

微机原理及应用教学 辅导与习题解析 (第2版)

- ◆ 微机基础知识
- ◆ 微处理器的结构
- ◆ 8086 CPU指令系统
- ◆ 汇编语言程序设计
- ◆ 存储器
- ◆ 输入输出和中断
- ◆ 接口技术



常凤筠 孙红星
张立松 吴文波

主编
参编



清华大学出版社

高等学校计算机应用规划教材

微机原理及应用教学辅导 与习题解析 (第2版)

常凤筠 孙红星 主编

张立松 吴文波 参编

清华大学出版社

北 京

内 容 简 介

本书是为了配合高等院校“微机原理及应用(微机原理及接口技术)”课程的教学而编写的辅导教材。全书共7章,内容包括微机基础知识、微处理器的结构、8086 CPU 指令系统、汇编语言程序设计、存储器、输入输出和中断、接口技术等。为帮助学生更好地理解和掌握该课程的内容,每章按教学基本知识点、重点与难点、典型例题精解、重要习题与考研题解析、习题及参考答案5个部分展开。

本书结构清晰,按照由浅入深、循序渐进的原则精选了大量具有代表性的例题,对每一例题的解题思路、方法进行了详细的分析与解答,每章都有一定数量的习题,并给出了全部习题答案供学生参考。书后的附录还给出了两套期末考试的模拟试题及参考答案。

本书可作为普通高等院校电气信息类专业、计算机专业的辅导教材,也可作为报考硕士研究生的辅导教材及教师的教学参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

微机原理及应用教学辅导与习题解析 / 常凤筠, 孙红星 主编. —2 版. —北京: 清华大学出版社, 2016
(高等学校计算机应用规划教材)

ISBN 978-7-302-44464-0

I. ①微… II. ①常… ②孙… III. ①微型计算机—高等学校—教学参考资料 IV. ①TP36

中国版本图书馆 CIP 数据核字(2016)第 171521 号

责任编辑: 王 军 韩宏志

封面设计: 牛艳敏

版式设计: 牛静敏

责任校对: 成凤进

责任印制: 何 芊

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 北京密云胶印厂

经 销: 全国新华书店

开 本: 185×260

印 张: 14.25

字 数: 329 千字

版 次: 2011 年 10 月第 1 版

2016 年 8 月第 2 版

印 次: 2016 年 8 月第 1 次印刷

印 数: 1~4000

定 价: 28.00 元

产品编号: 070413-01

前 言

“微机原理及应用”是国家教育部规定的各大、专院校计算机专业、电气信息类专业的一门重要的主干专业技术基础课,是机电类、信息工程等专业的必修课程,是相应的计算机等级考试科目,同时也是计算机、电子信息等专业研究生入学的考试科目。

“微机原理及应用”这门课程知识点多,教学内容比较抽象,前后联系紧密,初学者常感到该课程难理解、难学。书中通过大量具有代表性的例题对重点和难点内容进行详细分析与解答,其目的是让学生掌握课程的内容,掌握微机应用系统的分析和设计方法,提高综合运用软硬件能力,提高分析问题和解决问题的能力,为学生今后从事智能控制、计算机开发等工作打下良好基础。

本书是为配合高等院校“微机原理及应用(微机原理及接口技术)”课程的教学而编写的辅导教材。全书共7章,内容包括微机基础知识、微处理器的结构、8086 CPU 指令系统、汇编语言程序设计、存储器、输入输出和中断、接口技术等。

为帮助学生更好地理解 and 掌握该课程的内容,每章按基本知识点、重点与难点、典型例题精解、重要习题与考研题解析、习题及参考答案5个部分展开。教学基本知识点部分总结本章的学习内容、教学要求,并对每章的教学内容进行归纳、总结,使学生全面深入地掌握基本概念、基本原理。重点与难点部分指出每章的重点和难点内容,重点内容要求学生全面掌握。典型例题精解部分按照由浅入深、循序渐进的原则针对重点内容精选了大量具有代表性的例题,并对每一例题的解题思路、方法进行详细分析与讲解,不但使学生加深对基本原理、基本知识的理解和掌握,而且能够掌握解题的方法。重要习题与考研题解析部分是对知识点的综合,通过列举大量例题,掌握解题方法和技巧,提高分析问题和解决问题的能力。习题及参考答案部分提供一定数量的习题并给出全部习题答案,供学生参考。本书附录还给出了两套期末考试的模拟试题及参考答案。本书可作为普通高等院校电气信息类专业、计算机专业的辅导教材,也可作为报考硕士研究生的辅导教材及教师的教学参考书。

