[内部资料]

全国计算机应用软件人员水平考试辅导丛书

微型计算机操作员录入员培训教材

刘尊全 主 编

清华大学出版社

目　　录

011

第一部分　计算机基础知识

第二部分　中西文键盘录入技术

第八章　五笔字型汉字输入技术介绍

第九章　汉字 WORDSTAR 使用介绍

第一部分　计算机基础知识

第一章　概　　述

第一节　计算机的发展及其特点

自从第一台电子计算机问世以来，在四十多年的时间里，计算机的发展迅猛。今天，计算机科学已作为一门先进的学科独立存在；在工业部门，已形成独立的计算机工业体系。计算机的广泛应用已成为现代化的一个重要标志。但是，电子计算机并不是神秘的东西，它是人类生产实践和科学技术发展的必然产物。只要我们勇于去认识和学习，就一定能掌握电子计算机技术。

一、电子计算机的发展

人类在长期的劳动生产中，很早就创造和使用了各种计算工具。例如我国从唐宋时代开始流传至今的算盘，1642年法国制成的第一台机械计算机，十七世纪出现的计算尺，1887年制成的手摇计算机以及随着电的发明产生的电动齿轮计算机等都是计算工具。现代的电子计算机就是上述这些计算工具的继承和发展，至今它还在随着科学技术日新月异的发展变化，不断地更新换代。

按照通常的划分，自从1946年美国制成的第一台电子计算机ENIAC（Electronic Numenical Integrator And Calculator，电子数字积分机和计算机）以来，大体上经历了四代。

1．第一代电子计算机（1946～1957年）

第一代电子计算机的逻辑元件（指执行一个逻辑功能的装置）采用电子管，主存储器采用延迟线或磁鼓（磁鼓是一种磁记录设备，它是一个高速旋转的鼓形圆筒，表面涂上磁性材料，根据每一点的磁化方向，确定这一点的信息），辅助存储器开始用磁带机，一切操作都由中央处理机集中控制。

第一代电子计算机虽然因采用电子管而体积大、耗电多、运算速度较低，但它却奠定了计算机发展的技术基础。

2．第二代电子计算机（1958～1964年）

第二代电子计算机比第一代有很大改进，其主要特点是：

(1) 逻辑元件采用晶体管。

由于晶体管比电子管平均寿命高100到1000倍，耗电却只有电子管的十分之一，体积比电子管小一个数量级，机械强度较高等优点，所以很快地晶体管电子计算机代替了电子管计算

机，并开始成批生产。

(2) 主存储器以磁芯存储器为主，辅助存储器开始使用磁盘。

所谓磁芯是用铁氧化合物制成的直径不到一毫米的小圆环，每个磁芯可以记录一位零或1。由于磁芯价格比磁鼓便宜，工作稳定，用它组成的磁芯存储器具有速度快、成本低、非易失性能好等优点，所以人们在第二代、第三代计算机中以采用磁芯存储器为主。

(3) 软件开始使用高级程序设计语言，如FORTRAN、COBOL、BASIC等，并有了操作系统。

(4) 改革了以中央控制器为中心的集中控制方式，利用通道管理输入、输出设备。

通道和主机的控制器独立并行工作，分别与内存交换信号，从而使高速的控制器和慢速的输入输出设备分开，提高了计算机的工作效率。

总之，第二代电子计算机的性能和可靠性都比第一代提高了许多，在结构上向通用型方向发展。

3. 第三代电子计算机（1965～1971年）

第三代电子计算机的主要标志是逻辑元件采用集成电路。这种电路器件就是把几十个或几百个一个个分开的电子元件集中做在一块几平方毫米的芯片上（一般称为集成电路板），使计算机的体积和耗电大大减少，性能和稳定性进一步提高。

第三代电子计算机发展速度很快，主存储器在磁芯存储器的基础上出现了更可靠的半导体存储器。机种开始多样化、系列化。外部设备不断增加，品种繁多，尤其是终端设备和远程终端设备迅速发展，并与通信设备结合起来。高级程序设计语言发展很快，操作系统进一步发展和完善。这样就使得第三代电子计算机在存储器容量、运算速度、可靠性等方面较第二代又提高了一个数量级。

4. 第四代电子计算机（1972年以来）

第四代电子计算机是以采用大规模集成电路为标志的。按通常的划分标准，每个硅片上门电路数量在10个以下的，称为小规模集成电路；门电路数在10个以上、100个以下的称为中规模集成电路；门电路数在1000个以上到几千个的称为大规模集成电路。

