

2023Lecture 01 数制与编码OK

问题：数据在计算机内如何表达？

一、进位计数制

要点：

- ①一个基数 N (≥ 2)：计数时满 N 进1；
- ② N 个数字：循环使用 N 个数字。

位权：不同位置上的数字的含义不一样。

问题： N 进制数中有 N 吗？如果没有，那 N 进制数中的 N 长啥样？

表2.1.1 数字系统中常用的数制

数制	采用的数码	进位规则	基数	位权	位权展开表示
二进制	0, 1	逢二进一 借一当二	2	2^i	$\sum_{i=-m}^{n-1} a_i \times 2^i$
八进制	0, 1, 2, 3, 4, 5, 6, 7	逢八进一 借一当八	8	8^i	$\sum_{i=-m}^{n-1} a_i \times 8^i$
十进制	0, 1, 2, 3, 4, 5, 6, 7, 8, 9	逢十进一 借一当十	10	10^i	$\sum_{i=-m}^{n-1} a_i \times 10^i$
十六进制	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F (符号 A, B, C, D, E, F 分别代表 十进制数 10, 11, 12, 13, 14, 15)	逢十六进一 借一当十六	16	16^i	$\sum_{i=-m}^{n-1} a_i \times 16^i$

一般形式： $\left(x_{n-1}x_{n-2}\cdots x_1x_0.x_{-1}x_{-2}\cdots x_{-n}\right)_N$

多项式形式：

$$\left(x_{n-1}x_{n-2}\cdots x_1x_0.x_{-1}x_{-2}\cdots x_{-n}\right)_N$$

$$= x_{n-1} \times N^{n-1} + x_{n-2} \times N^{n-2} + \cdots + x_1 \times N + x_0$$

$$= x_{n-1} \times N^{n-1} + x_{n-2} \times N^{n-2} + \cdots + x_1 \times N + x_0$$

$$+ x_{-1} \times N^{-1} + x_{-2} \times N^{-2} + \cdots + x_{-n} \times N^{-n}$$

二、整数间的换算

N 进制到 **10** 进制:

$$(x_{n-1}x_{n-2}\cdots x_1x_0)_N$$

$$= x_{n-1} \times N^{n-1} + x_{n-2} \times N^{n-2} + \cdots + x_1 \times N + x_0$$

例如: $1234_5 = 1 \times 5^3 + 2 \times 5^2 + 3 \times 5 + 4 = 194$

$$1AB7_{16} = 1 \times 16^3 + 10 \times 16^2 + 11 \times 16 + 7$$

$$= \dots$$

$$10101_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2 + 1$$

$$= \dots$$

10 进制到 N 进制:

$$(x_{n-1}x_{n-2} \cdots x_1x_0)_N$$

$$= x_{n-1} \times N^{n-1} + x_{n-2} \times N^{n-2} + \cdots + x_1 \times N + x_0$$

$$= (x_{n-1} \times N^{n-2} + x_{n-2} \times N^{n-3} + \cdots + x_1) \times N + x_0$$

例如: $307 = 38 \times 8 + 3$, $38 = 4 \times 8 + 6$, $307 = 463_8$

又如: $1194 = 74 \times 16 + 10$, $74 = 4 \times 16 + 10$

$$1194 = 4AA_{16}$$

再如： $214 = 107 \times 2 + 0$, $107 = 53 \times 2 + 1$

$$53 = 26 \times 2 + 1, \quad 26 = 13 \times 2 + 0$$

$$13 = 6 \times 2 + 1, \quad 6 = 3 \times 2 + 0$$

$$3 = 1 \times 2 + 1, \quad 214 = 11010110_2$$

N 进制到 M 进制：以 10 进制为桥梁！

问题： $1234_5 = ?_7$ 能否直接转换？

三、小数间的换算

N 进制到 10 进制:

$$0.x_{-1}x_{-2}\cdots x_{-n}$$

$$= x_{-1} \times N^{-1} + x_{-2} \times N^{-2} + \cdots + x_{-n} \times N^{-n}$$

例如:

$$\begin{aligned} 0.123_4 &= 1 \times 4^{-1} + 2 \times 4^{-2} + 3 \times 4^{-3} \\ &= 0.421875 \end{aligned}$$

$$\begin{aligned} 0.1C3_{16} &= 1 \times 16^{-1} + 12 \times 16^{-2} + 3 \times 16^{-3} \\ &= 451 / 4096 \end{aligned}$$

$$0.1011_2 = 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4} \\ = 0.6875$$

10 进制到 N 进制:

$$(0.x_{-1}x_{-2}\cdots x_{-n})_N \\ = x_{-1} \times N^{-1} + x_{-2} \times N^{-2} + \cdots + x_{-n} \times N^{-n} \\ = (x_{-1} + x_{-2} \times N^{-1} + \cdots + x_{-n} \times N^{-(n-1)}) \times N^{-1}$$

