

计算机的功能就是进行数据处理(信息处理),目前的计算机不仅能处理数值型数据,还能处理非数值型数据,包括英文字符、汉字、图像、音频和视频等多种媒体数据。数据在计算机中的表示与存储是数据处理的基础。我们知道,所有数据在计算机内部都是以二进制形式存储的,那么现实中丰富多彩的多种媒体数据是如何转换为计算机中的二进制数据呢?本章对此进行介绍。

3.1 计算机中的进制

3.1.1 进位记数制

按进位的原则进行记数称为进位记数制,简称进制或数制。日常生活中,人们习惯于用十进制进行记数。但在计算机内部,为了便于数据的表示和计算,采用二进制记数方法。二进制数在计算机中易于表示(只有 0 和 1 两种形式)、存储和计算,但二进制数的一个缺点是表示一个数需要的二进制位数较多,给人们的阅读、书写、记忆等带来不便。例如,十进制数 $(2695)_{10}$,用二进制数表示则需要 12 位二进制数字 $(101010000111)_2$ 。为了便于人们阅读和书写,在编写程序时,经常使用十进制数,有时也用八进制数和十六进制数。

不同进制有不同的基数和位权。

1. 基数

每种进制中数码的个数称为该进制的基数。例如,二进制中只有两个数码(0 和 1),其基数为 2,计算时逢 2 进 1;十进制中有 10 个数码(0~9),其基数为 10,计算时逢 10 进 1。

2. 位权

在每种进制中,一个数码所处的位置不同,代表的数值大小也不同,称为具有不同的位权。例如,十进制数 9999,最左边的 9 代表 9 千,最右边的 9 代表 9 个。也就是说,该十进制数从右向左的位权依次是个(10^0)、十(10^1)、百(10^2)和千(10^3)。

在编写程序时,根据需要可以用二进制、十进制、八进制或十六进制来表示数据,但在计算机内部,只能以二进制形式表示和存储数据(包括程序代码也只能用二进制表示)。所以计算机在运行程序时,需要先把其他进制数据转换为二进制数据再进行处理,处理结果(二进制形式)在输出时再转换为其他进制,以方便用户阅读和使用。表 3.1 给出了常用进制的基数和所需要的数码,表 3.2 给出了常用进制的表示方法。从表 3.2 可以看出,十进制数 13 和二进制数 1101、八进制数 15、十六进制数 D 是等价的。

表 3.1 常用进制的基数和数码

进 制	基 数	数 码
二进制	2	0 1
八进制	8	0 1 2 3 4 5 6 7
十进制	10	0 1 2 3 4 5 6 7 8 9
十六进制	16	0 1 2 3 4 5 6 7 8 9 A B C D E F

注:因为只有 10 个数字符号(0~9),而十六进制需要 16 个数码,所以除 0~9 外,再选用 A~F(也可以写为小写形式 a~f)作为十六进制的数码。

表 3.2 常用进制的表示方法

十 进 制 数	二 进 制 数	八 进 制 数	十 六 进 制 数
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

3.1.2 不同进制数据的区分

为了便于区分各种进制的数据,常采用后缀或前缀法、角标法书写数据。

1. 后缀或前缀法

在数据后面或前面加写相应的英文字母作为标识,这种方式便于计算机识别。

B(binary)表示二进制数,二进制数 101 可写成 101B。

O(octonary)表示八进制数,八进制数 101 可写成 101O 或 101Q(由于字母 O 与数字 0 容易混淆,常用 Q 代替 O)。

D(decimal)表示十进制数,十进制数 101 可写成 101D(D 可省略)。

H(hexadecimal)表示十六进制数,十六进制数 101 可写成 101H。

2. 角标法

用圆括号把数据括起来,在括号外面加数字角标,这种方式便于人工阅读。

$(101)_2$ 表示二进制数 101。

(101)₈ 表示八进制数 101。
(101)₁₀ 表示十进制数 101,十进制数可省略角标。
(101)₁₆ 表示十六进制数 101。

3.2 不同进制数据的相互转换

不同进制的数据可以按一定规则进行相互转换。

3.2.1 二进制数与十进制数的相互转换

1. 二进制数转换为十进制数

二进制数转换为十进制数,按权展开相加即可。

【例 3.1】 把二进制数(1101101.10111)₂ 转换为十进制数。

$$\begin{aligned}(1101101.10111)_2 &= 1 \times 2^0 + 0 \times 2^1 + 1 \times 2^2 + 1 \times 2^3 + 0 \times 2^4 + 1 \times 2^5 + 1 \times 2^6 + \\ &\quad 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4} + 1 \times 2^{-5} \\ &= 1 + 4 + 8 + 32 + 64 + 0.5 + 0.125 + 0.0625 + 0.03125 \\ &= (109.71875)_{10}\end{aligned}$$

2. 十进制数转换为二进制数

十进制数转换为二进制数时对整数部分和小数部分分别转换,整数部分采用除 2 取余法,直至商为 0 止;小数部分采用乘 2 取整法,满足精度要求为止。

【例 3.2】 把十进制数(78.69)₁₀ 转换为二进制数,小数部分保留 4 位。

先采用除 2 取余法把 78 转换为二进制数,算式如图 3.1 所示,结果为 1001110;再采用乘 2 取整法把 0.69 转换为二进制数,算式如图 3.2 所示,结果的近似值为 0.1011。得到转换结果为(78.69)₁₀ ≈ (1001110.1011)₂。

2	78	
2	39	-----0
2	19	-----1
2	9	-----1
2	4	-----1
2	2	-----0
2	1	-----0
	0	-----1

图 3.1 除 2 取余

0.69	
× 2	
1.38	
× 2	-----1
0.76	
× 2	-----0
1.52	
× 2	-----1
1.04	-----1

图 3.2 乘 2 取整

说明: ①用除 2 取余法对十进制整数进行转换时,先得到二进制数的低位,后得到高位;用乘 2 取整法对十进制小数进行转换时,先得到二进制数的高位,后得到低位。②一般来说,把一个带小数的十进制数转换为二进制数时,都是得到近似值,即一个实数在计算机内部存储的是二进制形式的近似值,所以当有实数参与运算时,结果也可能是一个近似值。