本书第1章、第2章由吴文波、欧阳鑫玉编写,第3章由常凤筠编写,第4章由常凤筠、孙红星编写,第5章由欧阳鑫玉、常凤筠编写,第6章、第7章由张立松编写。全书由常凤筠统稿。

值得注意的是,为美观起见,作者对指令和代码中的逗号和冒号等均采用汉字标点符号,在实际编程时,同学们要注意采用英文标点符号。

由于编者水平有限,时间仓促,书中难免有疏漏和不当之处,敬请读者提出宝贵意见。

编 者

目 录

第1章 微机基础知识 1	
1.1 基本知识点..... 1	
1.1.1 计算机中的运算基础..... 1	
1.1.2 计算机中数据的编码..... 4	
1.1.3 微机系统的基本组成..... 6	
1.2 重点与难点..... 7	
1.3 典型例题精解..... 8	
1.4 重要习题与考研题解析..... 11	
1.5 习题及参考答案..... 13	
1.5.1 习题..... 13	
1.5.2 参考答案..... 15	
第2章 微处理器的结构 17	
2.1 基本知识点..... 17	
2.1.1 8086/8088 CPU 的结构..... 17	
2.1.2 8086/8088 CPU 芯片的引脚 及其功能..... 20	
2.1.3 8086/8088 存储器的结构..... 23	
2.1.4 总线结构和总线周期..... 25	
2.1.5 微处理器的发展..... 27	
2.2 重点与难点..... 29	
2.3 典型例题精解..... 29	
2.4 重要习题与考研题解析..... 33	
2.5 习题及参考答案..... 35	
2.5.1 习题..... 35	
2.5.2 参考答案..... 38	
第3章 8086 CPU 指令系统 41	
3.1 基本知识点..... 41	
3.1.1 8086 指令的一般格式..... 41	
3.1.2 8086 寻址方式..... 41	
3.1.3 8086 指令系统概述..... 43	
3.2 重点与难点..... 46	
3.3 典型例题精解..... 46	
3.4 重要习题与考研题解析..... 56	
3.5 习题及参考答案..... 61	
3.5.1 习题..... 61	
3.5.2 参考答案..... 74	
第4章 汇编语言程序设计 81	
4.1 基本知识点..... 81	
4.1.1 汇编语言语句的种类和 格式..... 81	
4.1.2 常用伪指令..... 82	
4.1.3 运算符..... 84	
4.1.4 汇编语言程序设计的 基本步骤..... 86	
4.1.5 程序的基本结构与基本 程序设计..... 87	
4.2 重点与难点..... 89	
4.3 典型例题精解..... 89	
4.3.1 伪指令典型例题精解..... 89	
4.3.2 顺序结构程序典型例题 精解..... 94	
4.3.3 分支结构程序典型例题 精解..... 95	
4.3.4 循环结构程序举例..... 101	
4.4 重要习题与考研题解析..... 108	
4.5 习题及参考答案..... 111	
4.5.1 习题..... 111	
4.5.2 参考答案..... 115	
第5章 存储器 123	
5.1 基本知识点..... 123	
5.1.1 存储器概述..... 123	
5.1.2 随机存储器..... 125	

5.1.3 只读存储器·····	126	6.5 习题及参考答案·····	155
5.1.4 半导体存储器与微处理器的 连接·····	126	6.5.1 习题·····	155
5.2 重点与难点·····	128	6.5.2 参考答案·····	164
5.3 典型例题精解·····	128	第7章 接口技术 ·····	169
5.3.1 存储器的位扩展和字扩展·····	128	7.1 基本知识点·····	169
5.3.2 存储器的片间寻址扩展·····	130	7.1.1 定时/计数器 8253·····	169
5.4 重要习题与考研题解析·····	132	7.1.2 并行接口 8255·····	172
5.5 习题及参考答案·····	134	7.2 重点与难点·····	175
5.5.1 习题·····	134	7.3 典型例题精解·····	175
5.5.2 参考答案·····	138	7.4 重要习题与考研题解析·····	181
第6章 输入输出和中断 ·····	141	7.5 习题及参考答案·····	190
6.1 基本知识点·····	141	7.5.1 习题·····	190
6.1.1 输入和输出·····	141	7.5.2 参考答案·····	198
6.1.2 中断·····	143	附录 A 微机原理及应用课程考试 试题 ·····	205
6.2 重点与难点·····	148	参考文献 ·····	219
6.3 典型例题精解·····	148		
6.4 重要习题与考研题解析·····	152		

