将下面二进制序列转换为相应的八进制、十六进制数。

$$1001011110010001110.101011101001 = ()_8 = ()_{16}$$

转换为八进制数: 从小数点向左、右各 3 位二进制数为一组, 位数不够补 0。

$$\begin{array}{cccccccccccc} 00 & 1' & 001' & 011' & 110' & 010' & 001' & 110. & 101' & 011' & 101' & 001 \\ 1 & 1 & 3 & 6 & 2 & 1 & 6. & 5 & 3 & 5 & 1 \end{array}$$

结果: $(1001011110010001110.101011101001)_2 = (1136216.5351)_8$

转换为十六进制数: 从小数点向左、右各 4 位二进制数为一组, 位数不够补 0。

$$0 \ 100' \ 1011' \ 1100' \ 1000' \ 1110. \ 1010' \ 1110' \ 1001$$

4 B C 8 E A E 9

结果: $(1001011110010001110.101011101001)_2 = (4BC8E.AE9)_{16}$

示例 2 将下面八进制数转换为二进制数。

$(6513.724)_8 = (\quad)_2$

从小数点分别向左、右, 将每位八进制数展开成 3 位二进制数。

6 5 1 3 . 7 2 4

转换得到的二进制数: 110 101 001 011 . 111 010 100

结果: $(6513.724)_8 = (110101001011.1110101)_2$

示例 3 将下面十六进制数转换为二进制数。

$(1F829.AB6)_{16} = (\quad)_2$

从小数点分别向左、右, 将每位十六进制数展开成 4 位二进制数。

1 F 8 2 9 . A B 6

转换得到的二进制数: 0001 1111 1000 0010 1001 . 1010 1011 0110

结果: $(1F829.AB6)_{16} = (11111100000101001.10101011011)_2$

将其他进制(二、八、十六进制)数转换为十进制数的方法是, 将任意进位计数制的数按任意进位计数制数表示通式展开, 按十进制运算规则进行运算, 所得结果即为相应的十进制数。

示例 4 二、八、十六进制数分别转换为十进制数。

$$(1001101.011)_2 = 1 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3} \\ = (77.375)_{10}$$

$$(246.13)_8 = 2 \times 8^2 + 4 \times 8^1 + 6 \times 8^0 + 1 \times 8^{-1} + 3 \times 8^{-2} = (166.171875)_{10}$$

$$(2AB.7C)_{16} = 2 \times 16^2 + 10 \times 16^1 + 11 \times 16^0 + 7 \times 16^{-1} + 12 \times 16^{-2} = (683.484375)_{10}$$

示例 5 将十进制数 735.824 转换为相应的二进制数。

整数除 2 取余

余数

2) 735..... 1 ← 最低有效位

2) 367..... 1

2) 183..... 1

2) 91..... 1

2) 40..... 0

2) 20..... 0

2) 10..... 0

2) 5..... 1

2) 2..... 0

2) 1..... 1 ← 最高有效位

小数乘 2 取整

乘积整数

0.824 × 2 = 1.648..... 1 ← 小数部分高位

0.648 × 2 = 1.296..... 1

0.296 × 2 = 0.592..... 0

0.592 × 2 = 1.184..... 1

0.184 × 2 = 0.368..... 0

0.368 × 2 = 0.736..... 0 ← 小数部分低位

近似到小数点后 6 位

转换结果: $(735.824)_{10} = (1010001111.110100)_2$

有了二进制数的转换结果, 得到相应的十六进制数就是非常方便容易的事了, 将二进制

结果4位为一组得 $(28F.D)_{16}$ 。

示例6 用降幂法把十进制数10000转换为十六进制数。

列出十六进制数第4~第0位的权值：

16的4次方	16的3次方	16的2次方	16的1次方	16的0次方
65536	4096	256	16	0

从上表可见，最接近于且小于10000的十六进制数权值是16的3次方4096，并要求出10000里含有几个整数倍的4096。显然， $(3 \times 4096) > 10000 > (2 \times 4096)$ ，故在权值 16^3 的数位上转换后的系数应是2。此位系数确定后，将 2×4096 从10000中减去： $10000 - 8192 = 1808$ 。