微课视频

3.2.2 二进制数与十六进制数的相互转换

1. 二进制数转换为十六进制数

(1) 计算方式: 先采用按权展开相加的方法把二进制数转换为十进制数, 再采用除 16 取余、乘 16 取整的方法把十进制数转换为十六进制数。

【例 3.3】 把二进制数 $(1101101.10111)_2$ 转换为十六进制数。

从例 3.1 可知, 二进制数 $(1101101.10111)_2$ 对应的十进制数是 $(109.71875)_{10}$ 。

把 109.71875 转换为十六进制数时, 整数部分和小数部分分别转换。把 109 转换为十六进制数时采用除 16 取余法, 算式如图 3.3 所示, 结果为 $(109)_{10} = (6D)_{16}$; 把 0.71875 转换为十六进制数时采用乘 16 取整法, 算式如图 3.4 所示, 结果为 $(0.71875)_{10} = (0.B8)_{16}$ 。

合并后的转换结果为 $(109.71875)_{10} = (6D.B8)_{16}$

$$\begin{array}{r|l} 16 & 109 \\ \hline 16 & 6 \quad \text{---D} \uparrow \\ & 0 \quad \text{---6} \downarrow \end{array}$$

图 3.3 除 16 取余

$$\begin{array}{r|l} & 0.71875 \\ \times & 16 \\ \hline & 11.50000 \quad \text{---B} \downarrow \\ \times & 16 \\ \hline & 8.00000 \quad \text{---8} \downarrow \end{array}$$

图 3.4 乘 16 取整

(2) 分组方式: 二进制数转换为十六进制数时, 以小数点为界, 分别向左、向右分成 4 位一组, 不够 4 位补 0, 分组后对应成十六进制数。

【例 3.4】 把二进制数 $(1101101.10111)_2$ 转换为十六进制数。

$$\begin{aligned} (1101101.10111)_2 &= (\underline{0110} \ \underline{1101} \ . \ \underline{1011} \ \underline{1000})_2 \\ &= (6D.B8)_{16} \end{aligned}$$

2. 十六进制数转换为二进制数

(1) 计算方式: 先采用按权展开相加的方法将十六进制数转换为十进制数, 再采用除 2 取余、乘 2 取整的方法将十进制数转换为二进制数。

【例 3.5】 把十六进制数 $(3AC.1E)_{16}$ 转换为二进制数。

$$\begin{aligned} \text{由于 } (3AC.1E)_{16} &= 3 \times 16^2 + 10 \times 16^1 + 12 \times 16^0 + 1 \times 16^{-1} + 14 \times 16^{-2} \\ &= 768 + 160 + 12 + 0.0625 + 0.0546875 \\ &= (940.1171875)_{10} \end{aligned}$$

再采用除 2 取余、乘 2 取整法把 $(940.1171875)_{10}$ 转换为二进制数, 得到的结果为 $(1110101100.0001111)_2$ 。

(2) 分组方式: 把十六进制数转换为二进制数时, 每一个十六进制位展开成 4 个二进制位即可。

【例 3.6】 把十六进制数 $(3AC.1E)_{16}$ 转换为二进制数。

$$\begin{aligned} (3AC.1E)_{16} &= (\underline{0011} \ \underline{1010} \ \underline{1100} \ . \ \underline{0001} \ \underline{1110})_2 \\ &= (1110101100.0001111)_2 \end{aligned}$$

说明: 在进行进制转换时, 计算方式比较适合编写程序实现, 分组方式比较适合手工完成。

3.3.2 机器数的编码

在确定了最高位用于表示机器数的符号之后,机器数通常有原码、反码和补码三种不同的具体表示形式。

【例 3.9】 分别给出十进制数+56 和-56 的原码、反码和补码,假定字长为 16 位。

通过除 2 取余运算,得到无符号十进制数 56 的二进制形式为 111000。

+56 的原码表示为 0000000000111000(正数的原码最高位为 0,数值位补足 15 位)。

-56 的原码表示为 1000000000111000(负数的原码最高位为 1,数值位补足 15 位)。

+56 的反码表示为 0000000000111000(正数的反码与其原码相同)。

-56 的反码表示为 111111111000111(负数的反码,在其原码的基础上,符号位不变,数值位按位取反)。

+56 的补码表示为 0000000000111000(正数的补码与其原码相同)。

-56 的补码表示为 111111111001000(负数的补码,在其反码的末位加 1)。

在计算机内部,一般用补码表示带符号数,这样就可以把减法运算转换为加法运算。

【例 3.10】 计算 $x=86-25$ 的值。

$x=86-25$ 可以转换为 $(x)_{\text{补码}}=(+86)_{\text{补码}}+(-25)_{\text{补码}}$ 进行计算,计算过程如下(假定字长为 8 位):

86 的二进制形式为 1010110,+86 的补码表示为 $(+86)_{\text{补码}}=01010110$;

25 的二进制形式为 11001,-25 的原码表示为 10011001,补码表示为 $(-25)_{\text{补码}}=11100111$;

$(x)_{\text{补码}}=01010110+11100111=00111101$,由于最高位为 0,因此 $(x)_{\text{原码}}=00111101$,其对应的十进制真值为 61,即 $x=61$,计算结果正确。

【例 3.11】 计算 $y=32-57$ 的值。

$y=32-57$ 可以转换为 $(y)_{\text{补码}}=(+32)_{\text{补码}}+(-57)_{\text{补码}}$ 进行计算,计算过程如下(假定字长为 8 位):

32 的二进制形式为 100000,其补码表示为 $(+32)_{\text{补码}}=00100000$;

57 的二进制形式为 111001,-57 的原码表示为 10111001,补码表示为 $(-57)_{\text{补码}}=11000111$;

$(y)_{\text{补码}}=00100000+111000111=11100111$,由于最高位为 1,因此 $(y)_{\text{原码}}=10011001$ (先对补码减 1,再对数值位按位取反),其对应的十进制真值为-25,即 $y=-25$,计算结果正确。

从以上示例可以看出,用补码表示带符号数,可以把减法运算转换为加法运算。我们知道,乘法运算可以转换为加法运算,除法运算可以转换为减法运算,这样 CPU 中只有一个加法器即可完成加、减、乘、除四则运算,即补码表示可以简化计算机的 CPU 设计。

3.3.3 机器数的表示范围

机器数的表示范围由计算机的字长决定,即由 CPU 中的寄存器决定。如果使用的是 16 位的寄存器,则字长为 16 位,一个无符号整数的最小值是 $(0000000000000000)_2=(0)_{10}$ 、最大值是 $(1111111111111111)_2=(65535)_{10}$,机器数的范围为 $0\sim 65\,535$ 。也就是说,对于一个 16 位的寄存器,只能表示 $0\sim 65\,535$ 的无符号整数。对于带符号整数,16 位寄存器的表示范围是 $-32\,768\sim +32\,767$ 。现在用的微型计算机一般是 64 位字长,其无符号整数的表