第1章 微机基础知识

1.1 基本知识点

1.1.1 计算机中的运算基础

1. 数制及其转换

1) 任意进制数的共同特点(n 进制) $n=2、8、10、16$

① n 进制数最多由 n 个数码组成

- 十进制数的组成数码为：0~9。
- 二进制数的组成数码为：0、1。
- 八进制数的组成数码为：0~7。
- 十六进制数的组成数码为：0~9、A~F。
- 十六进制数和十进制数的对应关系是：0~9 相同，A-10，B-11，C-12，D-13，E-14，F-15。

② n 进制数的基数或底数为 n ，作算术运算时，有如下特点：

- 低位向相邻高位的进位是逢 n 进 1(加法)。
- 低位向相邻高位的借位是以 1 当本位 n (减法)。

③ 各位数码在 n 进制数中所处位置不同，所对应的权也不同，以小数点为分界点：

- 向左(整数部分)：各位数码所对应的权依次是 $n^0、n^1、n^2, \dots$
- 向右(小数部分)：各位数码所对应的权依次是 $n^{-1}、n^{-2}、n^{-3}, \dots$

例 1.1

十进制数：	3	3	3	.	3	3
	↓	↓	↓		↓	↓
各位对应的权为：	10^2	10^1	10^0		10^{-1}	10^{-2}
二进制数：	1	0	1	.	1	1B
	↓	↓	↓		↓	↓
各位对应的权为：	2^2	2^1	2^0		2^{-1}	2^{-2}
十六进制数：	F	9	4			
	↓	↓	↓			
各位对应的权为：	16^2	16^1	16^0			

2) 数制的转换

① 非十进制数→十进制数

转换方法：按位权展开求和。

例 1.2

$$\begin{aligned} 101.11\text{B} &= 1 \times 2^2 + 1 \times 2^0 + 1 \times 2^{-1} + 1 \times 2^{-2} \\ &= 4 + 1 + 0.5 + 0.25 \\ &= 5.75 \end{aligned}$$

$$\begin{aligned} \text{F94H} &= 15 \times 16^2 + 9 \times 16^1 + 4 \times 16^0 \\ &= 3988 \end{aligned}$$

注意：只有十进制数的下标可以省略，其他进制数不可以省略。

② 十进制数→非十进制数(K 进制数)

转换方法：分成小数和整数，分别转换。

整数部分：除 K 取余，直至商为 0，先得的余数为低位。

小数部分：乘 K 取整，先得的整数为高位。

例 1.3

把 3988 转换成十六进制数表示。

16	3988	4	↑ 低位
16	249	9	↑ 高位
	15(F)		

所以：3988=F94H

十进制数转换为二进制数的另一种方法：逐次减 2 的最高次幂法。

$$2^1=2, 2^2=4, 2^3=8, 2^4=16, 2^5=32, 2^6=64, \dots$$

例 1.4

将 1539 转换为二进制数表示。

1539		→	2^{10}
-1024		→	2^9
515		→	2^8
-512		→	2^7
3		→	2^0
-2		→	2^1
1		→	2^0

所以：1539=110 0000 0011B

例 1.5

将 0001 1010 1110 1101 1011.0100B 转换为十六进制。

十六进制为：1 A E D B .4 H

若十六进制数转换为二进制数，则将每一位拆成 4 位。

2. 模的概念

若 a 和 b 除以 M , 余数相等, 则称 a 和 b 对于 M 是同余的, 可以写成: $a = b(\text{mod } M)$ 。
容器的最大容量称为模。可写成: $KM + X = X(\text{mod } M)$

