Weixingjisuanjiyuanli

微型计算机原理

初级教材

赵心文 田耕宇

吉林教育出版社

微型计算机原理

初 级 教 材

赵心文　田耕宇　编著

吉 林 教 育 出 版 社

内 容 提 要

该书是微型计算机的硬件教材。书中首先介绍 Z80—CPU的几种常用指令，目的是让学生先了解这种微型计算机的基本功能。然后，在对主机和外部设备的介绍中，着重说明它们各自是如何实现自己的功能的。并且结合实际芯片分析其工作原理。在中断和接口两章里，总结概括了它们的基本思想、实现过程以及接口电路的组成与工作原理。

本书起点较低，但上升幅度较大，如能按步学完此教材，除了对微型机原理能基本掌握外，对接口电路也能做些初步的实际应用。

微型计算机原理

初级教材

赵心文　田耕宇　编著

*

吉林教育出版社出版　吉林省新华书店发行

长春科技印刷厂印刷

*

787×1092毫米32开本　8.125印张　178,000字

1986年6月第1版　1986年6月第1次印刷

印数：1—1,270册

统一书号：7375·269　定价：1.30元

序　言

微型计算机的应用，正在深入到社会的各个领域。为了更好地使用微型计算机，首先应该对微型机的基础知识和基本原理有所了解。为此，我们编写了这本教材。

在这本教材里，主要介绍微型计算机硬件方面的基础知识和各组成部分的工作原理。针对微型机的特点，我们从汇编语言的常用指令入手，以指令为中心，循序渐进、由表及里地对各章内容进行分析与综合。

本书适于作为中等学校以及大专院校非计算机专业学生的教材。特别对初学者在短期内作为学习教材更为适宜。书中部分打星号章节为选学内容。

在本书编写过程中，吉林工学院计算机教研室部分同志曾提出许多宝贵意见。在此谨致衷心感谢。

本书力求通俗易懂、深入浅出，尽量抓住实质性问题，注意讲清基本概念是我们编写中的主导思想。但是，由于我们水平有限，时间匆促，缺点错误一定难免，敬请批评指正。

编者　一九八五年一月

目　录

第一章　计算机的基础知识

电子计算机的全部功能是通过算术运算和逻辑运算来实现的。在计算机里参加运算的数是二进制的，逻辑运算由逻辑电路完成。本章就来介绍这些计算机的基础知识。

第一节　二进制数及其运算

一、进位计数制

按一定规则进行计数的方法称为进位计数制。我们日常生活中常用的十进制，就是逢十进一的一种进位计数制。

在十进制数中，每位数所代表的数值是不同的。如1985这个十进制数，从右边起第一位可以表示为5×10^0，从右边起第二位可以表示为8×10^1，右边起第三位可以表示为9×10^2，右边起第四位可以表示为1×10^3，所以有：

$$1985=1\times10^3+9\times10^2+8\times10^1+5\times10^0$$

此式说明，在十进制中，任意一个数都可以由一个多项式来表示，多项式的每一项为相应位数字与10的不同次方的乘积。上例的多项式中的10叫基数，即十进制的基数是10。

二、二进制数

根据前面的介绍，二进制是以 2 为基数的一种进位计数

制。在二进制数中，每一位所能选择的数字只有1和0两种。如11010就是一个二进制数，它与十进制数的对应关系是

$$(11010)_2 = 1\times2^4 + 1\times2^3 + 0\times2^2 + 1\times2^1 + 0\times2^0 = (26)_{10}$$

括号外边的小字注脚是括号内的数的进位计数制的基数。

二进制数中的1或0代码，可用来代表电路中某一条导线上电位的高或低。如高电位表示1，低电位表示0。这种表示方法，在电路上实现是很方便的，所以，在电子计算机中都采用二进制数。