示范围是 $0 \sim 2^{64} - 1$, 带符号整数的表示范围是 $-2^{63} \sim +2^{63} - 1$ 。对于一般的应用场景足够了。

3.3.4 机器数中小数点的位置

在计算机内部表示小数点比较困难, 设计者把小数点的位置用隐含的方式表示。隐含的小数点位置可以是固定的, 也可以是变动的, 前者称为定点数, 后者称为浮点数。

1. 定点数

在定点数中, 小数点的位置一旦确定, 就不再改变。定点数又分为定点整数和定点小数。

小数点的位置约定在最低位的右边, 用来表示定点整数。小数点的位置约定在符号位(最高位)之后, 用来表示小于 1 的定点小数。

【例 3.12】 假定计算机的字长为 16 位, 用定点整数表示十进制数 387。

由于 387 的二进制形式为 $(110000011)_2$, 因此其在计算机内部的表示形式如图 3.6 所示。



图 3.6 计算机内部的定点整数

【例 3.13】 假定计算机的字长为 16 位, 用定点小数表示 0.625。

由于 0.625 的二进制形式为 $(0.101)_2$, 因此其在计算机内部的表示形式如图 3.7 所示。

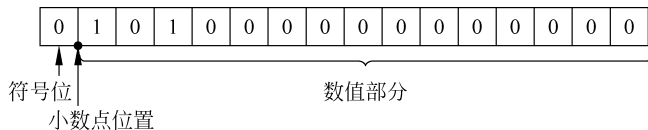


图 3.7 计算机内部的定点小数

2. 浮点数

如果要处理、存储的数据包括整数和小数, 则难以用定点数表示。对此设计人员设计了浮点数的表示方式, 即小数点位置不固定, 是浮动的。

将十进制数 758.2、-75.82、0.07582、-0.007582 用指数形式表示, 它们分别可以表示为 0.7582×10^3 、 -0.7582×10^2 、 0.7582×10^{-1} 、 -0.7582×10^{-2} 。

可以看出, 在原数据中无论小数点前后各有几位数, 它们都可以用一个纯小数(称为尾数, 有正负之分)与 10 的整数次幂(称为阶码, 也有正负之分)的乘积形式来表示, 这就是浮点数表示法。

同理, 一个二进制数 N 也可以表示为 $N = \pm S \times 2^{\pm P}$ 的形式。其中, N 、 S 、 P 均为二进制数。 S 称为 N 的尾数, 即全部的有效数字(数值小于 1 的纯小数), S 前面的 $\pm$ 是尾数的符号; P 称为 N 的阶码(通常是整数), 即指明小数点的实际位置, P 前面的 $\pm$ 是阶码的符号。

在计算机中一般浮点数的表示形式如图 3.8 所示。

在浮点数表示中, 尾数的符号和阶码的符号各占一位, 阶码是定点整数, 阶码的位数决定了所表示的浮点数的范围, 尾数是定点小数, 尾数的位数决定了浮点数的精度。阶码和尾

阶符	阶码 P	尾符	尾数 S
----	--------	----	--------

图 3.8 浮点数的表示形式

数都可以用补码表示。在字长有限的情况下,浮点数表示方法能扩大数的表示范围。

【例 3.14】 如果计算机的字长为 8 位,一个字长内,如果用定点整数形式,带符号数的表示范围为 $-128 \sim +127$ (十进制数)。如果用浮点数形式,可以表示十进制数的 $+1537$ 甚至更大的数。 $+1537$ 写成浮点数形式如下:

$$+1537 = (11000000001)_2 \approx 0.11 \times 2^{1011}$$

用一个 8 位字长表示,阶码数值位为 1011,符号位为 0,共 5 位;尾数为 11,符号位为 0,共 3 位。合起来就是 01011011B。

说明: ①由于字长的限制,尾数的数值位只有 2 位,因此原本的尾数 0.11000000001,只能表示成 0.11,所以 01011011B 是十进制数 $+1537$ 的近似表示。浮点数表示方法能扩大数的表示范围,但精度有所损失。合理确定尾数和阶码分别所占位数,既能扩大数的表示范围,又能保证一定的精度。②从上面的介绍可以看出,整数表示和浮点数表示在计算机内部的存储格式是不一样的,所以在程序设计语言里一般都要求,不同类型的数据是不能直接运算的,需要先转换为同一类型,再进行计算。

3.4 字符型数据的编码表示

计算机不仅能处理数值型数据,还能处理字符型数据,如英文字母、标点符号等。对于数值型数据,可以按照一定的转换规则将其转换为二进制数在计算机内部表示,但对于字符型数据,没有相应的转换规则可以使用。人们可以设计每个字符对应的二进制编码形式,但设计方案要科学、合理,才能得到业界广泛的认可和使用。当输入一个字符时,系统自动将输入的字符按规定的编码方式转换为相应的二进制形式存入计算机的存储单元中。在输出过程中,再由系统自动将二进制编码数据转换为用户可以识别的数据格式输出。

常用的字符型数据编码方式主要有 ASCII 码、EBCDIC 码等,前者主要用于微型计算机,后者主要用于超级计算机和大型计算机。

3.4.1 ASCII 码

目前微型计算机中使用最广泛的字符编码是 ASCII 码,即美国信息交换标准码(American Standard Code for Information Interchange)。ASCII 码包括 32 个通用控制字符(最左边两列)、10 个十进制数码、52 个英文大小写字母和 34 个专用符号(标点符号等),共 128 个符号,故需要用 7 位二进制数进行编码。通常使用一个字节(8 个二进制位)表示一个字符的 ASCII 码,规定其最高位总是 0,后 7 位为实际的 ASCII 码。ASCII 码如表 3.3 所示。

表 3.3 7 位 ASCII 码编码表

$b_4 b_3 b_2 b_1$	$b_7 b_6 b_5$							
	000	001	010	011	100	101	110	111
0000	NUL	DLE	SPACE	0	@	P	`	p



微课视频

续表

$b_4b_3b_2b_1$	$b_7b_6b_5$							
	000	001	010	011	100	101	110	111
0001	SOH	DC1	!	1	A	Q	a	q
0010	STX	DC2	"	2	B	R	b	r
0011	ETX	DC3	#	3	C	S	c	s
0100	EOT	DC4	\$	4	D	T	d	t
0101	ENO	NAK	%	5	E	U	e	u
0110	ACK	SYN	&	6	F	V	f	v
0111	BEL	ETB	'	7	G	W	g	w
1000	BS	CAN	(	8	H	X	h	x
1001	HT	EM	)	9	I	Y	i	y
1010	LF	SUB	*	:	J	Z	j	z
1011	VT	ESC	+	;	K	[	k	{
1100	FF	FS	,	<	L	\	l	
1101	CR	GS	-	=	M	}	m	}
1110	SO	RS	.	>	N	^	n	~
1111	SI	US	/	?	O	_	o	DEL