在这里我们要特别强调一下微型计算机。微型电子计算机（简称微型机）是1971年出现的，它由一片或几片大规模集成电路组成，存储设备大部分使用磁盘。微型机具有体积小、重量轻、功耗小、可靠性高、使用环境要求不严格、价格低廉、易于成批生产等特点，所以自出现以来，每二～三年就有一个重大的发展。目前有的微型机的功能不亚于一台小型计算机。在本书中我们将以介绍IBM PC微型电子计算机为主，向读者介绍电子计算机的基础知识，电子计算机键盘录入技术，BASIC语言及数据库管理系统等知识。

二、电子计算机的特点

电子计算机作为一种计算工具，与以往的计算工具相比较，有以下几个方面的特点：

(1) 运算速度快。

现代的巨型机已达每秒运算几亿次。大量复杂的科学计算过去需要几十年，现在用计算机只需要几个月、几天。如气象预报用手摇计算器要算一、二个星期，用一般中型电子计算机只要几分钟就完成了。

(2) 精确度高。

由于计算机内采用二进制数字进行运算，使得其计算精度可用增加表示数字的设备来获

此为试读，需要完整PDF请访问：www.ertongbook.com

得，再加上运用计算技巧，使得数值计算越来越精确。过去对圆周率π，数学家们经过艰苦的努力只能算到小数点后500多位。1981年，一位日本人利用计算机很快就算到小数点后200万位。

(3) 具有“记忆”功能和逻辑判断功能。

电子计算机有存储器，可以存储大量的数据。随着存储器容量的增大，计算机可以存储“记忆”的信息量也越来越大。电子计算机既可以进行算术运算又可以进行逻辑运算，它可以对文字、符号、大小、异同等进行判断和比较，利用计算机可以进行逻辑推理和证明，从而极大地扩大了计算机的应用范围。

(4) 具有自动运行能力。

电子计算机内部操作运算是根据人们事先编制的程序自动控制进行的，不需要人工干预。正因为如此计算机在现代科学中才越发显得重要。

第二节　计算机运算基础

电子计算机是对数据信息进行高速自动化处理的机器。这些数据信息是以数字、字符、符号、表达式体现的，它们以二进制编码形式与机器的电子元件状态相对应。因此要了解计算机如何工作，了解计算机中文字符号和数字的编码等，就需要了解二进制数字系统及和其它数制之间的关系。

一、进位计数制

1．进位计数制

按进位的方法进行计数，称为进位计数制。在日常生活中我们天天会遇到进位计数制的问题。例如通常从小学就学起的算术，逢十进一，是十进制。看钟表一小时等于六十分，一分钟等于六十秒，是六十进制。中国旧制市秤十六两为一斤，是十六进制。一打等于十二双，是十二进制等等。在电子计算机中，常用的是二进制、八进制、十六进制数。其中二进制使用更为广泛。

2．进位计数制的表示方法

进位计数制很多，我们这里主要介绍十进制、二进制、十六进制。这三种计数制的通常表示方法如表1-1所示。

表 1-1　十进制、二进制、十六进制数常用表示法

十进制	二进制	十六进制	十进制	二进制	十六进制
0	0000	0	9	1001	9
1	0001	1	10	1010	A
2	0010	2	11	1011	B
3	0011	3	12	1100	C
4	0100	4	13	1101	D
5	0101	5	14	1110	E
6	0110	6	15	1111	F
7	0111	7	16	10000	10
8	1000	8			

从表1-1中可以看出，十进制数具有两个特点：

① 有十个不同的数字符号 0，1，2，3，4，5，6，7，8，9。

② 每一个数字符号根据它在这个数中所处的位置(数位)按逢十进一来决定实际数值。

例如 $(999.99)_{10}$ 以小数点为界，从小数点往左是个位、十位、百位，从小数点往右是十分位、百分位。因此小数点左边第一位9代表数值9，即 9×10^0；小数点左边第二位9代表90，即 9×10^1；小数点左边第三位9代表900，即 9×10^2；小数点右边第一位9代表0.9，即 9×10^{-1}；小数点右边第二位9代表0.09，即 9×10^{-2}。所以这个数可以表示为：

$$(999.99)_{10}=9\times10^2+9\times10^1+9\times10^0+9\times10^{-1}+9\times10^{-2}$$

通常，任意一个十进制数正数D可以表示为：

$$(D)_{10}=D_{n-1}\times10^{n-1}+D_{n-2}\times10^{n-2}+\cdots+D_1\times10^1+D_0\times10^0+D_{-1}\times10^{-1}+D_{-2}\times10^{-2}+\cdots+D_{-m}\times10^{-m}$$

$$=\sum_{i=n-1}^{-m}D_i\times10^i$$

其中i表示数的某一位，D_i表示第i位的数码，可以是0，1，…，9中任何一个，m、n为正整数，10是十进制的基数。10^{-m}，10^{-m+1}，…，10^{-1}，10^0，10^1，…，10^{n-1}是十进制的权。