对1808继续按上述步骤转换，最接近于且小于1808的十六进制权值是16的2次方256，并要求出1808里含有几个整数倍的256。取 $1808 \div 256$ 商的整数7，即在权值 16^2 的数位上转换后的系数应是7。然后做： $1808 - 7 \times 256 = 16$ 。

16恰好是 16^1 数位的权值，即 $1 \times 16 = 16$ ，在权值 16^1 的数位上转换后的系数应是1。

综上所述可得：

数位权值	16^3	16^2	16^1	16^0
系数	2	7	1	0

转换结果： $(10000)_{10} = (2710)_{16}$

示例7 用 n 位二进制数所能表示的无符号数范围是 $0 \sim 2^n - 1$ ；用 n 位二进制数所能表示的有符号数范围是 $-2^{n-1} \sim +2^{n-1} - 1$ 。

8位二进制数所能表示的无符号数范围： $0 \sim 255$ ；16位二进制数所能表示的无符号数范围： $0 \sim 65535$ 。

8位二进制数所能表示的有符号数范围： $-128 \sim +127$ ；16位二进制数所能表示的有符号数范围： $-32768 \sim +32767$ 。

示例8 已知 X 为有符号数，求 $[X]_{原}$ 、 $[X]_{反}$ 和 $[X]_{补}$ 。

设： $X = (+100)_{10}$ ， $(+100)_{10} = (+01100100)_2$ ，因为是正数，所以符号位为0，且正数的原码、反码及补码编码形式相同，所以： $[+100]_{原} = [+100]_{反} = [+100]_{补} = 0\ 1100100$ 。

设： $X = (-100)_{10}$ ，因为是负数，符号位为1，所以有

$[X]_{原} = [-100]_{原} = 1\ 1100100$ ， $[X]_{反} = [-100]_{反} = 1\ 0011011$ ， $[X]_{补} = [-100]_{补} = 1\ 0011011 + 1 = 1\ 0011100$

设： $X = (-245)_{10}$ 。从前面介绍可知： -245 已超出8位二进制所能表示的有符号数范围，所以 -245 的原码、反码及补码要用16位二进制表示。有

$[-245]_{原} = 1\ 000000011110101$

$[-245]_{反} = 1\ 111111100001010$

$[-245]_{补} = 1\ 111111100001010 + 1 = 1\ 111111100001011$

示例9 对有符号数进行运算时往往需要将有符号数在位数上进行扩展，扩展部分是原有符号数符号位向高数位的延伸，扩展的位数与原数位相同。

用16位二进制数表示 $(+100)_{10}$ 的补码： $00000000\ 0\ 1100100$ ；在此，高8位的0就是对用8位二进制数表示的 $(+100)_{10}$ 补码的扩展。

用16位二进制数表示 $(-100)_{10}$ 的补码： $11111111\ 1\ 1100100$ ；在此，高8位的1就是对用8位二进制数表示的 $(-100)_{10}$ 补码的扩展。

示例 10 用补码实现下列运算。

① $105+(-47)$

$(+105)_{10}=(01101001)_2$, 因为 105 是正数, 所以: $[+105]_{\text{补}}=01101001$;

$[-47]_{\text{原}}=10101111$, $[-47]_{\text{补}}=11010001$;

$[+105]_{\text{补}}+[-47]_{\text{补}}$:

$$\begin{array}{r} 01101001 \\ +) 11010001 \\ \hline 00111010 \end{array}$$

结果的符号位为 0, 所以结果为正数; 所以: $105+(-47)=00111010=(3A)_{16}=(58)_{10}$ 。

② $-105+47$

$[-105]_{\text{补}}+[+47]_{\text{补}}$

$[-105]_{\text{补}}=10010111$, $[+47]_{\text{补}}=00101111$

$$\begin{array}{r} 10010111 \\ + 00101111 \\ \hline 11000110 \end{array}$$

结果的符号位为 1, 所以结果为负数的补码, 对结果再求补得到结果的原码, 进一步得到结果的真值: $[11000110]_{\text{补}}=10111001+1=10111010$, 这就是结果的原码, 写成正负数的形式为: $(-3A)_{16}=(-58)_{10}$ 。