【例 3.15】 给出英文单词 Computer 的 ASCII 码。

查看表 3.3 可知,Computer 中各字符的二进制形式的 ASCII 码分别为 01000011、01101111、01101101、01110000、01110101、01110100、01100101、01110010,写成十六进制形式为 43 6F 6D 70 75 74 65 72,存储在计算机中占用 8 字节单元,即 1 个字符占用 1 字节单元。

3.4.2 EBCDIC 码

EBCDIC(extended binary coded decimal interchange code) 码是对 BCD 码的扩展,称为扩展 BCD 码。BCD(binary-coded decimal) 码又称“二-十进制编码”,用二进制编码形式表示十进制数。BCD 码的编码方法很多,有 8421 码、2421 码和 5211 码等。最常用的是 8421 码,其方法是用 4 位二进制数表示一位十进制数,自左至右每一位对应的位权分别是 8、4、2、1。4 位二进制数有 0000~1111 共 16 种状态,而十进制数只有 0~9 共 10 个数码,BCD 码只用 10 种状态: 0000~1001。8421 码如表 3.4 所示。由于 BCD 码中的 8421 码应用最广泛,因此一般说 BCD 码就是指 8421 码。

表 3.4 8421 码表

十进制数	8421 码	十进制数	8421 码
0	0000	8	1000
1	0001	9	1001
2	0010	10	0001 0000
3	0011	11	0001 0001
4	0100	12	0001 0010
5	0101	13	0001 0011
6	0110	14	0001 0100
7	0111	15	0001 0101

【例 3.16】 写出十进制数 7852 的 8421 BCD 码。

十进制数 7852 的 8421BCD 码为 0111100001010010B, 实际存储时可以占用 4 字节, 每字节的高 4 位补 0000, 低 4 位存储一位十进制数的 BCD 码, 7852 对应的 4 字节 BCD 码值为 00000111 00001000 00000101 00000010, 即一位十进制数字占用一字节, 称为非压缩 BCD 码; 也可以用 2 字节存储, 7852 对应的 2 字节 BCD 码值为 01111000 01010010, 即两位十进制数字占用一字节, 称为压缩 BCD 码。

IBM 公司于 1963—1964 年推出了 EBCDIC 码, 除了原有的 10 个数字之外, 又增加了一些特殊符号、大小写英文字母和某些控制字符的表示。所以, EBCDIC 码也是一种字符编码, 如表 3.5 所示, 主要用于大型计算机和超级计算机。

表 3.5 EBCDIC 码

高位	低 位															
	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0000	NUL	SCH	STX	ETX	PF	HT	LC	DEL		RLF	SMM	VT	FF	CR	SR	SI
0001	DLE	DC1	DC2	TM	RES	NL	BS	IL	CAN	EM	CC	CU1	IFS	IGS	IRS	IUS
0010	DS	SOS	FS		BYP	LF	ETB	ESC			SM	CU2		ENQ	ACK	BEL
0011			SYN		PN	RS	UC	EQT				GU3	DC4	NAK		SUB
0100	SP										[	.	<	(	+	!
0101	&										]	\$	*	)	;	~
0110	—	/										,	%	_	>	?
0111	-										:	#	@	·	=	“
1000		a	b	c	d	e	f	g	h	i						
1001		j	k	l	m	n	o	p	q	r						
1010		—	s	t	u	v	w	x	y	z						
1011																
1100	{	A	B	C	D	E	F	G	H	I						
1101	}	J	K	L	M	N	O	P	Q	R						
1110	\		S	T	U	V	W	X	Y	Z						
1111	0	1	2	3	4	5	6	7	8	9						

3.5 汉字的编码表示

汉字与英文字母类似, 也没有可用的转换规则直接转换为二进制形式, 也需要设计每个汉字对应的二进制编码, 用于汉字在计算机中的表示与存储。汉字有一些与英文字母不同的特点, 常用汉字的个数比较多, 不能直接对应到键盘上 (一个英文字母对应一个按键), 所以还要设计汉字的输入编码, 即每个汉字通过哪几个按键输入。输出汉字时, 还需要汉字的

字形码。即对于汉字来说,要设计用于输入汉字的输入码,用于存储汉字的机内码和用于输出汉字的字形码。

3.5.1 汉字输入码



微课视频

在用计算机处理汉字时,首先遇到的问题是输入汉字。汉字输入码是指从键盘输入汉字时采用的编码,又称为外码,主要有数字码、拼音码和字形码等。

(1) 数字码。常用的数字码是国标区位码,用数字串代表一个汉字的输入码。区位码是将国家标准局公布的 6763 个常用汉字分为 94 个区,每个区再分为 94 位,实际上是把汉字和一些特殊符号组织在一个 94 行 94 列的矩阵中,每一行称为一个区,每一列称为一个位,共有 94 个区(区号为 01~94),每个区内有 94 个位(位号也为 01~94)。一个汉字所在的区号和位号组合在一起就构成了该汉字的区位码(4 位十进制数字)。例如,“机”字位于第 27 区 90 位,区位码为 2790。

(2) 拼音码。拼音码是以汉语拼音为基础的输入码,如搜狗拼音输入法、微软拼音输入法和 QQ 拼音输入法等。

(3) 字形码。字形码是根据汉字的形状形成的输入码。汉字个数虽多,但组成汉字的基本笔画和基本结构并不多。因此,把汉字拆分成基本笔画和基本结构,按笔画或基本结构的顺序依次输入,就能表示一个汉字。五笔字型编码是最有影响的一种字形码方法。

数字码的优点是每个汉字都有一个唯一的数字编码,不足是记忆量太大,一般人很难掌握。拼音码易于学习和掌握,凡熟悉汉语拼音的人,不需训练和记忆即可使用,但由于重码率高(好多字的拼音相同),难以实现盲打,导致打字速度不容易提高。字形码的拆字规则(把一个字拆成基本笔画或基本结构,再对应到键盘的按键上)较复杂,学习起来较为困难,但其重码率很低,一旦学会并熟练掌握,能够实现盲打,因而有比较快的输入速度。专业打字人员使用字形码(五笔字型)比较多,一般人使用拼音码比较多。

为了提高汉字输入速度,在上述方法的基础上,发展了词组输入、联想输入等多种快速输入方法。另外的输入方式是利用语音或图像识别技术实现语音输入和扫描输入,自动将文字输入计算机中。键盘输入、语音输入和扫描输入各有其特点及适用场景。

