计算机组成原理

PRINCIPLES OF COMPUTER ORGANIZATION

第3次课：3.2带符号的二进制数据的表示方法

杜国栋
信息科学与工程学院计算机科学与工程系
gddu@ysu.edu.cn

燕山大學
YANSHAN UNIVERSITY

厚德 · 博学 · 求是

计算机中数据的进制表示与中国传统文化

两弹一星的大数据与小算盘

课程目标

- 掌握将真值表示为原码、反码、补码和移码形式的机器数，或反之；
- 熟悉真值和机器数的概念；
- 了解进位计数制，能够进行不同数制之间的转换。

下列数中最大的数是

$(10010101)_2$

$(227)_8$

$(96)_{16}$

$(143)_5$

下列数中最大的数是

$(10010101)_2$

$(227)_8$

$(96)_{16}$

$(143)_5$

下列数中最小的数是

$(101001)_2$

$(52)_8$

$(00101001)_{BCD}$

$(233)_{16}$

下列数中最小的数是

$(101001)_2$

$(52)_8$

$(00101001)_{BCD}$

$(233)_{16}$

预备知识：数的相关概念

- 真值：±12.5、 ±83、 ±0.256；
- 机器数：数字化（0或1）的真值，可由$x_n x_{n-1} \ldots x_1 x_0$ 表示；
- 机器字长：计算机一次（1条指令）所能处理的最大数据长度；
- 机器数的编码：原码、反码、补码、移码；
- 进制的表示：二进制、八进制、十进制、十六进制

$(00101010.1)_{2/8/10/16}$ **或** 00101010.1B/Q/D/H

预备知识：进制转换

- 任意进制转换为十进制：
- 二进制、八进制、十六进制相互转换

$(\underline{1}\ \underline{101}.\underline{010}\ \underline{1})_2 \rightarrow (\qquad\qquad)_8$

$(\underline{1}\ \underline{1101}.\underline{0101})_2 \rightarrow (\qquad\qquad)_{16}$

$(15.24)_8 \rightarrow (\qquad\qquad)_{16}$

$(\underline{1}\ \underline{101}.\underline{010}\ \underline{1})_2 \rightarrow (\ 15.24\ )_8$

$(\underline{1}\ \underline{1101}.\underline{0101})_2 \rightarrow (\ 1D.5\ )_{16}$

$(15.24)_8 \rightarrow (\ D.5)_{16}$

预备知识：进制转换

- 任意进制转换为十进制:
- 二进制、八进制、十六进制相互转换
- 十进制转换为任意进制

$$(725.9325)_{10}=(\qquad\qquad)_2=(\qquad\qquad)_8=(\qquad\qquad)_{16}$$

$(725.9325)10=(1011010101.1110)_2=(1325.7)_8=(2D5.E)_{16}$

预备知识：进制转换

- 任意进制转换为十进制:
- 二进制、八进制、十六进制相互转换
- 十进制转换为任意进制
- BCD码（Binary-Coded Decimal，用二进制表示十进制）-8421码（有权）

0000	0	0001	1	0010	2		
0011	3	0100	4	0101	5		
0110	6	0111	7	1000	8	1001	9

1010~1111之间的6个编码被抛弃;

预备知识：进制转换

- BCD码（Binary-Coded Decimal，用二进制表示十进制）-8421码（有权）

① 如果任何两个对应位BCD数相加的结果向高一位无进位，此时可能有两种情况：若得到的结果小于或等于9，则该位不需修正($x \le 9$)；若得到的结果大于9且小于16时，该位进行加6修正($9 < x < 16$)。

② 如果任何两个对应位BCD数相加的结果向高一位有进位时(即结果大于或等于16)，该位进行加6修正($x \ge 16$)。

44+27=71

```
  0100   0100
 +0010   0111
 -------------
  0110   1011
 +       0110
 -------------
  0110 (1)0001
  0111    0001
```

99+99=198

```
       1001   1001
      +1001   1001
 ------------------
 (+1)  0011   0010
      +0110   0110
 ------------------
 (+1)  1001   1000
     100+98
```

带符号数的表示（数的机器码表示）

- 真值与机器数

真值：按一般书写形式表示的原值，在计算机技术中称为真值。
机器数：机器中使用的连同数符一起数码化的数就称为机器数。
正号和负号同样用数码0或1表示，约定数的最高位为符号位。