二进制数的进位关系是逢二进一。二进制数与十进制数的对应关系可参看表1—1。

表1—1　　二进制数与十进制数对照表

二进制数	十进制数	二进制数	十进制数
0	0	0.10000	$\frac{1}{2}=0.5$
1	1	0.01111	$\frac{15}{32}=0.46875$
10	2	0.01110	$\frac{14}{32}=0.4375$
11	3	0.01101	$\frac{13}{32}=0.40625$
100	4	0.01100	$\frac{12}{32}=0.375$
101	5	0.01011	$\frac{11}{32}=0.34375$
110	6	0.01010	$\frac{10}{32}=0.3125$

二进制数	十进制数	二进制数	十进制数
111	7	0.01001	$\frac{9}{32}=0.28125$
1000	8	0.01000	$\frac{8}{32}=0.25$
1001	9	0.00111	$\frac{7}{32}=0.21875$
1010	10	0.00110	$\frac{6}{32}=0.1875$
1011	11	0.00101	$\frac{5}{32}=0.15625$
1100	12	0.00100	$\frac{4}{32}=0.125$
1101	13	0.00011	$\frac{3}{32}=0.09375$
1110	14	0.00010	$\frac{2}{32}=0.0625$
1111	15	0.00001	$\frac{1}{32}=0.03125$
10000	$2^4=16$	0.000001	$2^{-6}=0.015625$
100000	$2^5=32$	0.0000001	$2^{-7}=0.0078125$
1000000	$2^6=64$	0.00000001	$2^{-8}=0.00390625$
$\underbrace{100\cdots00}_{n个0}$	2^n	$\underbrace{0.00\cdots0}01$ n个0包括小数点前面0	2^{-n}

三、二进制数的运算

二进制数的基数是 2 。每位可选择的数字只有 1 和 0 ，所以，二进制数的运算比较简单。

加法规则：$0+0=0$

$0+1=1$

$1+0=1$

$1+1=10$

乘法规则：$0\times0=0$

$0\times1=0$

$1\times0=0$

$1\times1=1$

根据运算规则，现举例如下：

〔例 1〕加法

$$\begin{array}{r} 1101 \\ +\ 1110 \\ \hline 11011 \end{array}$$

〔例 2〕减法

$$\begin{array}{r} 1101 \\ -\ 1011 \\ \hline 10 \end{array}$$

在作减法时，若不够减，则借 1 当 2 。这个借位关系与加法中逢二进一是相对应的。

〔例 3〕乘法

$$\begin{array}{r} 1101 \\ \times\ \ 101 \\ \hline 1101 \\ 0000\ \ \\ 1101\ \ \ \ \\ \hline 1000001 \end{array}$$

〔例 4〕除法

$$\begin{array}{r} 101 \\ 1011\overline{)110111} \\ \underline{1011} \\ 1011 \\ \underline{1011} \\ 0 \end{array}$$

由例 4 看出，二进制数的除法所得的商数中的每一位只能取 1 或 0。

第二节　不同计数制之间的转换

一、二进制数与十进制数之间的转换

1. 二进制数转换成十进制数

前面介绍过，任意一个二进制数都可以表示成多项式形式。把多项式中 2 的次方化为十进制数，即可把一个二进制数转换成十进制数。

如 $(11101)_2 = 1\times2^4+1\times2^3+1\times2^2+0\times2^1+1\times2^0$

$= (16)_{10}+(8)_{10}+(4)_{10}+(1)_{10}=(29)_{10}$

对于二进制数中的小数部分转换成十进制数的方法，与整数的转换方法一样。也是先把二进制数的小数部分用一个多项式表示，然后，把每一项所对应的十进制数值加在一起。

如 $(0.101)_2 = 1\times2^{-1}+0\times2^{-2}+1\times2^{-3}$

$= (0.5)_{10}+(0.125)_{10}=(0.625)_{10}$

例如：将 $(1110.1011)_2$ 转换成十进制数。

首先用多项式来表示二进制数

$$(1110.1011)_2 = 1\times2^3+1\times2^2+1\times2^1+0\times2^0+1\times2^{-1}$$
$$+0\times2^{-2}+1\times2^{-3}+1\times2^{-4}$$