3.5.2 汉字机内码

汉字机内码是指计算机内部存储和处理汉字时所用的编码,要求它与 ASCII 码兼容但又不能相同,以便实现汉字与英文的混合存储与处理。输入码经过键盘被计算机接收后就由有汉字处理功能的操作系统的“输入码转换模块”转换为机内码。机内码主要有国标码、Unicode 编码等。

1. 国标码

1980 年我国公布了 GB 2312—1980《信息交换用汉字编码字符集·基本集》,简称国标码,规定每个汉字编码由 2 字节构成,定义了 6763 个常用汉字和 682 个图形符号。

为了进一步满足信息处理的需要,在国标码的基础上,2000 年 3 月我国又推出了 GB 18030—2000《信息技术信息交换用汉字编码字符集基本集的扩充》,共收录 27 533 个汉字。GB 18030—2000 采用变长多字节编码,每个汉字编码可以由 2 字节或 4 字节组成,完全支持 Unicode 编码,对 GB 2312—1980 完全向后兼容。

2005 年发布的 GB 18030—2005《信息技术中文编码字符集》以汉字为主并包含多种我国少数民族文字,收录汉字 70 244 多个,相对于 GB 18030—2000,增加了 4 万多个汉字。2022 年发布的 GB 18030—2022《信息技术中文编码字符集》收录汉字 87 887 个,又增加了 1.7 万多个汉字。

通常将国标码的每个字节的最高位置 1 作为汉字的机内码,国标码由 2 字节或 4 字节表示一个汉字。由于英文符号的 ASCII 码的最高位为 0,而汉字符号的机内码的每字节的最高位都为 1,因此易于区分出某个字节数据表示的是一个英文字符,还是汉字字符的组成部分。

2. Unicode 编码

随着互联网的快速发展,需要满足跨语言、跨平台进行文本转换和处理的要求,还要与 ASCII 码兼容,因此,在 1994 年由统一码联盟发布了 Unicode 1.0,到 2022 年的最新版本是 Unicode 15.0。Unicode 编码(统一码、万国码、单一码)试图为全世界每种语言中的每个字符设定统一并且唯一的二进制编码。Unicode 编码的优点是包括了世界上所有语言的字符,但也有其不足。我们知道,英文字母只用 1 字节表示就够了,但如果用定长方式表示,每个符号的 Unicode 编码需要用 4 字节表示,那么每个英文字母的 4 字节编码中的 3 字节都是 0,这对于存储空间来说是很大的浪费,文本文件的大小会因此大二三倍。

Unicode 编码在很长一段时间内无法推广,直到互联网的出现。为解决 Unicode 编码字符如何在网络上传输的问题,于是面向网络传输的多种通用字符集传输格式(UCS transfer format,UTF)标准出现了,UCS 是 universal character set(通用字符集)的缩写形式。UTF-8 是在互联网上使用最为广泛的一种 Unicode 编码的实现方式,它每次可以传输 8 个数据位。变长编码方式是 UTF-8 的最大特点,它使用 1~4 字节表示一个符号,根据不同的符号而变化字节长度。当字符在 ASCII 码的范围时,就用 1 字节表示,保留了 ASCII 字符 1 字节的编码作为它的一部分。UTF-8 的一个中文字符占用 3 字节。从 Unicode 编码到 UTF-8 并不是直接对应的,要经过一些算法和规则的转换。

3.5.3 汉字字形码

汉字字形码又称汉字字模,是指汉字信息的输出编码,用于汉字的显示或打印机输出。汉字字形码有两种主要表示方式:点阵方式和矢量方式。汉字在计算机内部是以机内码的形式存储和处理的,当需要显示或打印这些汉字时,必须通过字形码将其转换为人们能看懂且能表示为各种字形字体的图形格式,然后通过输出设备输出。

1. 点阵字形码

不论一个字的笔画多少,都可以用一组点阵表示。每个点对应二进制的一位,由 0 和 1 表示不同状态,如黑白颜色等。一种字形码的全部汉字编码就构成字模库,简称字库。根据输出字符要求的不同,每个字符点阵中点的个数也不同。点阵越大,点数越多,输出的字形也就越清晰美观,占用的存储空间也就越大。汉字字形有 16×16 、 24×24 、 32×32 、 48×48 、 128×128 点阵等,不同字体的汉字需要不同的字库。点阵字库存储在文字发生器或字模存储器中。字模点阵的信息量是很大的,所占存储空间也很大。以早期显示用的 16×16 点阵为例,每个汉字的字形编码要占用 32 字节。打印一般用 24×24 点阵形式,每个汉字就要占用 72 字节。对于 128×128 点阵形式,每个汉字就要占用 2048 字节,将导致整个字库占用

大量的存储空间。图 3.9 所示是汉字“英”的点阵及对应编码。

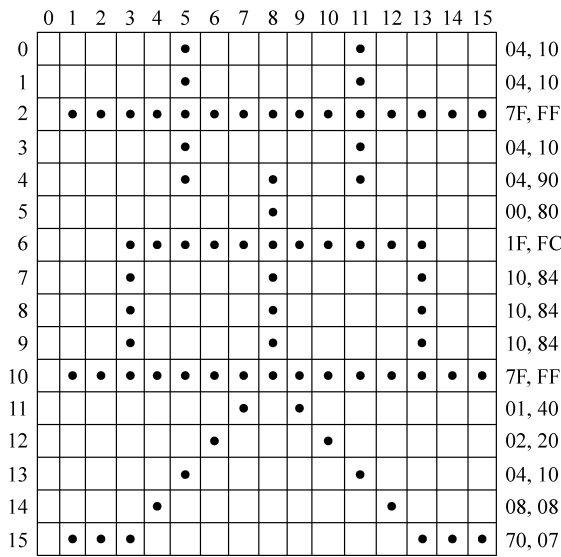


图 3.9 汉字“英”的点阵及对应编码

2. 矢量字形码

对于矢量方式的字形码,在计算机内存储的是一种用数学函数描述的曲线字库,采用了几何学中二次曲线及直线来描述字体的外形轮廓,含有字形构造、颜色填充、数字描述函数、流程条件控制、栅格处理控制、附加提示控制等指令。当要输出汉字时,通过计算机的计算,由汉字字形描述生成所需大小和形状的汉字。由于是用指令对字形进行描述,与分辨率无关,均以设备的分辨率输出,既可以屏幕显示,又可以打印输出,字符缩放时总是光滑的,不会有锯齿出现,因此可产生高质量的汉字输出。