基数是该计数制中数字符号状态的个数，基数为10就是十进制数，基数是2就是二进制数。

在按位计数中，为确定一个数位的实际数值，必须乘上一个因子。这个因子是权。例如：

…，10^{-m}，10^{-m+1}，…，10^{-1}，10^0，10^1，…，10^{n-1}，…是十进制的权。

…，2^{-m}，2^{-m+1}，…，2^{-1}，2^0，2^1，…，2^{n-1}，…是二进制的权。

…，16^{-m}，16^{-m+1}，…，16^{-1}，16^0，16^1，…，16^{n-1}，…是十六进制的权。

表1-2 十进制、二进制、十六进制数位权

数位	十进制权	二进制权	十六进制权
N_0	$1=10^0$	$1=2^0$	$1=16^0$
N_1	$10=10^1$	$2=2^1$	$16=16^1$
N_2	$100=10^2$	$4=2^2$	$256=16^2$
N_3	$1000=10^3$	$8=2^3$	$4096=16^3$
N_4	$10000=10^4$	$16=2^4$	$65536=16^4$
N_{n-1}	10^{n-1}	2^{n-1}	16^{n-1}

从表1-2中可以看出若设一个数为 $L=L_{n-1}L_{n-2}\cdots L_1L_0L_{-1}\cdots L_{-m}$(m、n为正整数)则各进位制中权的值恰巧是基数的某次幂。

从表1-1可知二进制数具有两个特点：

① 它有两个不同的数字符号：0，1。

② 每一个数字符号根据它在这个数中的数位按逢二进一来决定其实际数值。

例如 $(1001)_2=1\times2^3+0\times2^2+0\times2^1+1\times2^0=(8+1)_{10}=(9)_{10}$

$(11011.101)_2=1\times2^4+1\times2^3+1\times2^1+1\times2^0+1\times2^{-1}+1\times2^{-3}=(27.625)_{10}$

通常，任一个二进制数B可以表示为：

$(B)_2 = B_{n-1}\times 2^{n-1} + B_{n-2}\times 2^{n-2} + \cdots + B_1\times 2^1 + B_0\times 2^0 + B_{-1}\times 2^{-1} + \cdots + B_{-m}\times 2^{-m}$

其中 B_i 取 0 或 1，m、n 为十进制正整数，2 是基数，2^{n-1}，2^{n-2}，…，2^{-m} 是权。

从表1-1中可知十六进制数具有两个特点：

① 它有十六个不同的数字符号：0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F。

② 每一个数字符号根据它在这个数中的位置按逢十六进一来决定其实际数值。

例如 $(327)_{16} = 3\times 16^2 + 2\times 16^1 + 7\times 16^0 = (807)_{10}$

$(3AB.11)_{16} = 3\times 16^2 + A\times 16^1 + B\times 16^0 + 1\times 16^{-1} + 1\times 16^{-2} = (939.0664)_{10}$

通常，任意一个十六进制数 H 可以表示为：

$(H)_{16} = H_{n-1}\times 16^{n-1} + H_{n-2}\times 16^{n-2} + \cdots + H_1\times 16^1 + H_0\times 16^0 + H_{-1}\times 16^{-1} + \cdots + H_{-m}\times 16^{-m}$

其中 H_i 取 0～F 中任意一个，m、n为十进制正整数，16 为基数，16^{n-1}，16^{n-2}，…，16^{-m} 是权。

总结以上三种计数制，可以把它们的特点概括为：

① 每一种计数制都有一个固定的基数 J(J 为大于 1 的整数)，它的每一位可取 J 个不同的数值；

② 每一种计数制都有自己的“权”，并且符合逢 J 进一的原则。

通常，对任意进位计数制，基数用正整数 J 表示，N_i 表示各位数字符号，n，m为正整数，则对 J 进制数 N 可以表示为：

$$N = \pm(N_{n-1}J^{N-1} + N_{n-2}J^{N-2} + \cdots + N_1J^1 + N_0J^0 + N_{-1}J^{-1} + \cdots + N_{-m}J^{-m}$$

$$= \pm\sum_{i=n-1}^{-m} N_iJ^i$$

二、数制间转换

数制间的转换，实质上是基数间的转换，一般转换依据的原则是：

如果两个有理数相等，则两数的整数部分和小数部分一定分别相等。因此进行各数制间转换时，一般把整数部分和小数部分分别按转换方法进行转换。

1．二进制数或十六进制数转换成十进制数

把二进制数或十六进制数转换成十进制数通常采用按权相加法，即把二进制数(或十六进制数)写成 2(或 16) 的各次幂的和的形式，然后按十进制计算其结果。

例：把下列各数转换成十进制数：

① $(1011)_2$ ② $(111.101)_2$ ③ $(3AF)_{16}$

解：① $(1011)_2 = 1\times 2^3 + 0\times 2^2 + 1\times 2^1 + 1\times 2^0 = 8+2+1 = (11)_{10}$