把多项式中的每一项所代表的十进制数相加，得到 $(1110.1011)_2=(14.6875)_{10}$

2. 十进制数转换成二进制数

十进制数转换成二进制数的转换方法，分整数的转换和小数的转换两种情况。我们先介绍整数的转换方法。

取一个只有整数的十进制数（无小数部分）。转换成二进制数的方法是先把这个整数被2除。若余数是0，说明该十进制数恰为2的整数倍，也就是说，这时的二进制数的最低位必为0；若余数是1，说明该十进制数是奇数，这时的二进制数的最低位必为1。这样就确定出二进制数的最低一位（从右边数第一位）。确定出第一位后，再把除得的商被2除，若余数是0，说明二进制数右起第二位为0；若余数是1，说明二进制数右起第二位为1。如此继续作下去，直到商数得零为止。根据每次所得的余数就可以得到对应的二进制数。

将上面方法归纳起来，十进制整数转换成二进制整数是通过除2取余数方法完成的。故简称为“除2取余法”。

〔例1〕把 $(285)_{10}$ 转换成二进制数。

用除2取余法转换。

```
2|2 8 5
  2|1 4 2  ……余1（最低位）↑
   2|7 1   ……余0         |
     2|3 5 ……余1         |
     2|1 7 ……余1         |
     2|  8 ……余1         |
     2|  4 ……余0         |
     2|  2 ……余0         |
     2|  1 ……余0         |
         0 ……余1（最高位）|
```

所以，$(285)_{10}=(100011101)_2$。

〔例 2〕将 $(29)_{10}$ 转换成二进制数。

用除 2 取余法转换。

```
2|2 9           ↑
 2|1 4 ……余1    |
   2|7 ……余0    |
   2|3 ……余1    |
   2|1 ……余1    |
     0 ……余1    |
```

所以，$(29)_{10}=(11101)_2$

现在我们来研究十进制小数如何转换成二进制小数。

十进制纯小数，可以用“乘 2 取整法”转换成二进制小数。所谓“乘 2 取整法”就是先用 2 乘十进制小数，然后去掉乘积中的整数部分，再用 2 乘余下的纯小数部分。如此继续进行下去，其中每次乘 2 后得到的整数部分按着由前到后的次序排列起来，就是所求的二进制小数。

例如：有十进制小数 $(0.6875)_{10}$，用“乘 2 取整法”转换成二进制小数。具体作法如下：

$$
\begin{array}{r}
0.6875 \\
\times \quad\ \ 2 \\
\hline
1.3750
\end{array}
\cdots\cdots\text{小数点后第一位取 1}
$$

$$
\begin{array}{r}
0.375 \\
\times \quad\ \ 2 \\
\hline
0.750 \\
\times \quad\ \ 2 \\
\hline
1.500
\end{array}
\begin{array}{l}
\\
\\
\\
\cdots\cdots\text{小数点后第二位取 0} \\
\\
\\
\cdots\cdots\text{小数点后第三位取 1}
\end{array}
$$

$$
\begin{array}{r}
0.5 \\
\times \ \ 2 \\
\hline
1.0
\end{array}
\cdots\cdots\text{小数点后第四位取 1}
$$

所以，$(0.6875)_{10}=(0.1011)_2$

十进制小数转换成二进制小数时，不一定每个十进制小数都能精确地转换成二进制小数，这时乘 2 的过程会无限制的进行下去，究竟取几位二进制小数近似地表示十进制小数，必须根据精度要求来决定。

一个带小数的十进制数转换成二进制数时，十进制数的整数部分转换成二进制数的整数部分，十进制数小数部分转换成二进制数小数部分。

例如：$(29.6875)_{10}=(11101.1011)_2$

二、二进制数与八进制数之间的转换

计算机采用二进制数，这对人们来说是不习惯的。实践证明，用八进制数作为二进制数的书写形式是比较方便的，所以在人与计算机打交道时，八进制数是经常使用的一种计数制。

八进制数有两个特点，一是每位数字可以有 0～7 八种不同的选择（即以 8 为基数）；二是在作算术运算时逢八进一。

1. 二进制数转换成八进制数的方法

通过前面的分析知道，一个二进制数，从右起第一位表示的数值是1，右起第二位表示2，第三位表示4，这三位加起来为7，若再加1就产生八进制进位，因此从低位开始，每三位二进制数恰好表示一位八进制数。于是得出二进制数转换成八进制数的方法：把一个二进制数，从右边第一位数起，每三位分成一组。然后在每一组内把三位二进制数所代表的八进制数值写出来，最后将各组所求出的八进制数字连在一起即为所要转换的八进制数。