点阵方式和矢量方式各有特点。前者编码、存储方式简单,无须转换直接输出,字号变大后显示或打印效果变差,出现锯齿笔画甚至模糊不清;后者输出时需要进行转换,输出过程复杂一些,但字号变大后不会降低显示或打印质量。点阵汉字和矢量汉字放大后的输出效果对比如图 3.10 所示。

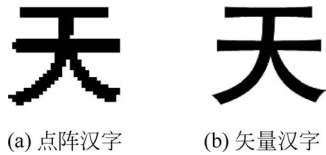


图 3.10 点阵汉字与矢量汉字放大后的输出效果对比

汉字通常通过输入码输入计算机内,再由汉字系统的输入管理模块通过查表或计算,将输入码(外码)转换为机内码存入计算机存储器中,对汉字的处理也是以机内码形式进行的。当存储在计算机内的汉字需要在屏幕上显示或在打印机上输出时,要借助汉字机内码在字模库中找出对应的字形码,在输出设备上将该汉字的图形信息显示或打印出来。

3.6 图像与声音数据的采集与表示

计算机能够存储和处理的信息,除了前面介绍的数值和文本(如英文字符、汉字等)外,还有图形、图像、音频、视频、动画等多媒体信息,而且随着计算机技术、网络技术的快速发展和应用的普及,生动形象、深受人们喜欢的音/视频信息越来越多。



微课视频

图像和声音都不能通过键盘输入计算机,而是通过信息采集设备(如数码相机、带摄像头的手机、录音笔、扫描仪等)变为二进制信息并存储的。

3.6.1 图像数据的采集与表示

1. 图形

图形(graphic)一般指用计算机绘制的几何形状,如直线、圆、圆弧、矩形、任意曲线和图表等,也称矢量图。

图形以矢量图形文件的形式存储,计算机中存储的是生成图形的指令,所以一般不需要对矢量图进行编码。计算机中常用的矢量图形文件有.3ds文件(用于三维造型)、.dxf文件(用于计算机辅助设计)、.wmf文件(用于桌面出版)等。

2. 图像

图像(image)又称位图(bitmap)或图片,是指由输入设备捕捉的实际场景画面,或以数字化形式存储的任意画面,如用手机拍摄的数码照片、扫描仪扫描的图片等。手机摄像头或扫描仪把图像点阵化,每个点称为像素(pixel)。类似于汉字的点阵,不同的是分辨率更高一些,目前一般都在 1024×1024 像素以上,采集的值为像素点的颜色,对于彩色图像,一般用24位二进制数表示一个像素点的颜色值,一共可以表示 2^{24} (16 777 216)种不同的颜色,基本可以把人眼能识别的所有不同颜色表达出来,称为真彩色。BMP格式的图像文件保存原始像素数据,即每个像素点的颜色值。对于一幅分辨率为 1024×1024 像素的真彩色图片,其BMP文件的大小为3MB。JPG格式为一种压缩格式。除BMP、JPG格式外,图像文件还有GIF、PNG、PCX、TIF、TGA等格式。

3. 视频

视频是视频图像(video)的简称,是一种活动影像,它与电影和电视原理是一样的,都是利用人眼的视觉暂留特征,将足够多的静态画面(frame,帧)连续播放,只要能够达到每秒24帧以上,人的眼睛就察觉不出画面之间的不连续性。视频文件在计算机中的存储格式有AVI、WMV、MPEG、ASF、RM、MOV等。

视频文件存储的是视频所包含的所有图像的数据。

4. 动画

动画(animation)也是一种活动影像,最典型的是“卡通”片。动画也称为动漫,它与视频的不同在于视频一般是指对自然界真实影像的记录,如用摄像机拍摄下来的自然风光,而动画通常指人工制作出来的连续图像或图形所组合成的动态影像。动画文件存储的是动画所包含的所有(“卡通”)图像的数据。

3.6.2 声音数据的采集与表示

声音也称为音频(audio),声音(声波)是通过空气传播的一种连续的波。声音有多种,如人说话、唱歌的声音,演奏乐器的声音,刮风、下雨、打雷的声音等。声音一般用一种模拟的连续波形表示,如图3.11所示。声波可以用振幅和频率两个参数来描述,振幅的大小反映了声音的强弱,用分贝(dB)表示;频率是指声音信号每秒变化的次数,频率的大小反映了音调的高低,用赫兹(Hz)表示。

对声音的数字化就是对声波采样和量化,即按一定的时间间隔对声波的振幅进行测量

并转换为二进制数,如图 3.12 所示。

采样频率越高(采样时间间隔越小)、采样精度越高(表示振幅值的二进制位数越多),就越能真实地把声音记录下来(播放时质量高),当然,需要的存储容量也就越大。

例如,每秒采样 40 000 次,用 16 个二进制位表示采样值,可以把采样值量化为 65 536 个级别,采样一秒钟产生的数据为: $40\,000 \times 16/8\text{B} = 80\,000\text{B} = 78.125\text{KB}$ 。

音频文件在计算机中的存储格式有 MP3、WAV、MIDI、WMA 等。

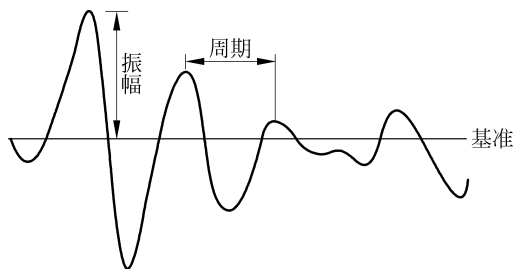


图 3.11 声音波形

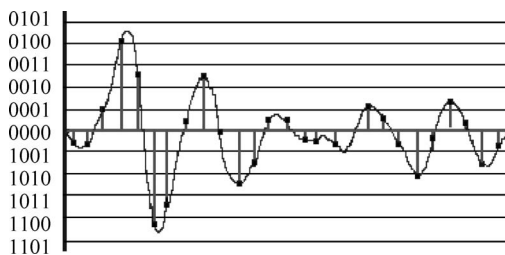


图 3.12 声音的采样和量化

3.7 小 结

计算机能够处理数值、文本、图形、图像、视频、动画和音频等多种媒体信息,虽然展现的这些信息千姿百态、异彩纷呈,但所有信息在计算机内部都是以二进制数据形式存储的。

数值、文本、图形、图像、视频、动画和音频以不同的方式转换为二进制形式。数值按一定的规则转换为对应的二进制数,文本(如英文字符、汉字等)按设计好的编码规则得到对应的二进制数(如 ASCII 码、EBCDIC 码、汉字国标码、Unicode 编码等),图形对应的是生成图形的指令,图像和声音通过采集设备获取对应的二进制数,图像对应的是点阵数据,声音得到的是声波的量化数据。作为计算机专业的学生,明晰各种信息到二进制数据的转换与存储,了解数值的符号与小数点如何在计算机内部表示,对于深入理解计算机的工作原理、深入理解程序设计时的一些要求都是非常有益的。

