微型计算机在医院中的应用

人民軍医出版社

微型计算机在医院中的应用

WEIXING JISUANJI ZAI YIYUAN ZHONG DE YINGYONG

陶乃煌　杨友春
邵立生　廖伟民　刘安滨　编著

人民军医出版社

1987年·北京

微型计算机在医院中的应用
陶乃煌　杨友春 等编著
·
人民军医出版社出版
（北京市复兴路22号甲 3 号）
解放军出版社发行
中国人民解放军第一二〇二工厂印刷
·
开本：787×1092毫米1/32．印张：6.125．字数：126千字
1987 年 1 月第 1 版
1987 年 1 月（北京）第 1 次印刷
统一书号：14281·GW05

内 容 提 要

本书是一本关于医院如何应用微型计算机的简明参考书，内容包括微机在辅助诊断，医院管理，生理信号、医学图象分析，药物代谢动力学，生物医学数学模型和统计处理等各方面的具体应用，并介绍了微机的基本知识。由国内开展微机应用较早的南京军区总医院的同志编写。内容较切合实际，体现了作者的经验和国内、外的技术理论。

读者对象：医院微机工作人员，医疗行政管理人员，一般医生、护士。

前　言

近几年来，我国很多医院陆续开始应用电子计算机，其范围涉及辅助诊断，生理信号处理，医学影象处理，临床药理，制定放射治疗计划，医学监护，病案及情报资料检索，统计处理以及医院管理等各个方面。不少同志是在计算机技术力量和物质基础比较薄弱的情况下，开展工作并做出成绩的，这是一种非常可贵的精神。虽然现在医院里的计算机的应用还不是很普遍，开展的项目也不是很多，但是我们确实已经起步，未来的美好前景已经在望。

目前，一个以电子计算机应用为重要标志的新技术革命正在世界兴起，为了迎接这个挑战，我们应积极开展计算机在医学中的应用，特别是微型机的应用。由于我国医院和卫生系统中应用计算机，在人力物力方面受到一定限制，当前以微型机起步，用以处理各种专题项目，是比较适宜并切实可行的。通过微型机的实践，探索计算机在医院的应用，逐步积累经验，待条件成熟时，联成微机局部网络，或以至进而建成比较完整的医院信息系统。而微型机仍将为处理某些专题或成为系统的智能终端继续发挥作用。根据目前情况，就我国国情而言，在医院广泛开展微型机的应用，是非常重要的步骤。

本书内容主要包括微型计算机基本知识及其在辅助诊断、医院管理、生理信号分析、医学图象分析、药代动力学、生物医学数学模型和统计处理等各个方面的应用。本

书系根据文献资料和一些工作实践编写而成，书中可能有许多不足之处，敬请读者不吝指正。

谨以此书奉献给有志于发展医院计算机事业的同志们。

作　者

1985年6月于南京军区总医院

目　录

第一章　微型计算机基础知识

一、概　　述

微型计算机是在以往大中小型数字电子计算机技术发展的基础上和大规模集成电路相结合的产物。自从1971年美国Intel公司研制成功第一片4位微处理器芯片4004以来，微型计算机的发展极为迅速，在短短的十几年中，它已经历了4位机、8位机、16位机和32位机4个发展阶段。

微型计算机和大中小型计算机相比，具有许多独特的优点：体积小，结构合理，扩展、适应性强，尤其价格低廉，使用方便，稳定可靠等。因此微型计算机应用非常广泛，现已深入到包括医学科学在内的社会生活的各个领域，有着强大的生命力和广阔的发展前途。

微型机采用的计数方法和工作原理，同大中小型计算机基本上一致。本章将介绍有关计算机的一些基本知识。

二、计算机中数的表示方法

数字电子计算机最基本的功能是对"数"进行运算、加工和处理，由于在计算机里用两种不同的物理状态来表示数，即采用二进制数(所谓二进制数就是由0和1这二个数符所组成的数)，其他形式的数、文字、符号等都需用一定的方法将其转换成二进制数，才能被计算机加工处理。

日常生活中最常用的数制是十进制数，此外还有十二进制数、六十进制数等。由于各种不同进位的数制之间可以相互转换，因此，在微型计算机中，常用的计数方法有二进制、八进制、十进制和十六进制。

（一）各种进位计数制及其表示方法

进位计数制就是按一定的进位方法进行计数，例如十进制是“逢十进一”，二进制是“逢二进一”。各种进位计数制尽管进位方式不同，但都有以下特点：

1. 在R进位制中，任何一个数都是由R个基本数码（或称数符）0，1，2，…，R-1所组成，R叫做基数。例如在十进制中，基数为10，数码为0，1，2，…，9；在二进制中，基数为2，数码为0，1；在十六进制中，基数为16，数码为0，1，2，…，9，10（用A代表，下同），11(B)，12(C)，13(D)，14(E)，15(F)。