带符号数的表示（数的机器码表示）

➢ 原码的表示法

有符号定点整数 $\underline{x_n x_{n-1} \cdots x_1 x_0}$

$$[x]_{原} = \begin{cases} 0, x & 0 \le x < 2^n \\ 2^n - x & -2^n < x \le 0 \end{cases}$$

其中，x 为真值（带 ± 号），n 为该数数值位数;

例:

x=+1110 时，$[x]_{原} = 0,1110$

x=-1110 时，$[x]_{原} = 2^4 - (-1110) = 10000 + 1110 = 1,1110$

有符号定点纯小数 $x_n \bullet \underline{x_{n-1} \cdots x_1 x_0}$

$$[x]_{原} = \begin{cases} x & 0 \le x < 1 \\ 1 - x & -1 < x \le 0 \end{cases}$$

其中，x 为真值（带 ± 号），n 为该数数值位数;

例:

x=+0.1101 时，$[x]_{原} = 0.1101$

x=-0.1101 时，$[x]_{原} = 1 - (-0.1101) = 1.0000 + 0.1101 = 1.1101$

带符号数的表示（数的机器码表示）

➢ 补码的表示法

即逆时针拨 2 格，顺时针方向拨 10 格。

我们称在模 12 的情况下，-2 与 10 互为补数。

$$-2 \equiv 10 \qquad (\text{mod}12)$$

现在是北京时间3点整，而时钟却指向5点。

5-2=3

5+10=3（12自动丢失。12就是模）

模： 也称为溢出量，例如 16 进制（4 位二进制表示）时，16（10000）就为溢出量。

若机器数（字长）格式 $x_n \underline{x_{n-1} \cdots x_1 x_0}$，其溢出量为 $2^{(n+1)}$

在此基础上，存在负数转换为其补数的可能，即负数$+2^{(n+1)}$

带符号数的表示（数的机器码表示）

- 补码的表示法

有符号定点整数 $x_n\underline{x_{n-1}\cdots x_1x_0}$

$$[x]_{补}=\begin{cases}0,x & 0\le x<2^n\\ 2^{n+1}+x & -2^n\le x<0\end{cases}\qquad (mod\ \ 2^{n+1})$$

其中，x 为真值（带 ± 号），n 为该数数值位数;

例:

x=+1110 时，$[x]_{补}=0,1110$

x=-1110 时，$[x]_{补}=2^5+(-1110)=100000-1110=1,0010$

有符号定点纯小数 $x_n\bullet\underline{x_{n-1}\cdots x_1x_0}$

$$[x]_{补}=\begin{cases}x & 0\le x<1\\ 2+x & -1\le x<0\end{cases}\qquad (mod\ \ 2)$$

其中，x 为真值（带 ± 号），n 为该数数值位数;

例:

x=+0.1101 时，$[x]_{补}=0.1101$

x=-0.1101 时，$[x]_{补}=2+(-0.1101)=10.0000-0.1101=1.0011$

带符号数的表示（数的机器码表示）

- 补码的表示法

当 x=0 时，

$$[+0.0000]_{补} = 0.0000$$

$$[-0.0000]_{补} = 2 + (-0.0000) = 10.0000 = 0.0000 \qquad (\bmod 2)$$

由此可见，$[+0]_{补} = [-0]_{补}$，即补码中零的形式唯一。

$$[-1]_{补} = 2 + (-1) = 10.0000 - 1.0000 = 1.0000$$

求补码 1.0001 的真值？

$$x = [x]_{补} - 2 = 1.0001 - 10.0000 = -0.1111$$

1：$[x]_{补}$=1000，求x？ x= ？

2：$[x]_{补}$=010011011，求x？ x= ？

3：$[x]_{补}$=110011011，求x？ x= ？

例：[x]补=1000，求x?

x=-8　（111）->1000

例：[x]补=010011011，求x?

x=155　（10011011）128+16+8+2+1

例：[x]补=110011011，求x?

x=-101　01100100->01100101　64+32+4+1=101

若[X]补=10000H，则 X=　？？？？

带符号数的表示（数的机器码表示）

➢ 反码的表示法

有符号定点整数 $x_n\underline{x_{n-1}\cdots x_1x_0}$

$$[x]_{反}=\begin{cases}0,x & 0\le x<2^n\\(2^{n+1}-1)+x & -2^n<x\le 0\end{cases}\qquad (mod\ \ 2^{n+1}\text{-}1)$$

其中，x 为真值（带 ± 号），n 为该数数值位数;