② $(111.101)_2 = 1\times 2^2 + 1\times 2^1 + 1\times 2^0 + 1\times 2^{-1} + 0\times 2^{-2} + 1\times 2^{-3} = (7.625)_{10}$

③ $(3AF)_{16} = 3\times 16^2 + 10\times 16^1 + 15\times 16^0 = (943)_{10}$

2．十进制整数转换成二进制整数或十六进制整数

把十进制数转换成二进制数，可以先把十进制数化成 2 的各次幂的和，各次幂的系数只能采用 0 或 1，从最高次幂起各次幂的系数就依次是二进制从左到右各个数位上的数字。

例：$(13)_{10} = 1\times 2^3 + 1\times 2^2 + 1\times 2^1 + 0\times 2^0$

$\quad\quad\quad\quad\;\; 1 \quad\quad\;\; 1 \quad\quad\;\; 1 \quad\quad\;\; 0$

所以 $(13)_{10} = (1110)_2$

一般地，如果能把一个任意的十进制数D写成如下展开式：

$$(D)_{10}=B_{n-1}\times 2^{n-1}+B_{n-2}\times 2^{n-2}+\cdots+B_1\times 2^1+B_0\times 2^0\quad(n\text{为正整数})$$

就能完成十进制数转化为二进制数，问题是如何找出展开式中的系数$B_{n-1}B_{n-2}\cdots B_0$？通常采用除 2 取余法。

例：把十进制数107转换为二进制数。

$$\begin{array}{r|l}
2 & 107 \\
2 & 53 \cdots\cdots 1\ (B_0=1)\uparrow \\
2 & 26 \cdots\cdots 1\ (B_1=1) \\
2 & 13 \cdots\cdots 0\ (B_2=0) \\
2 & 6 \cdots\cdots 1\ (B_3=1) \\
2 & 3 \cdots\cdots 0\ (B_4=0) \\
2 & 1 \cdots\cdots 1\ (B_5=1) \\
 & 0 \cdots\cdots 1\ (B_6=1)
\end{array}$$

所以，$(107)_{10}=(1101011)_2$

除 2 取余法的转换规律是：用 2 不断地去除要转换的十进制数，余数为 1，则相应位为 1；余数为 0，则相应位为 0（从低位到高位依次求得相应位的数字符号）一直到用 2 去除出现商为 0 时为止。最后一次除法除得余数为 B_{n-1}（相应于最高位的数字），然后再按从高位到低位的顺序写出这个转换后的二进制数。

类似地把十进制整数转换成十六进制整数，只需要用除 16 取余法，下面给出几个例子。

例：把下列各十进制数转换成十六进制数。

① 212　　② 835

解：

$$\begin{array}{r|l}
16 & 212 \\
16 & 13 \cdots\cdots 4\uparrow \\
 & 0 \cdots\cdots D
\end{array}
\qquad
\begin{array}{r|l}
16 & 835 \\
16 & 52 \cdots\cdots 3\uparrow \\
16 & 3 \cdots\cdots 4 \\
 & 0 \cdots\cdots 3
\end{array}$$

所以$(212)_{10}=(D4)_{16}$　　$(835)_{10}=(343)_{16}$

3．十进制小数转换成二进制或十六进制小数

因为二进制纯小数为$0.B_{-1}B_{-2}\cdots B_{-m}$（m 为正整数，$B_i=0$或 1）形成，而任一十进制纯小数若可以写成下列展开式：

$$(D)_{10}=B_{-1}\times 2^{-1}+B_{-2}\times 2^{-2}+\cdots+B_{-m}\times 2^{-m}\ (m\text{为正整数},B_i=0\text{或}1,i=-1,-2,\cdots,-m)$$

那么就可以完成十进制小数转换成二进制小数。所以问题变为找出展开式系数B_{-1}，B_{-2}，…，B_{-m}，即二进制小数从左向右依次排列各数位上的数码。通常采用乘 2 取整法。

例：把十进制小数 0.6875 转换成二进制小数。

$$\begin{array}{rr|l}
 & 0. & 6875 \\
\times) & & 2 \\ \hline
 & 1. & 3750 \cdots\cdots \text{整数部分 } 1=B_{-1} \\
 & 0. & 3750 \\
\times) & & 2 \\ \hline
 & 0. & 7500 \cdots\cdots \text{整数部分 } 0=B_{-2} \\
 & 0. & 7500 \\
\times) & & 2 \\ \hline
 & 1. & 5000 \cdots\cdots \text{整数部分 } 1=B_{-3} \\
 & 0. & 5000 \\
\times) & & 2 \\ \hline
 & 1. & 0000 \cdots\cdots \text{整数部分 } 1=B_{-4}
\end{array}$$