2. 在计数或运算时“逢R进一”。

3. 每个数码代表的数值大小与它所处的位置有关。若数a位于小数点前第i位，则其数值为$a \times R^{i-1}$，这里R^{i-1}称为小数点前第i位的权；若位于小数点后第j位，则其值为$a \times R^{-j}$，R^{-j}称为小数点后第j位的权。例如二进制数100.1，小数点前第3位的权为$2^{3-1}=2^2=4$，小数点后第一位的权为$2^{-1}=0.5$。

4. 在R进位制中，任何一个数S可表示为R的多项式：

$$
\begin{aligned}
S &= \pm a_{m-1}a_{m-2}\cdots a_1 a_0 \cdot a_{-1}\cdots a_{-n+1}a_{-n} \\
&= \pm(a_{m-1}\times R^{m-1} + a_{m-2}\times R^{m-2} + \cdots \\
&\quad + a_1\times R^1 + a_0\times R^0 + a_{-1}\times R^{-1} + \cdots \\
&\quad + a_{-n+1}\times R^{-n+1} + a_{-n}\times R^{-n})
\end{aligned}
$$

$$= \pm \sum_{i=-n}^{m-1} a_i \times R^i$$

其中 a_i 是R进制的数码，下标i表示该数码所处的位置；m表示该数的整数部分的位数，n表示小数部分的位数。

例如二进制数

$$1011.1_{(2)} = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1}$$
$$= 11.5_{(10)}$$

八进制数 $327_{(8)} = 3 \times 8^2 + 2 \times 8^1 + 7 \times 8^0 = 215_{(10)}$

十进制、二进制、八进制和十六进制数码的等值表，如表1所示。

表1　十进制、二进制、八进制和十六进制数码的等值表

十进制数	二进制数	八进制数	十六进制数
0	0000	00	0
1	0001	01	1
2	0010	02	2
3	0011	03	3
4	0100	04	4
5	0101	05	5
6	0110	06	6
7	0111	07	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

（二）二进制的特点及运算规则

在计算机中广泛采用二进制，是因其具有以下优点：（1）二进制只有0、1两个状态，这两个对立的状态可以用微机中器件的两种物理状态来表示，如可用脉冲的有、无，或电位的高、低表示1和0，因此，二进制在计算机中容易用物理器件、电子线路来实现，但是，十进制有十个状态，要用某种物理器件表示十种状态就比较困难。（2）二进制只有两种状态，可以应用后面介绍的逻辑代数这一数学工具，用逻辑运算来实现二进制算术运算，同时可对计算机逻辑电路进行分析和综合。（3）二进制四则运算比较简单。（4）十进制数和其他各种进制数与二进制数可相互转换。

二进制数的四则运算规则类似于十进制数，不同的是在加法中"逢二进一"，减法中"借一当二"。

（1）加法

0＋0＝0，0＋1＝1，1＋0＝1，1＋1＝10（逢二进一）

例如111(7)＋1101(13)＝10100(20)（括号内为十进制数），过程是：

$$\begin{array}{rrr} & 111 & (7) \\ +) & 1101 & (13) \\ \hline & 10100 & (20) \end{array}$$

（2）减法

0－0＝0，1－0＝1，1－1＝0，10－1＝1（借一当二）

例如1101(13)－111(7)＝110(6)：

$$\begin{array}{rrr} & 1101 & (13) \\ -) & 111 & (7) \\ \hline & 110 & (6) \end{array}$$

（3）乘法

$0\times0=0$，$0\times1=0$，$1\times0=0$，$1\times1=1$

例如111(7)×101(5)＝100011(35)

$$
\begin{array}{rr}
 & 111\ (7) \\
\times) & 101\ (5) \\
\hline
 & 111 \\
+) & 1110 \\
\hline
 & 100011\ (35)
\end{array}
$$

（4）除法

$0\div1=0$，$1\div1=1$

例如100011(35)÷111(7)＝101(5)，

$$
\begin{array}{r}
101(5) \\
111\,\big/\,\overline{100011} \\
\underline{111} \\
111 \\
\underline{111} \\
0
\end{array}
$$

（三）十进制数与二进制数之间的转换

根据任意两个数相等必然是它们的整数部分和小数部分分别相等的原则，数制之间的转换，可以分二步即整数部分和小数部分进行转换。

1． 十进制整数转换成二进制整数——除2取余法

因为十进制整数S可用二进制表示为，

$$
\begin{aligned}
S_{10} &= (a_{m-1}a_{m-2}\cdots a_1a_0)_2 \\
&= a_{m-1}\times2^{m-1}+a_{m-2}\times2^{m-2}+\cdots+a_0\times2^0
\end{aligned}
$$