③ $-105-47$

$[-105]_{\text{补}}+[-47]_{\text{补}}$

$[-105]_{\text{补}}=10010111$, $[-47]_{\text{补}}=11010001$;

$$\begin{array}{r} 10010111 \\ + 11010001 \\ \hline 01101000 \end{array}$$

结果的符号位为 0, 似乎运算结果是正数, 但这是不符合逻辑的——两个负数的代数和应为负。错误的原因: -105 与 -47 之和为 -152 , 而用 8 位二进制数只能表示到 -128 , 所以此题的运算结果 -152 超出了 8 位二进制数所能表示的有符号数范围, 这种现象叫做溢出。

如果一定要进行 $(-105)+(-47)$ 的运算, 则应把数据长度扩展到 16 位, 即用 16 位二进制补码运算: $[-105]_{\text{补}}=11111111\ 10010111$, $[-47]_{\text{补}}=11111111\ 11010001$

$$\begin{array}{r} 11111111\ 10010111 \\ + 11111111\ 11010001 \\ \hline 11111111\ 01101000 \end{array}$$

显然结果的符号位为 1, 运算结果是负数的补码。对此结果再求补得到结果的原码: $10000000\ 10011000$, 结果的真值: $(-98)_{16}=(-152)_{10}$, 这才是正确的合乎逻辑的结果。

④ $105+47$

用 8 位二进制补码对这个题进行运算显然也要溢出, 会出现两个正数代数和结果的符号位是 1 的现象 (具体演算请读者自己推导)。要得到正确的运算结果就应把补码位数扩展到 16 位二进制。

$[+105]_{\text{补}} = 00000000\ 01101001$, $[+47]_{\text{补}} = 00000000\ 00101111$

$$\begin{array}{r} 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 0\ 1 \\ +\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 1\ 1\ 1 \\ \hline 0\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 1\ 1\ 0\ 0\ 0 \end{array}$$

结果的符号位为 0, 运算结果是正数, 正数的 $[X]_{\text{补}} = [X]_{\text{原}}$, 所以直接可得结果的真值:
 $(+98)_{16} = (+152)_{10}$ 。

示例 11 8421BCD 码: 取 4 位二进制编码的前 10 种自然状态, 顺序表示十进制的 10 个数字符号。4 位二进制编码的权值分别是 8、4、2、1。

0000	0001	0010	0011	0100	0101	0110	0111	1000	1001
0	1	2	3	4	5	6	7	8	9

在使用 8421BCD 码处理数据时有如下两种码型。

- ① 压缩的 BCD 码: 用一个字节表示两个十进制数的 BCD 码。
- ② 非压缩的 BCD 码: 用一个字节表示一个十进制数的 BCD 码。

用压缩的 8421BCD 码表示 $(74)_{10}$: $(0111\ 0100)_{\text{BCD}}$; 用非压缩的 8421BCD 码表示 $(74)_{10}$: $(00000111\ 00000100)_{\text{BCD}}$ 。

在此要注意, 8421BCD 码仅仅是十进制数在计算机中的一种编码形式, 它与十进制转换为等量的二进制完全是两件事。例如, $(74)_{10}$ 的 8421BCD 码形式是 $(0111\ 0100)_{\text{BCD}}$, 而将 $(74)_{10}$ 转换为二进制数则是 $(01001010)_2$ 。

示例 12 用 ASCII 来表示 “I’m a Chinese.”。

“I’m a Chinese.” 这句话中的每个成分在计算机中都是作为符号用 ASCII 的形式表示的, 具体形式为 49 27 6D 20 61 20 43 68 69 6E 65 73 65 2E (均为十六进制), 其中包括单引号、空格及句号等。

1.3 习题、答案与解析

1.3.1 习题

一、填空题

1. 存储器是计算机的_____部件。
2. CPU 叫做_____, 它内部主要包括_____, 它的基本功能是_____。
3. 计算机系统包括_____和_____。
4. 在微型计算机层次结构中软件系统包括_____, 硬件系统包括_____。
5. 构成微型计算机原理结构的五大部分是_____。
6. 字长作为计算机的主要技术指标, 它标志着_____。
7. 当 CPU 地址线位数为 16、20、24 时, 则 CPU 能够寻址存储单元的最多个数分别