例如: $0.765 \times 7 = 5 + 0.355$

$$0.355 \times 7 = 2 + 0.485$$

$$0.485 \times 7 = 3 + 0.395$$

$$0.395 \times 7 = 2 + 0.765$$

$$0.765 = \overline{0.5232}$$

又如: $0.252 \times 16 = 4 + 0.032$

$$0.032 \times 16 = 0 + 0.512$$

$$0.512 \times 16 = 8 + 0.192$$

$$0.192 \times 16 = 3 + 0.072$$

$$0.252 = 0.4083126E \dots$$

再如： $0.375 \times 2 = 0 + 0.75$, $0.75 \times 2 = 1 + 0.5$

$$0.5 \times 2 = 1 + 0.0, \quad 0.375 = 0.011_2$$

N 进制到 M 进制：以 10 进制为桥梁！

说明： 小数间未必能彻底转换！ 但有理数仍是有理数，无理数还是无理数。

四、2、8、10、16 进制数间的换算

表2.1.2 二、八、十、十六进制数码对应关系

十进制	二进制	八进制	十六进制	十进制	二进制	八进制	十六进制
0	0	0	0	8	1000	10	8
1	01	1	1	9	1001	11	9
2	10	2	2	10	1010	12	A
3	11	3	3	11	1011	13	B
4	100	4	4	12	1100	14	C
5	101	5	5	13	1101	15	D
6	110	6	6	14	1110	16	E
7	111	7	7	15	1111	17	F

2、8 进制： $2^3 = 8$ ，3 位二进制数对应于 1 位八进制数。

将八进制数转换为二进制数时，只需要将每 1 位八进制数替换为等值的 3 位二进制数，然后去掉整数部分高位的 0，以及小数部分低位的 0 就可以了。

$$\begin{aligned} \text{例如: } 263.74_8 &= 010\ 110\ 011.111\ 100_2 \\ &= 10110011.1111_2 \end{aligned}$$

将二进制数转换为八进制数时，需要从小数点开始往两边，分别对二进制数的整数部分和小数部分按 3 位进行分组，整数部分往左每 3 位一组，左边不足 3 位的用 0 补足，小数部分往右每 3 位一组，右边不足 3 位的用 0 补足，然后将每组的 3 位二进制数替换为等值的 1 位八进制数。

例如：

$$1011101.10111_2 = 001\ 011\ 101.101\ 110_2$$
$$= 135.56_8$$

2、16 进制： $2^4 = 16$ ，4 位二进制数对应于 1 位十六进制数。

类似于二进制数和八进制数之间的转换。

例如：

$$3BA8.E_{16} = 0011\ 1011\ 1010\ 1000.1110_2$$
$$= 11\ 1011\ 1010\ 1000.111_2$$

$$1011101.10111_2 = 0101\ 1101.1011\ 1000_2$$
$$= 5D.B8_{16}$$

2、8、10、16 进制：找合适的桥梁！

五、几种常用的编码

1、BCD 码：用 4 个 2 进制位表示 1 个 10 进制位！

表2.1.3 常用BCD码

十进制数码	8421BCD 码	5421BCD 码	2421BCD 码	余 3 码
0	0000	0000	0000	0011
1	0001	0001	0001	0100
2	0010	0010	0010	0101
3	0011	0011	0011	0110
4	0100	0100	0100	0111
5	0101	1000	1011	1000
6	0110	1001	1100	1001
7	0111	1010	1101	1010
8	1000	1011	1110	1011
9	1001	1100	1111	1100
各位位权值	8, 4, 2, 1	5, 4, 2, 1	2, 4, 2, 1	

8421BCD 码、5421BCD 码和 2421BCD 码各数位都有确定的**位权**值，可以按位权展开求得所代表的十进制数码。

例如： $(1001)_{8421\text{BCD}} = 1 \times 8 + 0 \times 4 + 0 \times 2 + 1 \times 1 = (9)_{10}$

$$(1001)_{5421\text{BCD}} = 1 \times 5 + 0 \times 4 + 0 \times 2 + 1 \times 1 = (6)_{10}$$

$$(1011)_{2421\text{BCD}} = 1 \times 2 + 0 \times 4 + 1 \times 2 + 1 \times 1 = (5)_{10}$$

这些 BCD 码被称为**有权 BCD 码**。其中，因

为 **8421BCD 码** 各位的位权值与二进制数各位的位权值一致，所以应用最为普遍。

余 3 码 的各数位并没有确定的位权值，因此被称为 **无权 BCD 码**。从 8421BCD 码和余 3 码的编码可以看出，在有效编码范围内，8421BCD 码加 3（对应二进制的 0011）就可以获得余 3 码，这一特点能够方便地实现 8421BCD 码和余 3 码之间的转换。