例:

x=+1110 时，$[x]_{反}=0,1110$

x=-1110 时，$[x]_{反}=2^5+(\text{-}1110)=100000\text{-}0001-1110=1,0001$

有符号定点纯小数 $x_n\bullet\underline{x_{n-1}\cdots x_1x_0}$

$$[x]_{反}=\begin{cases}x & 0\le x<1\\(2\text{-}2^{\text{-}n})+x & -1<x\le 0\end{cases}\qquad (mod\ \ 2\text{-}2^{\text{-}n})$$

其中，x 为真值（带 ± 号），n 为该数数值位数;

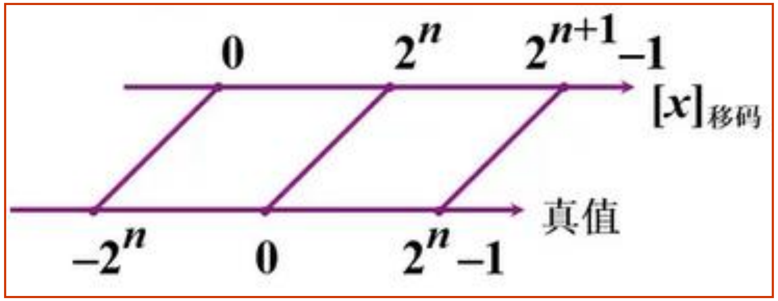
例:

x=+0.1101 时，$[x]_{反}=0.1101$

x=-0.1101 时，$[x]_{反}=2-0.0001+(\text{-}0.1101)=10.0000-0.1101=1.0010$

带符号数的表示（数的机器码表示）

$0 \le [x]_{移} \le 255$



$-128 \le x \le 127$

- 移码的表示法

当真值用补码表示时，真值和符号位一起编码，与习惯上的表示不同，人们很难从补码形式上直接判断其真值的大小，例如

x=21 时，对应的二进制为+10101，则[x]补=0,10101

x=-21 时，对应的二进制为-10101，则[x]补=1,01011

从人们习惯上看，1,01011>0,10101，但事实正好相反，所以为了解决此类问题（抑或浮点数阶码问题），引入移码的概念。

$$[x]_{移} = 2^n + x \qquad -2^n \le x < 2^n$$

带符号数的表示（数的机器码表示）

➢ 移码的表示法

移码与补码之间的关系：

由移码和补码的定义可得：

$$[x]_{移} = \begin{cases} [\mathrm{x}]_{补} + 2^{n} & 0 \le x < 2^{n} \\ [\mathrm{x}]_{补} - 2^{n} & -2^{n} \le x < 0 \end{cases}$$

由此可得，x 真值对应的移码等于 x 真值对应的补码的符号位取反。

带符号数的表示（数的机器码表示）

➤ 移码的表示法

真值 x (n=5)	$[x]_{补}$	$[x]_{移}$	$[x]_{移}$对应的十进制整数
-100000	100000	000000	0
- 11111	100001	000001	1
- 11110	100010	000010	2
⋮	⋮	⋮	⋮
- 00001	111111	011111	31
± 00000	000000	100000	32
+ 00001	000001	100001	33
+ 00010	000010	100010	34
⋮	⋮	⋮	⋮
+ 11110	011110	111110	62
+ 11111	011111	111111	63