所以$(0.6875)_{10}=(0.1011)_2$

乘2取整法转换的规律是：用2反复去乘十进制纯小数的小数部分，每次乘上2之后，所得新数的整数部分为1，则相应位为1；整数部分为0，则相应位为0。从高位向低位依次进行，直到满足精度要求为止。最后一次乘积的整数部分为B_{-m}，然后按从高位到低位写成$0.B_{-1}B_{-2}\cdots B_{-m}$。

注意：十进制小数不一定能转换成其它数制小数，结果往往是无限小数。所以只要能满足精度要求就可以了。

例：把十进制小数0.357转换成二进制数，精确到小数点后第四位。

解：

$$\begin{array}{r l}
 0.357 & \\
\times)\quad 2 & \\
\hline
0.714 & \cdots\cdots B_{-1}=0\\
0.714 & \\
\times)\quad 2 & \\
\hline
1.428 & \cdots\cdots B_{-2}=1\\
0.428 & \\
\times)\quad 2 & \\
\hline
0.856 & \cdots\cdots B_{-3}=0\\
0.856 & \\
\times)\quad 2 & \\
\hline
1.712 & \cdots\cdots B_{-4}=1
\end{array}$$

所以$(0.357)_{10}\approx(0.0101)_2$

类似地可以把十进制纯小数转换成十六进制纯小数，采用乘16取整法。

例：把十进制数0.357转换成十六进制数，精确到小数点后三位。

解：

$$\begin{array}{r l}
0.357 & \\
\times)\quad 16 & \\
\hline
2142 & \\
357\ & \\
\hline
5.712 & \cdots\cdots \text{整数取}5\\
0.712 & \\
\times)\quad 16 & \\
\hline
4272 & \\
712\ & \\
\hline
11.392 & \cdots\cdots \text{整数取B}\\
0.392 & \\
\times)\quad 16 & \\
\hline
2352 & \\
392\ & \\
\hline
6.272 & \cdots\cdots \text{整数取}6
\end{array}$$

所以$(0.357)_{10}\approx(0.5B6)_{16}$

如果要转换十进制混合小数为二进制或十六进制数，那么只需要把十进制混合小数的整数部分和小数部分按2、3中讲的方法分别进行转换，然后再将其组合起来即可。

例：把十进制数35.6875转换成二进制数。

解：

```
2|35
2|17 ……1 ↑
  2|8 ……1
  2|4 ……0
  2|2 ……0
  2|1 ……0
    0 ……1
```

```
     0.6875
×)        2
     1.3750 ……1 |
     0.3750
×)        2
     0.7500 ……0
     0.7500
×)        2
     1.5000 ……1
     0.5000
×)        2
     1.0000 ……1 ↓
```

$(35)_{10} = (100011)_2$

所以 $(35.6875)_{10} = (100011.1011)_2$　　　$(0.6875)_{10} = (0.1011)_2$

例：把十进制数 35.6875 转换成十六进制数。

解：

```
16|35
16| 2 ……3 ↑
    0 ……2
```

```
     0.6875
×)       16
     41250
     6875
    11.0000 ……整数部分取B ↓
```

$(35)_{10} = (23)_{16}$　　　$(0.6875)_{10} = (0.B)_{16}$

所以 $(35.6875)_{10} = (23.B)_{16}$

4.二进制数与十六进制数之间的转换

因为 $2^4 = 16$，四位二进制数与一位十六进制数相对应。因此二进制数与十六进制数之间的转换十分方便。

如果把十六进制数转换成二进制数，只要把每一位十六进制数用四位二进制数表示。

例：把下列十六进制数转换成二进制数。

① $(563)_{16}$　　② $(0.76C)_{16}$　　③ $(23.B)_{16}$

解：①把数 $(563)_{16}$ 的各位5，6，3分别用四位二进制数表示，不足四位时左边补零。

```
 5     6     3
 ↓     ↓     ↓
0101  0110  0011
```

所以 $(563)_{16} = (010101100011)_2$

②

```
0.   7     6     C
     ↓     ↓     ↓
0.  0111  0110  1100
```

所以　$(0.76C)_{16} = (0.011101101100)_2$

③

```
 2     3.     B
 ↓     ↓      ↓
0010  0011.  1011
```

所以 $(23.B)_{16} = (00100011.1011)_2$

如果把二进制数转换成十六进制数，只要以小数点为界，小数点左边的整数部分自右到左每四位二进制数合成一位，不足四位时左边补 0；小数点右边的小数部分自左到右将每四位二进制数合成一位，不足四位时右边补 0，然后把分别求出的整数部分与小数部分组合在一起。