3. 有符号数在计算机中的表示方法

在计算机中, 一个有符号数可以用原码、补码和反码表示。

1) 共同规律。

① 用 0 表示正号, 用 1 表示负号, 且摆放在数据的最高位。有符号数和无符号数表示的根本区别在于, 无符号数的最高位是数值位, 有符号数的最高位是符号位。

② 同一正数的原、补、反码都相同。

③ 定义区间均对模 2^n 而言, 其中 n 表示有符号数的二进制代码位数。

2) 其他规律。

① 任一负数的原码和对应的正数(绝对值相等)的原码仅是符号位不同。

② 任一负数的反码是对应的正数的反码的各位求反, 反之亦然。

③ 任一负数的补码是对应的正数的补码的各位求反, 然后加 1, 反之亦然。

④ 从定义区间上看, 原码和反码的定义区间相同, 是 $-2^{n-1} < X < 2^{n-1}$; 补码的定义区间是 $2^{n-1} \leq X < 2^n$;

⑤ 0 的原码、反码有 +0 和 -0 之分; 0 的补码只有一种表达方式。

4. 补码、反码加减运算规则

$$[X+Y]_{\text{补}} = [X]_{\text{补}} + [Y]_{\text{补}}$$

$$[X+Y]_{\text{反}} = [X]_{\text{反}} + [Y]_{\text{反}}$$

$$[X-Y]_{\text{补}} = [X]_{\text{补}} + [-Y]_{\text{补}}$$

$$[X-Y]_{\text{反}} = [X]_{\text{反}} + [-Y]_{\text{反}}$$

$$[-Y]_{\text{补}} = [[Y]_{\text{补}}]_{\text{补}}$$

$$[-Y]_{\text{反}} = [[Y]_{\text{反}}]_{\text{反}}$$

5. 基本名词

位: BIT, 缩写为 b;

字节: BYTE, 由 8 位二进制数代码表示, 缩写为 B;

字: WORD, 取决于计算机 CPU 的字长, 内部寄存器的位数, 其中 8086 CPU 为 16 位, 386、486 CPU 为 32 位;

千字节: $1\text{KB} = 1024\text{B} = 2^{10}\text{B}$

兆字节: $1\text{MB} = 2^{20}\text{B} = 1024\text{KB}$

吉字节: $1\text{GB} = 2^{30}\text{B} = 1024\text{MB}$

太字节: $1\text{TB} = 2^{40}\text{B} = 1024\text{GB}$

6. 带符号数运算时的溢出问题

溢出和进位的区别: 进位是指最高位向更高位的进位, 而溢出是指运算结果超出数所能表示的范围。

带符号数所能表示的范围: (若用 n 位二进制数码表示)

原码: $-(2^{n-1}-1) \leq X \leq 2^{n-1}-1$

补码: $-2^{n-1} \leq X < 2^{n-1}$

反码: $-(2^{n-1}-1) \leq X \leq 2^{n-1}-1$

溢出的判断方法:

设 CD7 是符号位向更高位的进位, CD6 是数值位向符号位的进位, 则溢出可用 $V=CD7 \oplus CD6$ 判断, $V=1$ 表示有溢出, $V=0$ 表示无溢出。

对于加减法, 也可以这样判断, 只有下面 4 种情况有可能产生溢出:

- 正数+正数, 结果应为正, 若为正, 则无溢出; 若为负, 则有溢出。
- 负数+负数, 结果应为负, 若为负, 则无溢出; 若为正, 则有溢出。
- 正数-负数, 结果应为正, 若为正, 则无溢出; 若为负, 则有溢出。
- 负数-正数, 结果应为负, 若为负, 则无溢出; 若为正, 则有溢出。

对于乘(除)法, 乘积(商)超过了能存放的范围有溢出, 否则无溢出。其他情况肯定无溢出。

注意: 无符号数和带符号数表示方法有区别。无符号数: 无符号位, 所有位都是数值位, 即最高位也是数值位; 带符号数: 有符号数, 且在最高位, 其余各位才是数值位。

1.1.2 计算机中数据的编码

1. 十进制数在计算机中的表示方法

BCD(Binary Coded Decimal)是用 4 位二进制代码表示一位十进制数, 4 位二进制代码表示 16 种状态, 而十进制数只取其中 10 种状态。选择不同的对应规律, 可以得到不同形式的 BCD 码。最常用的是 8421BCD 码。

例 1.6

$59 = (0101, 1001)BCD;$

$465 = (0100, 0110, 0101)BCD$

$(011010000010)BCD = (0110, 1000, 0010)BCD = 682$

注意: BCD 码与二进制数之间不能直接转换, 需要将 BCD 码先转换成十进制数, 再由十进制数转换为二进制数。与十六进制数的区别在于: 组内逢 2 进 1, 组间逢 10 进 1。