是_____、_____和_____。

8. 对应某二进制数第7数位上的位权值等于_____。
9. 若将字符串‘What time is it?’存到内存中,那么在内存单元中应是_____。
10. 给出下列字符的ASC II (十六进制):

FROG:

Arc:

Water:

Well:

11. 把-34存入内存,在内存单元中该数的存储形态应是_____。
12. 把十六进制的123A存入内存,在内存单元中应是_____。
13. 请把下列十进制数转化为压缩的和非压缩的BCD码形式。
 十进制数102: 压缩BCD码形式_____, 非压缩BCD码形式_____。
 十进制数44: 压缩BCD码形式_____, 非压缩BCD码形式_____。
 十进制数301: 压缩BCD码形式_____, 非压缩BCD码形式_____。
 十进制数1000: 压缩BCD码形式_____, 非压缩BCD码形式_____。

二、选择题

1. 请选择下列二进制数转化为十进制数的正确结果。

① 1101.01

A. 13.25 B. 13.01 C. 31.01 D. 31.25

② 111001.0011

A. 39.3 B. 71.1875 C. 71.3 D. 57.1875

③ 101011.0101

A. 53.5 B. 43.3125 C. 2B.5 D. 223.5

2. 请选择下列八进制数转化为十进制数的正确结果。

① 234.5

A. 1876 B. 156.625 C. 29.3125 D. 234.5

② 12.3

A. 98.4 B. 20.3 C. 10.375 D. 1.5375

③ 7767.07

A. 970.9625 B. 4087.109375 C. 7775.15 D. 62136.56

3. 请选择下列十六进制数转化为十进制数的正确结果。

① A3.3

A. 163.1875 B. 208.48 C. 163.3 D. 103.3

② 129.C

A. 129.192 B. 2064.192 C. 129.12 D. 297.75

③ AC.DC

A. 1012.1312 B. 254.876 C. 160.375 D. 172.859375

4. 请选择下列十六进制数转化为二进制数的正确结果。

① 23

- A. 00010111 B. 10001100 C. 11001000 D. 00100011
- ② AD4
A. 100000110100 B. 101011010100 C. 101001010100 D. 010111010100
- ③ 34.AD
A. 00100010. 10100011 B. 11000100.00010011
C. 00110100.10101101 D. 10101101.00110100
5. 请选择下列二进制数转化为十六进制数的正确结果。
- ① 1100 0010
A. C2 B. 32 C. 2C D. 23
- ② 0001 0000 1111 1101
A. 10FD B. 10EB C. CE10 D. DF01
- ③ 1011 1100
A. DC B. CB C. BC D. DB
6. 请选择下列十进制数转化为 8 位有符号二进制数的正确结果。
- ① +32
A. 00100000 B. 11100000 C. 10100000 D. 00110010
- ② -12
A. 10010010 B. 11101101 C. 00010010 D. 11110100
- ③ +100
A. 00011011 B. 11100100 C. 00010000 D. 01100100
7. 请选择下列二进制数补码转化为有符号十进制数的正确结果。
- ① 1000 0000
A. 80 B. -128 C. 128 D. -80
- ② 0011 0011
A. +51 B. +33 C. -52 D. -53
- ③ 0011 0010
A. +32 B. -47 C. +50 D. -68
8. 请选择下列压缩的 BCD 码转化为十进制数的正确结果。
- ① 1000 1001
A. -9 B. -77 C. 89 D. -89
- ② 0100 1001
A. 73 B. 49 C. +73 D. -37
- ③ 0011 0010
A. -50 B. -47 C. +50 D. 32

三、简答题

1. 微型计算机硬件与微型计算机系统是一回事吗?
2. 请说明以下汇编伪指令的原意是将数据定义为何种类型? 其长度是多少二进制位?

DB —— define byte

DW —— define word

DD —— define double word

3. 请解释位、字节和双字。
4. 回车键的 ASCII 是什么?它的用途是什么?