例：把下列二进制数转换成十六进制数。

① $(11101011011)_2$　② $(0.1011000111)_2$　③ $(10.01101)_2$

此为试读，需要完整PDF请访问：www.ertongbook.com

解：① $\frac{.0111}{7}$ $\frac{0101}{5}$ $\frac{1011}{B}$

所以 $(11101011011)_2 = (75B)_{16}$

② $\frac{0.1011}{0.B}$ $\frac{0001}{1}$ $\frac{1100}{C}$

所以 $(0.1011000111)_2 = (0.B1C)_{16}$

③ $\frac{0010.}{2.}$ $\frac{0110}{6}$ $\frac{1000}{8}$

所以 $(10.01101)_2 = (2.68)_{16}$

三、二进制编码

本节开始已谈到电子计算机内数字、字母、符号等以二进制编码表示，即二进制编码形式与机器的电子元件状态相对应。下面我们就来谈这种对应关系。

1. 机器数与字长

电子计算机采用二进制，用电子器件的两种稳定状态表示数据信息。即电压的高低、脉冲的有无使用 1 与 0 表示。计算机中的数据信息必须保存在有记忆功能的电子器件中，通常使用触发器。因为每个触发器可以记忆一位二进制代码 0 或 1，所以N位数就要用N位触发器。通常把N位数所需的N位触发器构成的一个部件叫寄存器，这个部件有几个触发器就叫几位寄存器。目前常用的有 8 位，16位，32位，64位等。我们主要介绍 IBM PC 机的 8 位和16位两种寄存器。若用矩形框表示一个寄存器，则八位数在机器中的表示形式如下：

1	0	1	0	1	1	1	0

八位无符号整数

数在机器中的表示形式称为机器数；表示机器数所需要的寄存器的位数就是机器数的字长。字长取决于构成寄存器的触发器的个数，对应于上面所说的 8位、16 位、32 位 等寄存器，常用的字长就有 8 位、16 位、32 位等。一般把 8 位字长称为一个字节。现在机器的字长大多是字节的整数倍。机器数具有以下三个特点：

（1）机器数所表示的数的范围受设备限制。

因为机器数的字长由寄存器的位数决定，所以一台机器的设备配制一定，字长就定了，机器数表示范围的大小也就定了。例如，使用8位寄存器这时字长为 8 位，无符号整数的表示范围最大是 $(255)_{10} = (11111111)_2$。运算时，若数值超出机器数所能表示的范围，就会停止运算和对数据的处理。计算机中称这种情况为“溢出”。

（2）机器数的符号位数值化。

因机器数有正数和负数，而计算机中只能用数字化信息来表示，所以人们规定用 0 表示正号，用 1 表示负号。这样在表示机器数的设备中要用一位来表示符号位。例如用 8 位寄存器表示有理数，用最高位 B_7 表示符号位，这时数的表示范围最大数为 $(01111111)_2 = (127)_{10}$，最小数为 $(11111111)_2 = (-128)_{10}$

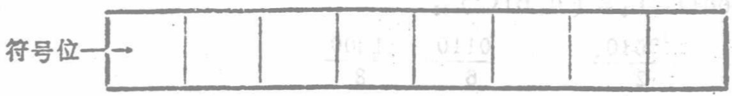
（3）机器数的小数点处于约定位置。

在计算机中，小数点的位置有两种约定：一种是按规定小数点位置固定不变，这时的机器数为定点数；一种是小数点的位置可以浮动，这时的机器数为浮点数。微型机多使用定点

数。

例如：规定小数点在最低位右边时为定点纯整数，小数点在最高位左边时为定点纯小数。

定点纯整数



定点纯小数

又如：把任一个二进制数B写成 $B=2^{P}\times s$ 的形式，其中P为阶码。P=1时，阶码为负数；P=0时，阶码为正数。S为数B的尾数，2为阶码的底数。如果在计算机中采用下面的表示形式，即为浮点表示法：

阶码符号	阶码	尾数符号	尾　数

2. 机器数的原码和补码

机器数中数值和符号全部数值化。数据运算时，计算机采用把各种符号位和数值位一起编码的方法。常见的有原码和补码。

（1）原码

原码是最简单的机器数表示法。其符号位用0表示正号，1表示负号，数值是一般二进制形式表示。如果设一个机器数为x，则原码可记作〔x〕原。

例：　$x_1=+1001010$

　　　$x_2=-1010110$

记作：〔+1001010〕原 =01001010

　　　〔−1010110〕原 =11010110

（2）补码

机器数的补码可以由原码得到。如果机器数是正数，则这个机器数的补码和原码一样。

例如，设有一个机器数x，x的补码记作〔x〕补，那么

$x=+1010110$

〔x〕原 =01010110

〔x〕补 =01010110

即　〔x〕补 =〔x〕原 =01010110

如果机器数是负数，则这个机器数的补码是对它的原码除符号位外各位取反、末位加1而得到的。

例如，设有一个机器数x，

$x=-1010110$

〔x〕原 =11010110

〔x〕补 =10101001+1=10101010

关于机器数的原码和补码的使用方法，请见第二章。

3. 数字编码

计算机内多采用二进制编码，而人们习惯使用十进制编码。为解决人机之间在计数制方面的这个矛盾，通常采用把十进制数的每一位写成二进制数形式的编码，简称二-十进制编码或BCD（Binary-Coded Decimal）编码。二-十进制编码方法很多，最常用的是8421编码，这种编码最自然和最简单。其方法是用四位二进制数表示一位十进制数，自左至右每一位对应的权是8、4、2、1。值得注意的是四位二进制数有0000～1111十六种状态，而对0～9十个数字编码只取0000～1001十种状态，其余六种不用。见表1-3。