表 1-1 是 8421 BCD 码。

表 1-1 8421 BCD 码

十 进 制 数	8421 BCD 码	十 进 制 数	8421 BCD 码
0	0000	5	0101
1	0001	6	0110
2	0010	7	0111
3	0011	8	1000
4	0100	9	1001

2. 字符在计算机中的表示方法

由于大、小写英文字母、0~9 数字字符、标点符号、计算机特殊控制符一共不超过 128 个,所以只要用 7 位二进制数码来表示,称为 ASCII 码,见表 1-2。国际标准为 ISO 646,我国国家标准为 GB 1988。在计算机中,一个字符通常用一个字节(八位)表示,最高位通常为 0 或用于奇偶校验位。ISO 2022 标准在兼容 ISO 646 的基础上扩展成 8 位码,可表示 256 个字符,扩充了希腊字母、数学符号、非拉丁字符、商用图符、游戏符号等。

例 1.7

‘A’= 41H = 01000001B; ‘0’= 30H = 00110000B;
‘a’= 61H = 01100001B; ‘;’= 3BH = 00111011B。

3. 机器数和真值

机器数: 一个数及其符号位在机器中的一组二进制数的表示形式;

真值: 机器数所表示的值。

例 1.8

机器数 34H, 用原码表示为+52; 用反码表示为+52; 用补码表示为+52; 用 BCD 码表示为 34; 用 ASCII 码表示为 4。

即[+52]原=[+52]反=[+52]补=34H

[34]BCD = 34H

[4]ASCII = 34H

机器数 97H, 用原码表示为-23; 用反码表示为-104; 用补码表示为-105; 用 BCD 码表示为 97; 用 ASCII 码表示为 ETB。

表 1-2 ASCII 码字符表

编 码	控 制 字 符	编 码	字 符	编 码	字 符	编 码	字 符
00	NUL	20	SPACE	40	@	60	`
01	SOH	21	!	41	A	61	a
02	STX	22	"	42	B	62	b
03	ETX	23	#	43	C	63	c
04	EOT	24	\$	44	D	64	d
05	ENQ	25	%	45	E	65	e
06	ACK	26	&	46	F	66	f
07	BEL	27	'	47	G	67	g
08	BS	28	(	48	H	68	h
09	TAB	29	)	49	I	69	i
0A	LF	2A	*	4A	J	6A	j
0B	VT	2B	+	4B	K	6B	k

(续表)

编 码	控 制 字 符	编 码	字 符	编 码	字 符	编 码	字 符
0C	FF	2C	,	4C	L	6C	l
0D	CR	2D	-	4D	M	6D	m
0E	SO	2E	.	4E	N	6E	n
0F	SI	2F	/	4F	O	6F	o
10	DLE	30	0	50	P	70	p
11	DC1	31	1	51	Q	71	q
12	DC2	32	2	52	R	72	r
13	DC3	33	3	53	S	73	s
14	DC4	34	4	54	T	74	t
15	NAK	35	5	55	U	75	u
16	SYN	36	6	56	V	76	v
17	ETB	37	7	57	W	77	w
18	CAN	38	8	58	X	78	x
19	EM	39	9	59	Y	79	y
1A	SUB	3A	:	5A	Z	7A	z
1B	ESC	3B	;	5B	[	7B	{
1C	FS	3C	<	5C	\	7C	
1D	GS	3D	=	5D	]	7D	}
1E	RS	3E	>	5E	^	7E	~
1F	US	3F	?	5F	_	7F	DEL

1.1.3 微机系统的基本组成

由硬件系统和软件系统两部分组成, 并采用总线结构。

1. 硬件系统

硬件系统是指构成微机系统的全部物理装置。通常, 计算机硬件系统由 5 部分组成:

- 1) 存储器: 用来存放数据和程序, 例如半导体存储器、磁介质存储器。
- 2) 微处理器(包括运算器和控制器): 运算器用来完成二进制编码的算术和逻辑运算; 控制器控制计算机进行各种操作的部件。微机硬件系统只不过把运算器和控制器用大规模集成电路工艺技术集成在一块芯片上, 这块芯片称为 CPU(中央处理单元)。
- 3) 输入设备及其接口电路: 用来输入数据、程序、命令和各种信号, 例如键盘、鼠标等。
- 4) 输出设备及其接口电路: 用来输出计算机处理的结果, 例如打印机、CRT 等。
- 5) 网络设备。