拓展阅读: Intel 公司与 CPU

Intel(英特尔)公司成立于 1968 年,名字取自两个英文单词 Integrated(集成)和 Electronic(电子)的组合,当时公司只有 8 个人。罗伯特·诺伊斯(R. Noyce,1927—1990)、戈登·摩尔(G. Moore,1929—)和安迪·葛洛夫(A. Grove,1936—)是公司的主要创始人并领导公司在微处理器领域取得了辉煌的成就。信息技术(information technology, IT)领域著名的摩尔定律就是由摩尔提出的,定律的内容是“集成电路上可容纳的晶体管个数,约每隔 18 个月便会增加一倍,性能也将提升一倍”。

Intel 公司的早期产品主要是存储器,用半导体存储器取代了磁芯存储器,大大提高了存储容量和数据存取速度。1971 年,当时还处在早期发展阶段的 Intel 公司推出了世界上第一枚微处理器 4004,这是第一个用于计算器的 4 位微处理器。4004 含有 2300 个晶体管,

现在看来功能有限,速度也不快。但是,辉煌成就就是从这开始的,从此以后,Intel 公司逐步占据微处理器市场的霸主地位。1972 年推出 8 位的 8008,1974 年推出比 8008 更先进的 8080。

1978 年,首次生产出名为 8086 的 16 位微处理器,同时生产了与之相配合的数字协处理器 8087,在 8087 指令集中增加了一些专门用于对数、指数和三角函数等数学计算的指令,以提高数学运算的速度。由于这些指令集应用于 8086 和 8087,因此人们也把这些指令集统一称为 x86 指令集。虽然以后 Intel 公司又陆续生产出第二代、第三代等若干代更先进和更快速的新型微处理器,但都兼容原来的 x86 指令,而且 Intel 公司在后续微处理器的命名上沿用了原先的 x86 序列,直到后来因商标注册问题,才放弃了继续使用阿拉伯数字命名的做法,新起名为“奔腾”(Pentium)。

1979 年,Intel 公司推出了 8088 芯片,属于准 16 位微处理器,内部数据总线是 16 位,外部数据总线是 8 位,内含 29 000 个晶体管,时钟频率为 4.77MHz,地址总线为 20 位,可使用 1MB 内存。

1981 年,8088 芯片首次用于 IBM PC 中,使微型计算机的发展进入了一个新的时代。也正是从 8088 开始,PC 的概念开始在全世界范围内流行起来。

1982 年,Intel 公司推出了新产品 80286 芯片,该芯片比 8086 和 8088 都有很大的改进,虽然它仍旧是 16 位结构,但是在 CPU 的内部含有 13.4 万个晶体管,时钟频率由最初的 6MHz 逐步提高到 20MHz。其内部和外部数据总线均为 16 位,地址总线为 24 位,寻址能力扩大到 16MB 内存。从 80286 开始,CPU 的工作方式也变成实模式和保护模式两种方式。

1985 年,Intel 公司推出了 80386 芯片,它是 80x86 系列中的第一款 32 位微处理器,而且制造工艺也有了很大的进步,80386 内部集成有 27.5 万个晶体管,时钟频率为 12.5MHz,后提高到 20MHz、25MHz、33MHz 几个档次。80386 的内部和外部数据总线都是 32 位,地址总线也是 32 位,可寻址高达 4GB 的内存。它除具有实模式和保护模式外,还增加了一种叫虚拟 86 的工作方式,可以通过同时模拟多个 8086 处理器来提供多任务处理能力。

1989 年,Intel 公司推出 80486 芯片,80486 是将 80386 和数字协处理器 80387 以及一个 8KB 的高速缓存(cache)集成在一个 CPU 芯片内,这种芯片集成了 120 万个晶体管。80486 的时钟频率从 25MHz 逐步提高到 33MHz、50MHz,并且在 80x86 系列中首次采用了精简指令集计算机(reduced instruction set computing,RISC)技术,可以在一个时钟周期内执行一条指令。它还采用了突发总线方式,大大提高了与内存的数据交换速度。由于这些改进,80486 的性能比带有 80387 数字协处理器的 80386 提高了 4 倍。

Intel 公司于 1993 年推出了全新一代的高性能处理器 80586。由于 CPU 市场的竞争越来越激烈,提出了商标注册问题,但美国法律不允许用阿拉伯数字注册商标,于是 Intel 公司用新的名字——Pentium 注册了商标。Pentium 中的 Pent 来自希腊语,意思是 5,ium 使芯片看起来像基本元素(确实是组成计算机系统的基本元素)。Intel 公司还给它起了一个非常好听的中文名字——“奔腾”。Pentium 内部集成的晶体管数量高达 310 万个,时钟频率由最初的 60MHz 和 66MHz,逐步提高到 200MHz。66MHz 的 Pentium 比 33MHz 的 80486 要快 3 倍多。

1995 年,Intel 公司推出了第 6 代 x86 系列 CPU——Pentium Pro。Pentium Pro 内部

集成有 550 万个晶体管,内部时钟频率为 133MHz,处理速度大约是 100MHz 的 Pentium 的 2 倍。Pentium Pro 的一级(片内)缓存为 8KB 指令和 8KB 数据。在 Pentium Pro 的一个封装中除 Pentium Pro 芯片外,还包括有一个 256KB 的二级缓存芯片。

1996 年,Intel 公司又推出了 Pentium 系列的改进版本——多能奔腾(Pentium MMX)。MMX 技术是 Intel 公司发明的一项多媒体增强指令集技术,为 CPU 增加了 57 条 MMX 指令,还将 CPU 芯片内的一级缓存由原来的 16KB 增加到 32KB(16KB 指令+16KB 数据),MMX CPU 比普通 CPU 在运行含有 MMX 指令的程序时,处理多媒体的能力提高了 60% 左右。

1997 年 5 月,Intel 公司又推出了 Pentium II。Pentium II 采用了与 Pentium Pro 相同的核心结构,但它加快了段寄存器写操作的速度,并增加了 MMX 指令集,以加快 16 位操作系统的执行速度。Pentium II 比 Pentium Pro 多集成了 200 万个晶体管。