表 1-3　8421 码与十进制数的关系

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

当我们了解了8421码编码的原则后，我们就很容易把一个十进制数写成 8421 码。

例：十进制数 953 可以写成

$$\underset{9}{\underline{1001}}\quad \underset{5}{\underline{0101}}\quad \underset{3}{\underline{0011}}$$

这种编码是用得很广泛的一种二-十进制编码。

4．字符编码

最常见的符号信息是文字符号，所以数字、字母和各种字符必须按约定的规则用二进制编码才能在机器中表示。目前在国际上广泛使用的是美国标准信息交换代码，简称ASCII码（American Standard Code for Information Interchange）。这种编码由七位码组合而成，见表1-4。

表 1-4　ASCII 码

b_7	b_6	b_5	b_4	b_3	b_2	b_1	行＼列	0	1	2	3	4	5	6	7
→								0	0	0	0	1	1	1	1
	→							0	0	1	1	0	0	1	1
		→						0	1	0	1	0	1	0	1
			0	0	0	0	0	NUL	DLE	SP	0	@	P	@	p
			0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
			0	0	1	0	2	STX	DC2	"	2	B	R	b	r
			0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
			0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
			0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
			0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
			0	1	1	1	7	BEL	ETB	'	7	G	W	g	w

1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10	LF	SS	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	;	K	[	k	{
1	1	0	0	12	FF	FS	,	<	L	~	l	¬
1	1	0	1	13	CR	GS	—	=	M	]	m	}
1	1	1	0	14	SO	RS	·	>	N	∧	n	|
1	1	1	1	15	SI	US	/	?	O	—	o	DEL

第三节　计算机系统组成

电子计算机系统由硬件和软件两大部分组成。硬件一般指用电子器件和机电装置组成的计算机实体。软件一般指为计算机运行工作服务的全部技术和各种程序。对一台计算机来说，硬件和软件二者缺一不可。自从电子计算机问世以来，它的更新换代实质上就是硬件和软件的更新换代。但是不论怎样变化，目前的电子计算机工作原理都是存储程序的原理，其基本结构属于冯·诺依曼型计算机。

一、程序和指令

电子计算机是由电子线路构成的机器，它的“聪明能干”完全是人赋予它的。当我们要求计算机完成某项任务时，例如要解一道复杂的数学题目，就必须设法把复杂的数学问题的解算方法分解成许多计算机可以实现的基本操作。由这些基本操作按一定顺序排列起来实现解题任务的步骤称之为“程序”；每一种基本操作称为一条“指令”。指令就是人对电子计算机发出的一道工作命令，它通知计算机执行某种特定的操作。计算机按顺序执行这些指令，实现解题任务。如果对计算机所进行的工作要有所变动，只需修改程序，即修改指令的内容或执行顺序就可以了。请看一个例子。

例：计算　2＋1＝？

为计算机编排计算步骤（就是写程序）：

第一步　把数据2由存储器取出来，暂存；

第二步　把数据1由存储器取出来，暂存；

第三步　作2＋1的运算，结果为3，暂存；

第四步　结果3送存储器保存起来；

第五步　停机。

这样一个解题步骤就是我们为计算机解2＋1＝？编制的程序。具体实现每一步骤要对计算机下达一条操作命令，称为“机器指令”。

机器指令必须满足两个条件：

（1）机器指令的形式用计算机能够理解的数字编码形式表示。

（2）机器指令规定的操作必须是计算机能够执行的，也就是说机器指令要和实现此指令操作的电子线路一一对应。例如，某计算机有取数、存数、加法、停机等四种机器指令，我们便可以把上面步骤编制成指令序列，构成2＋1＝3的计算程序。

指令序列：①　取数指令（取2）；

　　　　　②　取数指令（取1）；

③ 加法指令（2+1）；

④ 存数指令（存3）；

⑤ 停机指令。

程序中的每一条指令代表一种操作。例如取数指令完成将需要的数据送入存储器，加法指令完成两数相加的操作等。因此每台机器的机器指令有标准格式和具体含义。通常一条指令包括两部分内容：