例如：将十进制数 $(93.26)_{10}$ 分别表示成 8421BCD 码和余 3 码。

$$(93.26)_{10} = (1001\ 0011.0010\ 0110)_{8421\text{BCD}}$$

9 3 2 6

$$(93.26)_{10} = (1100\ 0110.0101\ 1001)_{\text{余}3\text{码}}$$

9 3 2 6

解答：

又如：将 $(101110101000.0010)_{8421\text{BCD}}$ 转换为十进制数。

解答： $(101110101000.0010)_{5421\text{BCD}}$

$= (1011'1010'1000.0010)_{5421\text{BCD}}$

$= (875.2)_{10}$

BCD 码表示的十进制数具有二进制码的形式，同时又有十进制数的特点，码组（也就是 4 位二进制数码）之间是逢十进一的。

2、循环码：也称为**格雷码**，是一种无权码，具有

多种形式。但不论是那种循环码，都有一个共同的特点：任意两个相邻码之间只有一位不同。

表 2.1.4 列出了十进制数、二进制数与典型 2 位、3 位、4 位循环码的对应关系。

问题：你能列出 10 进制数与典型 5 位循环码的对应关系吗？

表2.1.4 十进制数、 二进制数与典型2位、 3位、
4位循环码的对应关系

十进制数	二进制数	2 位循环码	3 位循环码	4 位循环码
0	0000	00	000	0000
1	0001	01	001	0001
2	0010	11	011	0011
3	0011	10	010	0010
4	0100		110	0110
5	0101		111	0111
6	0110		101	0101
7	0111		100	0100
8	1000			1100
9	1001			1101
10	1010			1111
11	1011			1110
12	1100			1010
13	1101			1011
14	1110			1001
15	1111			1000

3、奇偶校验码

二进制信息代码在传输或存储过程中，可能会由于噪声或干扰而产生错误，致使某些位由 0 变成了 1，或使某些位由 1 变成了 0。

为了避免或减少错误产生的影响，通常采用信息冗余的方法编码，也就是在原有**信息位**的基础上增加若干位**校验位**，通过这些校验位来检出错误，进而纠正错误。

具有校验位的信息码称为**校验码**，其中能够检出错误的校验码称为**检错码**，而能够发现错误并纠正错误的校验码则称为**纠错码**。

奇偶校验码是应用最多也是最简单的检错码，有**奇校验**和**偶校验**两种方式。奇校验就是在信息位之前或之后增加 1 位校验位，使得校验位与信息位一起构成的**码字**中所含 1 的个数为奇数。而偶校验码则是通过增加 1 位校验位，使码字中所

含 1 的个数为偶数。

表 2.1.5 给出了一些 4 位信息码的奇偶校验码示例，这里约定校验位为码字的高位。

表2.1.5 奇偶校验码示例

4 位信息码	奇校验码	偶校验码
0000	10000	00000
0001	00001	10001
0011	10011	00011
0111	00111	10111
1001	11001	01001
1011	01011	11011
1111	11111	01111

注：校验码中最高位是校验位。

六、有符号整数的原码、反码、补码

计算机中的加法运算、减法运算一般都采用补码进行，下面简单介绍**有符号整数**的原码、反码、补码。

最高位：**符号位**，1 表示负数，0 表示正数

其它位：**数值位**

原码：数值位为绝对值

反码：正数的反码与原码相同；负数的反码在原码的基础上，符号位不变，数值位按位取反

补码：正数的补码与原码相同；负数的补码在反码的基础上，符号位不变，数值位加 1

例如：4 比特有符号整数表

10 进制	2 进制	原码	反码	补码
+7	+111	0,111	0,111	0,111

+6	+110	0,110	0,110	0,110
+5	+101	0,101	0,101	0,101
+4	+100	0,100	0,100	0,100
+3	+11	0,011	0,011	0,011
+2	+10	0,010	0,010	0,010
+1	+1	0,001	0,001	0,001
+0	+0	0,000	0,000	0,000
-0	-0	1,000	1,111	0,000

-1	-1	1,001	1,110	1,111
-2	-10	1,010	1,101	1,110
-3	-11	1,011	1,100	1,101
-4	-100	1,100	1,011	1,100
-5	-101	1,101	1,010	1,011
-6	-110	1,110	1,001	1,010
-7	-111	1,111	1,000	1,001
-8	-1000			1,000

结论：4 比特整数补码表示的 10 进制数范围

-8-----+7

加法： $[x + y]_{\text{补}} = x_{\text{补}} + y_{\text{补}}$

$$[x - y]_{\text{补}} = x_{\text{补}} - y_{\text{补}}$$

减法：

$$= x_{\text{补}} + [-y]_{\text{补}}$$

$$= x_{\text{补}} + [[y]_{\text{补}}]_{\text{补}}$$

作业 01：教材第 66 页，习题 1-8。