例如：$(11101110)_2=(11,101,110)_2=(356)_8$

$(10101110)_2=(10,101,110)_2=(256)_8$

2. 八进制数转换成二进制数的方法

把一个八进制数转换成二进制数的方法和上面过程相反。即把八进制数中的每一位都写成三位二进制数，再把各位连接起来就得到二进制数了。

例如：$(3265)_8=(011,010,110,101)_2$

$=(11010110101)_2$

$(1740)_8=(001,111,100,000)_2$

$=(1111100000)_2$

三、二进制数与十六进制数之间的转换

十六进制的基数是16。进行算术运算时是逢十六进一。每位可以选择的数字是0、1、2、3、4、5、6、7、8、9、A、B、C、D、E、F。其中字母A～F分别表示十进制数的10～15，如$A=(10)_{10}$，$B=(11)_{10}$等等。

参照二进制数与八进制数的对应关系，每四位二进制数恰好对应一位十六进制数。所以它们之间的转换方法也与二进制数与八进制数之间的转换相类似。下面用几个实例来说

明。

〔例 1〕 $(11110110)_2=(1111,0110)_2=(F6)_{16}$

〔例 2〕 $(01110010)_2=(0111,0010)_2=(72)_{16}$

〔例 3〕 $(10101011)_2=(1010,1011)_2=(AB)_{16}$

〔例 4〕 $(F93D)_{16}=(1111100100111101)_2$

〔例 5〕 $(E82C)_{16}=(1110100000101100)_2$

〔例 6〕 $(47AB)_{16}=(100011110101011)_2$

思　考　题

1. 有二进制数：10011，1111010，110110，101010，100001。

① 把它们转换成十进制数；

② 把它们转换成八进制数；

③ 把它们转换成十六进制数。

2. 把十进制数256，4391，78652转换成二进制数。

3. 把下列各数转换成二进制数。

$(FEDCB)_{16}$，$(A9876)_{16}$，$(54321)_{10}$，$(427)_{10}$，$(635)_8$，$(201)_8$。

第三节　原码、反码和补码

一、机器数和真值

先看两个二进制数：

$$X_1=+1011001$$

$$X_2=-1001011$$

这两个数不但有数值，而且还有正负号。在计算机中数的正

负号也是用0、1来表示的。规定每个数的最高位是符号位，用0表示正，用1表示负。符号位后面是数的数值部分。按着这个规定，上面的X_1、X_2在计算机中应表示为：

01011001、11001011

把在计算机中数的符号位用0、1表示的这些数叫做机器数。在计算机内部使用的数都是机器数。因为机器数的最高位是符号位，它与原来的形式有所不同。我们把原来用正负号表示符号的数称为真值。真值是指我们平时常用的二进制数。如$X_1 = +1011001$，$+1011001$就是真值，而01011001就是机器数。

在计算机中，机器数有三种常用的表示方法，它们分别是原码、反码和补码。对正数来说，它的原码、反码和补码完全是一样的；但对负数来说，三种代码的表示形式完全不同。

二、原　码

原码是一种最简单的机器数表示法。它的符号位是用0表示正数，用1表示负数。它的数值部分取真值的数值部分。

例如：真值$X_1 = +0110101$

X_1的原码为00110101

真值$X_2 = -0110101$

X_2的原码为10110101

原码表示的二进制数简单易懂，是计算机常用的一种机器数的表示法。

但是，当作减法运算时，用原码很麻烦，为了使运算简便，在计算机里，机器数还有两种表示法也是经常采用的，这就是反码和补码。