（1）指出机器进行什么操作的操作码；

（2）指出参与操作的数放在存储器中的地址码。

机器指令格式：

操作码	地址码

例如我们使用Z-80机器指令的数字编码形式，操作码为：

操　作	取数	加	存　数	停机	…
操作码	3E	80	02	76	

上面2+1=3的指令序列可写成的数字编码形式为：

① 00111110

② 00000110

③ 10000000

④ 00000010

⑤ 01110110

每台计算机都规定了一定数量的基本指令，也就是说，为它设计好实现一批基本操作的线路。这种机器指令的总和称为计算机的指令系统。不同机器的指令系统所具有的指令种类和数目不同。如Z-80微型机有158种机器指令，APPLE Ⅱ微型机有56条指令，IBM-PC使用的8088约有100种机器指令。无论指令系统差异多大，但基本功能必须具备。

一般机器指令系统应具备下述功能的指令：

1．数据传送指令

这类指令的任务主要是把保存在存储器中的数据取出来进行运算，或把运算结果送到存储器中保存起来，或在各存储器和寄存器之间传送数据。

2．进行算术运算和逻辑运算指令

功能强的计算机可有几十条算术运算指令。在各种算术运算中，最基本的是加法和移位。能执行这两种运算，其它运算就可以用程序实现。

逻辑运算是一种对取值只有两种情况的逻辑变量运算，基本运算在第二章第二节中介绍。

3．程序控制类指令（也称转移指令）

这类指令的功能是根据指令中给定的条件改变程序执行的顺序，这些指令使计算机有了逻辑判断的功能。

4．输入输出指令

这些指令的功能是完成由外部的一些设备把数据送到计算机内。或把计算机中的结果数据送到某些外部设备的传送任务。

5. 各种控制管理机器的指令

例如停机、启动、复位、清除等指令。

上述几类指令是各种计算机不可缺少的。计算机指令系统影响着计算机的结构，一般指令系统的功能越强，人们编制程序就越方便，但随之机器的结构也就越复杂。

二、存储程序原理

存储程序原理是计算机结构设计的基础。由于计算机的工作就是执行程序，而程序是一条条按一定规则顺序排列的机器指令，所以计算机要实现自动连续操作，必须使它开始工作后就能自动按程序中规定顺序取出要执行的指令，然后执行它规定的操作。因此要解决两个问题：

1. 计算机应能知道在什么时间到什么地方去取哪条指令；
2. 计算机在执行完一条指令后又自动去取要执行的下一条指令。

为解决这两个问题，计算机中设置了一个担任指挥员功能的电子部件——控制器。当计算机工作时，控制器使计算机只要知道程序中第一条指令放在什么地方，就能顺序依次取出每条指令加以识别，并执行相应的操作。这就是计算机自动连续工作的基础——存储程序原理。值得注意的是：

（1）计算机在执行程序时必须有一个先决条件，即计算机开始工作前，要把人预先编好的程序和数据按一定顺序一条条地通过一定方式送到有记忆功能的电子部件——存储器中保存起来。

（2）程序中的指令系统为编码化数字组成，以便使程序和数据一样保存在存储器中。

（3）计算机能直接理解并执行的程序中的指令都属于这台计算机的指令系统，即这些程序是面向机器的机器语言程序。

存储程序原理是前人智慧和创造的结晶。许多科学家如帕斯卡、莱布尼兹、巴贝奇等为探讨现代计算机原理都付出了长期的辛勤劳动，甚至花费了毕生的精力，直到1946年才由冯·诺依曼正式提出并论证。存储程序原理实现了计算机自动计算，同时也确定了冯·诺依曼型计算机的基本组成。

三、冯·诺依曼型计算机基本组成

1946年，冯·诺依曼领导的研制小组提出的计算机新设计方案中，明确指出电子计算机至少应由运算器、控制器、存储器、输入设备、输出设备五部分组成，并描述了五部分的职能和关系。同时，还确定了指令和数据均以二进制数的形式存储。这一方案简化了计算机结构，提高了计算机运算速度，使计算机具有了通用性。冯·诺依曼思想被誉为计算机发展史上的里程碑，标志着电子计算机时代的真正开始。

迄今为止，各类计算机基本组成仍属于冯·诺依曼型计算机。冯·诺依曼型计算机的基本组成及信息通路如图1-1所示。

1. 存储器

要实现存储程序，计算机中必须设有存储信息的装置——存储器。存储器的主要功能是保存大量信息，它的作用类似一台录音机，能把记录内容长期保存。使用时可以取出原记录内容，而不破坏其信息，也可以把原来内容抹去，重新记录内容。存储器装置一般用电子或电磁技术实现。然而在实现上，容量和速度存在尖锐的矛盾，所以几乎所有的电子计算机系统

此为试读，需要完整PDF请访问：www.ertongbook.com