2. 软件系统

软件系统是指计算机所编制的各种程序的集合, 可分为两大类:

1) 系统软件

系统软件是用来实现对计算机资源的管理、控制和维护, 便于人们使用计算机而配置的软件, 该软件由厂家提供。它包括操作系统(或监控管理程序), 各种语言的汇编、解释、编译程序, 数据库管理程序, 编辑、调试、装配、故障检查和诊断等工具软件。

操作系统在系统软件中具有特殊地位。只要计算机处于工作状态, 就有操作系统的有关部分在内存存储器中, 负责接受、分析并调度执行用户的程序和各种命令。Windows 是目前最流行的微机操作系统。

2) 应用软件

应用软件是指用户利用计算机以及它所提供的各种系统软件编制的解决各种实际问题的程序。它包括支撑软件 and 用户自己编制的程序。

支撑软件有:

- 文字处理软件: Wordstar、Write、WPS、Word、中文之星等。
- 表格处理软件: Lotus1-2-3、CCED、Excel 等。
- 图形处理软件: AutoCAD、TANGO、PowerPoint、PROTEL 98 以及 2000 等。
- 图文排版软件: 华光、科印、方正等。
- 防治病毒软件: SCAN、KILL、CLEAN、MSAV、KV 3000。
- 工具软件: PCTOOLS 等。
- 套装软件: Microsoft-Office, 它基于 Windows, 包括 Word、Excel、PowerPoint、MS Mail 等。

3. 软、硬件的关系

硬件系统是人们操作微机的物理基础; 软件系统是人们与微机系统进行信息交换、通信对话、按人的思维对微机系统进行控制和管理的工具。

4. 微机的总线结构

1) 总线: 是指连接多于两个部件的公共信息通路, 或者说是多个部件之间的公共连线。

2) 按照总线上传送的信息内容分类:

- 数据总线(DB): 传送数据信息。
- 控制总线(CB): 传送控制信息, 确定数据信息的流向。
- 地址总线(AB): 传送地址信息, 确定数据信息的传送地址。

1.2 重点与难点

重点: 掌握计算机中的各种数制及其相互转换, 机器数的编码表示及其相互转换与运

算：搞清微型计算机的基本组成及其各模块的功能。

难点：掌握二进制运算中溢出和进位的区别；弄清机器数和真值；理解指令在计算机中的执行过程。

1.3 典型例题精解

例 1.9

求 $152.76 = \underline{\hspace{1cm}} B = \underline{\hspace{1cm}} Q = \underline{\hspace{1cm}} H$ 。

解：

整数部分：

$$\begin{array}{r} 8 \overline{) 152} \quad \dots 0 \\ \underline{8 \overline{) 19}} \quad \dots 3 \\ \underline{8 \overline{) 2}} \quad \dots 2 \\ 0 \end{array}$$

逆取法得： $152 = 230Q = 10\ 011\ 000B = 98H$

小数部分(精确到小数点后 3 位)：

$$0.76 \times 8 = 6.08 \quad \text{取整} = 6$$

$$0.08 \times 8 = 0.64 \quad \text{取整} = 0$$

$$0.64 \times 8 = 5.12 \quad \text{取整} = 5$$

顺取法得： $0.76 = 0.605Q = 0.\underline{011}\ \underline{000}\ \underline{101}B = 0.628H$

所以： $152.76 = \underline{1001}\ \underline{1000}.\underline{0110}\ \underline{0010}B = 230.605Q = 98.628H$

注意：手工变换时，可先变换成八进制，再变为其他进制，这样会减少计算工作量和变换次数。八进制转换为二进制时，将每一位八进制数用 3 位二进制数表示，再去掉首位的零即可(观察划线部分)。二进制数转换为十六进制时，将每 4 位二进制数用 1 位十六进制数表示即可(观察划线部分)，注意要以小数点为分界线分别向左和向右表示。

例 1.10

求 $7A.18H = \underline{\hspace{1cm}} B = \underline{\hspace{1cm}} D = \underline{\hspace{1cm}} Q$ 。