1998 年,Intel 公司推出了面向低端市场、性能价格比较高的赛扬处理器(Celeron CPU)。该处理器去掉了二级缓存,因而可以降低处理器的成本。

1998—1999 年,Intel 公司还推出一款比 Pentium II 功能还要强大的至强处理器(Xeon CPU)。至强处理器的目标就是挑战高端的、基于 RISC 的工作站和服务器。赛扬处理器和至强处理器的推出就是要全方位占领微处理器市场,满足各种不同用户的需要。

Pentium III 处理器是 Intel 公司的又一代产品,内部集成了 960 万个晶体管,拥有 32KB 一级缓存和 512KB 二级缓存,增加了能够增强音频、视频和三维图形效果的数据流单指令多数据扩展(streaming SIMD extensions,SSE)指令集,和 Pentium II Xeon 一样,Intel 公司也推出了面向服务器和工作站系统的高性能 CPU——Pentium III Xeon。最初发布的 Pentium III 有 450MHz 和 500MHz 两种规格。

1999 年 10 月底,Intel 公司正式发布代号为 Coppermine 的新一代 Pentium III 处理器,CPU 主频最高达到 733MHz。Coppermine 采用全新的核心设计,内置 256KB 与 CPU 主频同步运行的二级缓存,新一代的 Coppermine 处理器的集成度大为提高,它的核心集成了 2800 万个晶体管。CPU 芯片可以做得更小,从而使芯片面积更小,功耗大为减小,成本也得以降低。所以,该款 CPU 适用于笔记本计算机使用。

2000 年 11 月,Intel 公司推出了功能更为强大的 Pentium 4。Pentium 4 采用了 Intel 的 NetBurst 技术,与 Pentium III 相比,Pentium 4 体系结构的流水线深度增加了一倍,达到了 20 级,集成了 4200 万个晶体管,时钟频率高达 1.4GHz 和 1.5GHz。改进的浮点运算功能使 Pentium 4 提供更加逼真的视频和三维图形,带来更加精彩的游戏和多媒体享受。算术逻辑单元以双倍的时钟速度运行,从而提高了总体运算速度。

2003 年,Intel 公司发布了专门用于移动运算的 Pentium M 处理器。Pentium M 处理器结合了 855 芯片组与 Intel PRO/Wireless 2100 网络联机技术,成为 Centrino(迅驰)移动运算技术的最重要组成部分。

2005 年,Intel 公司推出了双核心处理器 Pentium D 和 Pentium Extreme Edition,同时推出 945/955/965/975 芯片组来支持新推出的双核心处理器。

2006 年 7 月,Intel 公司发布了酷睿 2(Core 2 Duo),酷睿 2 是一个跨平台的构架体系,包括服务器版、桌面版、移动版三大领域。

2010 年 6 月,Intel 公司发布了 Core i3/i5/i7,Core i3/i5/i7 基于全新的 sandy bridge 微

架构,具有更低的功耗、更高的性能、更强的浮点运算与加密解密运算能力。2012年4月, Intel公司发布了ivy bridge(IVB)处理器。2013年6月,Haswell CPU问世。2017年5月, Intel公司发布了Core i9处理器,Core i9最多可包含18个内核,主要面向游戏和高性能需求。2023年,Intel公司推出了第13代Core i5、i7、i9 CPU,13代Core i5-13600KF CPU拥有14核心20线程,睿频最高可达5.1GHz,24MB的L3和20MB L2高速缓存。13代Core i7-13700KF CPU拥有16核心24线程,睿频最高可达5.4GHz,30MB的L3和24MB L2高速缓存。13代Core i9-13980HX CPU拥有24个内核,包括8个性能核和16个能效核,性能核的睿频可达5.6GHz,能效核的睿频可达4GHz。

在高性能的微处理器研发中使用了超线程技术和多核技术。超线程技术(hyper threading technology)就是利用特殊的硬件指令,把处理器内部的两个逻辑内核模拟成两个物理芯片,从而使单个处理器能够使用线程级的并行计算。多线程技术可以在支持多线程的操作系统和软件上,有效地增强处理器在多任务、多线程处理上的处理能力。对于单线程芯片来说,虽然也可以每秒处理成千上万条指令,但是在某一时刻,只能对一条指令(单线程)进行处理,而超线程技术则可以使处理器在某一时刻,同步并行处理更多指令和数据。多核技术指在一个处理器上集成多个运算核心,从而提高处理器的计算能力。

Intel公司除了传统的微处理器、芯片组、板卡市场外,近几年在向可穿戴设备、人工智能领域拓展,2023年营业收入为542亿美元。

习 题 3

一、填空题

1. 在计算机内部用_____进制存储程序和数据。
2. 对于十六进制,所用数码除了0~9外,还有_____。
3. 用后缀区分不同进制时,二、十、八、十六进制的后缀分别为_____,_____,_____,_____。
4. 把一个十进制数转换为二进制数,整数部分采用_____,小数部分采用_____。
5. 在计算机内部将数值型数据(真值)全面、完整地表示成一个二进制数(机器数),需要考虑4个因素:_____,_____,_____,_____。
6. 在计算机内部表示浮点数,包括4部分:_____,_____,_____,_____。
7. ASCII码共包括_____个符号。
8. ASCII码主要用于_____,EBCDIC码主要用于_____和_____。
9. 用计算机处理汉字,一般要考虑_____码、_____码、_____码的设计。
10. 汉字的字形码分为_____字形码、_____字形码。
11. 影响电子图片数据量大小的主要因素有_____和_____。
12. 影响采集声音数据量大小的主要因素有_____和_____。

二、计算题

1. 分别将下列数值转换为二进制数(小数点后保留4位)。
(216.73)₁₀ (7563.42)₈ (1A4E.3B)₁₆
2. 分别将下列二进制数转换为十进制数、八进制数和十六进制数。

$$(101101111.10111)_2 \quad (10110110.110111)_2$$

3. 分别将下列十进制数转换为八进制数和十六进制数。

$$(175.25)_{10} \quad (357)_{10}$$

4. 字长为8位时,分别求 $(+62)_{10}$ 和 $(-62)_{10}$ 的原码、反码和补码。

5. 计算十进制数 $37-59$ 的值,先把十进制数转换为二进制数,然后用补码计算,并体会补码的作用。

三、简答题

1. 简要说明汉字的输入码、机内码和输出码的作用。
2. 对比说明数值、英文字符、汉字、图像和声音是如何转换为二进制数据的。

思考题3

1. 对比说明带符号整数和实数在计算机内部的不同存储格式。
2. 既要表示世界上所有的语言文字,又要尽可能少占用存储空间,Unicode编码是如何处理这个关系的?