因此，只要求出a_{m-1}，a_{m-2}，…，a_1，a_0就可得到十进制整数S的二进制整数表示式。

在上式右边的多项式中，除a_0项外，其余各项均能被2整除，只有a_0除不尽成为余数。

$$\frac{S_{10}}{2} = \underbrace{a_{m-1} \times 2^{m-2} + a_{m-2} \times 2^{m-3} + \cdots + a_1}_{商} + 余数$$

若余数为 0（即 S_{10}能被 2 整除），则得$a_0 = 0$；若余数为 1（即 S_{10}不能被 2 整除），则得$a_0 = 1$。

同理，将所得商再除以 2，得到新的商和新的余数，该余数就是a_1的值。依次类推，继续用 2 整除，直到商数等于 0 为止，这时，最后的余数就是a_{m-1}。因此

$$S_{10} = (a_{m-1}\, a_{m-2} \cdots a_1 a_0)_2$$

上述方法称为“除 2 取余法”。

例如将$(725)_{10}$转换成二进制数，采用除 2 取余法

$$\begin{array}{r|l l}
2 & 725 & \cdots\cdots\text{余数为 } 1,\ a_0 = 1 \\ \hline
2 & 362 & \cdots\cdots\text{余数为 } 0,\ a_1 = 0 \\ \hline
2 & 181 & \cdots\cdots\text{余数为 } 1,\ a_2 = 1 \\ \hline
2 & 90 & \cdots\cdots\text{余数为 } 0,\ a_3 = 0 \\ \hline
2 & 45 & \cdots\cdots\text{余数为 } 1,\ a_4 = 1 \\ \hline
2 & 22 & \cdots\cdots\text{余数为 } 0,\ a_5 = 0 \\ \hline
2 & 11 & \cdots\cdots\text{余数为 } 1,\ a_6 = 1 \\ \hline
2 & 5 & \cdots\cdots\text{余数为 } 1,\ a_7 = 1 \\ \hline
2 & 2 & \cdots\cdots\text{余数为 } 0,\ a_8 = 0 \\ \hline
2 & 1 & \cdots\cdots\text{余数为 } 1,\ a_9 = 1 \\ \hline
 & 0 &
\end{array}$$

因此得到$(725)_{10}=(1011010101)_2$

2. 十进制纯小数转换成二进制小数——乘 2 取整法

十进制纯小数 S_{10}可表示成

$$S_{10}=(0.a_{-1}a_{-2}\cdots a_{-n+1}a_{-n})_2$$
$$=a_{-1}\times 2^{-1}+a_{-2}\times 2^{-2}+\cdots+a_{-n+1}\times 2^{-n+1}$$
$$+a_{-n}\times 2^{-n}$$

只要求得a_{-1}，a_{-2}，…，a_{-n+1}，a_{-n}的值，就得到了十进制纯小数 S 的二进制小数的表示式。

在上式中，两边同乘以 2 得

$$2\times S_{10}=a_{-1}+\underbrace{a_{-2}\times 2^{-1}+a_{-3}\times 2^{-2}+\cdots+a_{-n}\times 2^{-n+1}}_{\text{纯小数}}$$

此式右边除第一项a_{-1}外，其余各项均为小数。因此，$2\times S_{10}$的整数部分的值，即为a_{-1}的值。若$2\times S_{10}$的整数部分为 0，则$a_{-1}=0$；若$2\times S_{10}$的整数部分为 1，则$a_{-1}=1$。设 S'为 S 乘 2 之后的小数部分，得

$$S'_{10}=a_{-2}\times 2^{-1}+a_{-3}\times 2^{-2}+\cdots+a_{-n}\times 2^{-n+1}$$

同理，上式两边再乘以 2，即 $S'\times 2$的整数部分即为a_{-2}。依次类推，可以求得a_{-3}，a_{-4}，…直到被乘的十进制数的小部分为 0 或者当整个计算过程可能无限制地进行下去时，取所求的二进制小数满足指定的精度要求为止，这样就可得到十进制纯小数的二进制表示：

$$S_{10}=(0.a_{-1}a_{-2}\cdots a_{-n})_2$$

此法称为“乘 2 取整法”。

例：将$(0.6875)_{10}$转换成二进制小数，采用乘 2 取整法

$$\begin{array}{r} 0.6875 \\ \times\qquad 2 \\ \hline 1.3750 \end{array}$$ ……整数为 1，$a_{-1}=1$