解：十六进制可直接转换为二进制，二进制再直接转换为八进制，十六进制转换为十进制采用定义变换。

根据定义变换：

$$7A.18H = 7 \times 16^1 + 10 \times 16^0 + 1 \times 16^{-1} + 8 \times 16^{-2} = 122.09375D$$

$$7A.18H = \underline{0111}\ \underline{1010}.\underline{0001}\ \underline{1000}B = 1111010.00011B$$

$$1111010.00011B = \underline{001}\ \underline{111}\ \underline{010}.\underline{000}\ \underline{110}B = 172.06Q$$

所以： $7A.18H = 1111010.00011B = 122.09375D = 172.06Q$

注意：十六进制转换为二进制时，将每一位十六进制数用4位二进制数表示，再去掉首位的零即可(观察划线部分)；二进制数转换为八进制时，将每3位二进制数用1位八进制数表示即可(观察划线部分)，注意要以小数点为分界线分别向左和向右表示。

例 1.11

写出下列数的原码、反码及补码表示(设机器数字长为8位)。

+24, -24, +0, -0, +1, -1, +127, -127

解：首先将所给的数转换为二进制数，然后根据原码、反码和补码的表示法及其字长，写出指定数据的原码、反码和补码表示。

例如，写出“+24”、“-24”的原码、反码和补码，表示如下：

(1) 写出24的二进制数表示：24D=00011000B

(2) $[+24]_{\text{原}}=00011000\text{B}$ $[-24]_{\text{原}}=10011000\text{B}$

最高位(D7)为符号位，为1表示负数，为0表示正数，其余7位为24对应的二进制数值位。

(3) $[+24]_{\text{反}}=00011000\text{B}$ $[-24]_{\text{反}}=11100111\text{B}$

正数的反码就是正数的原码，负数的反码等于负数的原码的符号位不变，其余7位数值位取反。

(4) $[+24]_{\text{补}}=00011000\text{B}$ $[-24]_{\text{补}}=11101000\text{B}$

正数的补码就是正数的原码，负数的补码等于负数的原码的符号位不变，其余7位数值位取反，并且在末位加1。

依照上述方法，可写出其余各数的原码、反码及补码表示：

0D=00000000; $[+0]_{\text{原}}=00000000\text{B}$; $[+0]_{\text{反}}=00000000\text{B}$; $[+0]_{\text{补}}=00000000\text{B}$;

$[-0]_{\text{原}}=10000000\text{B}$; $[-0]_{\text{反}}=11111111\text{B}$; $[-0]_{\text{补}}=00000000\text{B}$;

1D=00000001; $[+1]_{\text{原}}=00000001\text{B}$; $[+1]_{\text{反}}=00000001\text{B}$; $[+1]_{\text{补}}=00000001\text{B}$;

$[-1]_{\text{原}}=10000001\text{B}$; $[-1]_{\text{反}}=11111110\text{B}$; $[-1]_{\text{补}}=11111111\text{B}$;

127D=11111111; $[+127]_{\text{原}}=01111111\text{B}$; $[+127]_{\text{反}}=01111111\text{B}$; $[+127]_{\text{补}}=01111111\text{B}$

$[-127]_{\text{原}}=11111111\text{B}$; $[-127]_{\text{反}}=10000000\text{B}$; $[-127]_{\text{补}}=10000001\text{B}$ 。

注意：解答这类题时，要注意正数的原码、反码和补码表示形式是一样的，千万不要用求负数的原码、反码和补码表示方法来做。

例 1.12

已知 $X=-101011\text{B}$, $Y=+101100\text{B}$, 机器数的字长为8位, 求 $[X+Y]_{\text{补}}$, $X+Y$, $[X-Y]_{\text{补}}$, $X-Y$ 。

解：

(1) 求出 $[X]_{\text{原}}$, $[Y]_{\text{原}}$

$[X]_{\text{原}}=10101011\text{B}$ $[Y]_{\text{原}}=00101100\text{B}$

(2) 求出 $[X]_{\text{补}}$, $[Y]_{\text{补}}$

$[X]_{\text{补}}=11010101\text{B}$ $[Y]_{\text{补}}=00101100\text{B}$

(3) 求出 $[X+Y]_{\text{补}}$

$$[X+Y]_{\text{补}} = [X]_{\text{补}} + [Y]_{\text{补}} = 11010101\text{B} + 00101100\text{B} = 00000001$$

(4) 求出 $X+Y$

根据 $[X+Y]_{\text{补}}$ 求出 $X+Y$ 。其符号位为“0”表示结果为正，其余 7 位就是 $X+Y$ 的值。
所以 $X+Y=1\text{D}$ 。