四、判断题 (判断以下说法的正误)

1. 任何进制的数总可以展开成一个乘法多项式。 ()
2. 任何进制的数总可以展开成一个求和多项式。 ()
3. 任何进制的数总可以展开成一个阶乘多项式。 ()
4. 任何进制的数总可以展开成一个系数与权值相乘的求和多项式。 ()
5. 求有符号数的补码是将其原码求反加 1。 ()
6. 正数的原码、反码及补码形式是一致的。 ()
7. ASCII 用 8 位二进制码表示一个符号。 ()
8. ASCII 的最高位恒为 0。 ()
9. ASCII 最高位的作用是进行奇偶校验。 ()
10. ASCII 用 7 位二进制码表示一个符号。 ()
11. BCD 码用 4 位二进制码表示一位十进制数。 ()
12. 等式: $74_{(BCD)} = 74H$ 成立。 ()
13. 等式: $74_{(BCD)} = 74H$ 不成立。 ()
14. 任何有符号数在计算机中都是用补码表示补码运算的。 ()
15. 在计算机中用浮点数表示小数也可以表示整数。 ()
16. 在计算机中用浮点数表示数据时小数点的位置是不确定的。 ()
17. 在计算机中用浮点数表示数据时小数点的位置是确定的。 ()
18. 浮点数由符号、尾数和阶构成。 ()
19. 浮点数中的阶作为移码而存储。 ()
20. 整个浮点数都是用移码存储的。 ()

1.3.2 答案与解析

一、填空题

1. 信息存储记忆部件
2. 中央处理器, 运算器、控制器以及暂存单元, 数据运算、控制
3. 硬件系统、系统软件
4. 操作系统、高级程序编译、解释程序、各种应用程序, 计算机硬件逻辑、微程序控制、指令系统
5. 运算器、控制器、存储器、输入电路及设备、输出电路及设备
6. CPU 一次可以同时处理的二进制代码的能力
7. $2^{16}=65536$ 个、 $2^{20}=1048576$ 个、 $2^{24}=16777216$ 个
8. $2^7=128$
9. 字符串中各字符的 ASCII。
- 10.

FROG: 46 52 4F 47

Arc: 41 72 63

Water: 57 61 74 65 72

Well: 57 65 6C 6C

11. -34 的补码。

12. 12 存在相对较高的地址单元里, 3A 存在相邻的低地址单元里。

13.

十进制数 102 压缩 BCD 码形式 0001 0000 0010, 非压缩 BCD 码形式 00000001 00000000 00000010。

十进制数 44 压缩 BCD 码形式 01000100, 非压缩 BCD 码形式 00000100 00000100。

十进制数 301 压缩 BCD 码形式 001100000001, 非压缩 BCD 码形式 00000011 00000000 00000001。

十进制数 1000 压缩 BCD 码形式 0001000000000000,

非压缩 BCD 码形式 00000001 00000000 00000000 00000000。

二、选择题

1. ① A ② D ③ B

2. ① B ② C ③ B

3. ① A ② D ③ D

4. ① D ② B ③ C

5. ① A ② A ③ C

6. ① A ② D ③ D 既然是有符号数, 所以正确的答案应是相应数据的补码形式。

7. ① B ② A ③ C 此题是把已给的二进制码序列当作有符号数的补码处理。

8. ① C ② B ③ D

三、简答题

1. 微型计算机硬件与微型计算机系统不是一个概念, 微型计算机硬件是构成微型计算机系统的一部分; 微型计算机系统除了要有硬件外还包括系统软件。

2. DB —— define byte, 将数据定义为字节类型, 长度是 8 个二进制位。

DW —— define word, 将数据定义为字类型, 长度是 16 个二进制位。

DD —— define double word, 将数据定义为双字类型, 长度是 32 个二进制位。

3. 位表示一个二进制数位, 是计算机中表示信息的最基本量化单位。

字节是人为规定的度量信息长度的基本单位。一个字节等于 8 个二进制位。

在汇编语言中 4 个字节为一个双字。

4. 回车键的 ASCII 是 13 (十进制) 或 0D (十六进制), 作用是使光标或打印头返回到一行的左边起始位置。

四、判断题

1~4. 4 正确。乘法多项式、求和多项式的说法都不完整, 阶乘多项式显然是错误的。

5. 错。对于正数的补码就等于其原码。只有负数的补码是将其原码求反加 1。

6. 对。

7. 错。此说法不准确。标准 ASCII 为 7 位代码, 最高有效位在不同的标准和不同的系统中有不同的处理。