(5) 求出 $[X-Y]_{\text{补}}$

$$[X-Y]_{\text{补}} = [X]_{\text{补}} - [Y]_{\text{补}} = 11010101\text{B} - 00101100\text{B} = 10101001\text{B}$$

(6) 求出 $X-Y$

根据 $[X-Y]_{\text{补}}$ 求出 $X-Y$ 。其符号位为“1”表示结果为负，其余 7 位二进制数按位取反后，末位再加“1”可得到 $X-Y$ 的值。所以 $X-Y=-87\text{D}$ 。

注意：计算时要注意补码的求法及补码加减法的规则。

例 1.13

完成下列 BCD 码运算， $64+56=$ ____， $64-56=$ ____。

解：

(1) 将给定的十进制数用 BCD 码表示

$$64\text{D}=01100100\text{BCD}$$

$$56\text{D}=01010110\text{BCD}$$

(2) 进行 BCD 加法运算得到加法中间结果

$$01100100\text{BCD} + 01010110\text{BCD} = 10111010\text{BCD}$$

(3) 调整得到加法最终结果

十进制调整的方法：

运算后低 4 位=1010，超过 1001，低 4 位加 6；运算后高 4 位=1011，超过 1001，高 4 位加 6。

$$10111010\text{BCD} + 01100110\text{BCD} = 00100000\text{BCD}, \text{CF}=1。$$

(4) $64+56=(1)20$ ，其中百位为进位位。

(5) 进行 BCD 减法运算得到减法中间结果

$$01100100\text{BCD} - 01010110\text{BCD} = 00001110\text{BCD}$$

(6) 调整得到减法最终结果

十进制调整的方法：运算后低 4 位=1110，超过 1001，低 4 位减 6；运算后高 4 位=0000，不超过 1001，高 4 位减 0。

$$00001110\text{BCD} - 00000110\text{BCD} = 00001000\text{BCD}。$$

(7) $64-56=8$

注意：本题中 BCD 的加减法运算仍采用二进制运算规则，得到的数为十六进制数，需要进行十进制调整。这部分内容将在下一章讲解。

例 1.14

概述计算机的基本组成部件及其各组成部件的功能。

答：一台计算机由控制器、运算器、存储器、输入设备和输出设备组成。

(1) 存储器

存储器是用来存放数据、程序、运算的中间结果和最终结果的部件。存储器采用按地址存取的工作方式，它由许多存储单元组成，每一个存储单元可以存放一个数据代码。为了区分不同存储单元，把全部存储单元按照一定的顺序编号。这个编号称为存储单元的地址。当 CPU 要把一个数据代码存入某存储单元或从某存储单元取出时，首先要提供该存储单元的地址，然后查找相应的存储单元，最后才能进行数据的存取。

(2) 运算器

运算器是对信息进行加工、运算的部件，它对二进制进行基本逻辑运算和算术运算，将结果暂存或送到存储器保存。

(3) 控制器

控制器是计算机的控制中心。存储器进行信息的存取，运算器进行各种运算，信息的输入和输出都是在控制器的统一控制下进行的。控制器的工作就是周而复始地从存储器中取指令、分析指令，向运算器、存储器以及输入输出设备发出控制命令，控制计算机工作。

(4) 输入设备

程序员编好的程序和数据是经输入设备送到计算机中去的。输入设备要将程序和数据转换为计算机能识别和接受的信息，如电信号等。目前常用的输入设备有键盘、鼠标、扫描仪等。

(5) 输出设备

输出设备是把运算结果转换为人们所需要的易于理解、阅读的形式。目前常用的输出设备包括显示器、打印机、绘图仪等。软磁盘、硬磁盘、可读写光盘及其驱动器既是输入设备也是输出设备，只读光盘及其驱动器属于计算机的输入设备。软盘、硬盘及光盘又统称为计算机的外存储器。

1.4 重要习题与考研题解析

例 1.15

(上海大学 2001 年考研题)下列无符号数中，最大的数是()。

A. (1100110)二进制数

B. (143)八进制数

C. (10011000)BCD

D. (65)十六进制数

分析：本题主要考查不同进制下数的大小，即考查学生对各种进制之间的互换掌握程度。可考虑都转换为二进制。

(1) $143_{10} = 11000111_2$

$65_{16} = 1100101_2$