码值为80H

若为真值0，则为 [填空1] 码；若为真值-127，则为 [填空2] 码；

若表示-128，则为 [填空3] 码；若表示-0，则为 [填空4] 码；

码值为80H

若为真值0，则为（）码；若为真值-127，则为（）码；

若表示-128，则为（）码；若表示-0，则为（）码；

80H 1000 0000 移码 反码 补码 原码

码值为FFH

若为真值127，则为 [填空1] 码；若为真值-127，则为 [填空2] 码；

若表示-1，则为 [填空3] 码；若表示-0，则为 [填空4] 码；

码值为FFH

若为真值127，则为（）码；若为真值-127，则为（）码；

若表示-1，则为（）码；若表示-0，则为（）码；

FFH 1111 1111 移码 原码 补码 反码

若小数点约定在8位二进制数的最右端（整数），试分别写出下列各种情况下的W、X、Y和Z的真值。

（1）$[W]_{补}=[X]_{原}=[Y]_{反}=[Z]_{移}=00H$

（2）$[W]_{补}=[X]_{原}=[Y]_{反}=[Z]_{移}=80H$

（3）$[W]_{补}=[X]_{原}=[Y]_{反}=[Z]_{移}=FFH$

若小数点约定在8位二进制数的最右端（整数），试分别写出下列各种情况下的W、X、Y和Z的真值。

（1）$[W]_{补}=[X]_{原}=[Y]_{反}=[Z]_{移}=00H$

（2）$[W]_{补}=[X]_{原}=[Y]_{反}=[Z]_{移}=80H$

（3）$[W]_{补}=[X]_{原}=[Y]_{反}=[Z]_{移}=FFH$

（1）X=W=Y=0 Z=-128

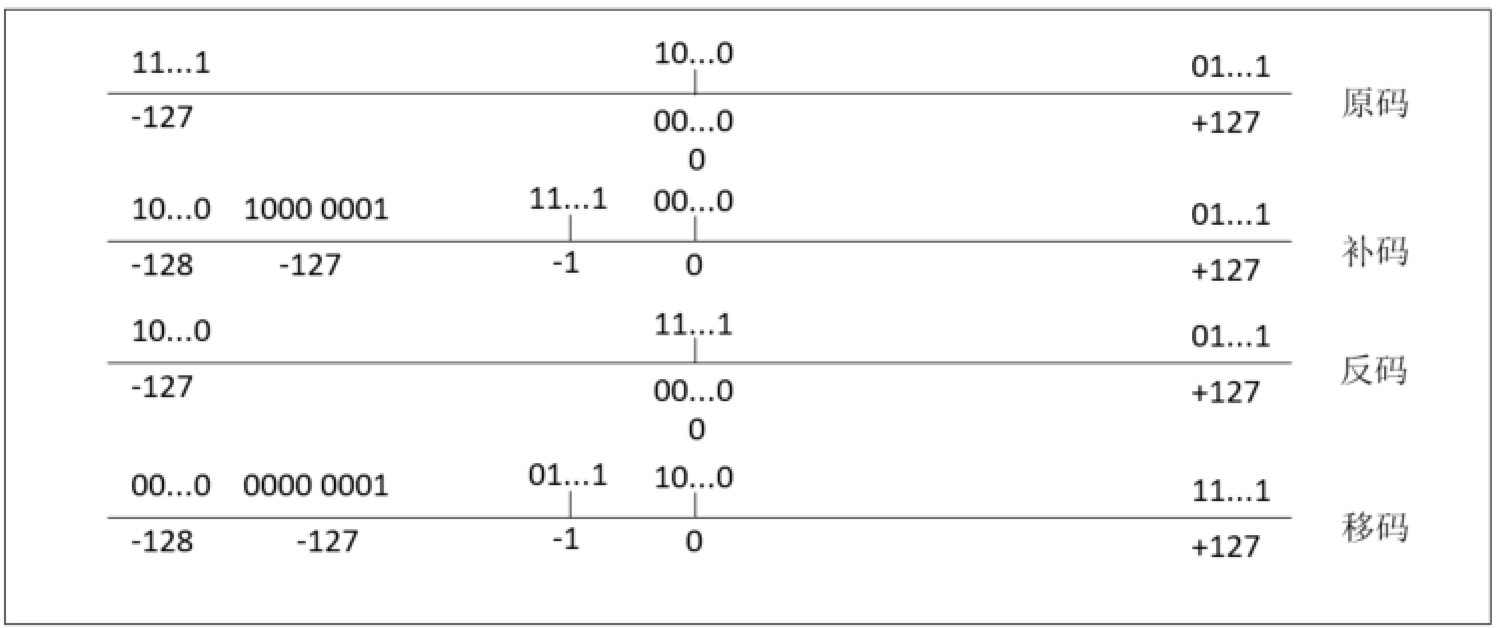
（2）X=-0 Y=-127 W=-128 Z=0

（3）X=-127 Y=-0 W=-1 Z=127

字长为8位，机器数的移码为2^7、0和255的真值分别为：

字长为8位，机器数的移码为2^7、0和255的真值分别为：

0　-128　+127



已知定点小数x的反码为$1.x_1x_2x_3$，且$x<-0.75$，则必须有（　）

A．$x_1=0$，$x_2=0$，$x_3=1$

B．$x_1=1$

C．$x_1=0$，且x_2，x_3不全为0

D．$x_1=0$，$x_2=0$，$x_3=0$

已知定点小数x的反码为$1.x_1x_2x_3$，且$x<-0.75$，则必须有（　）

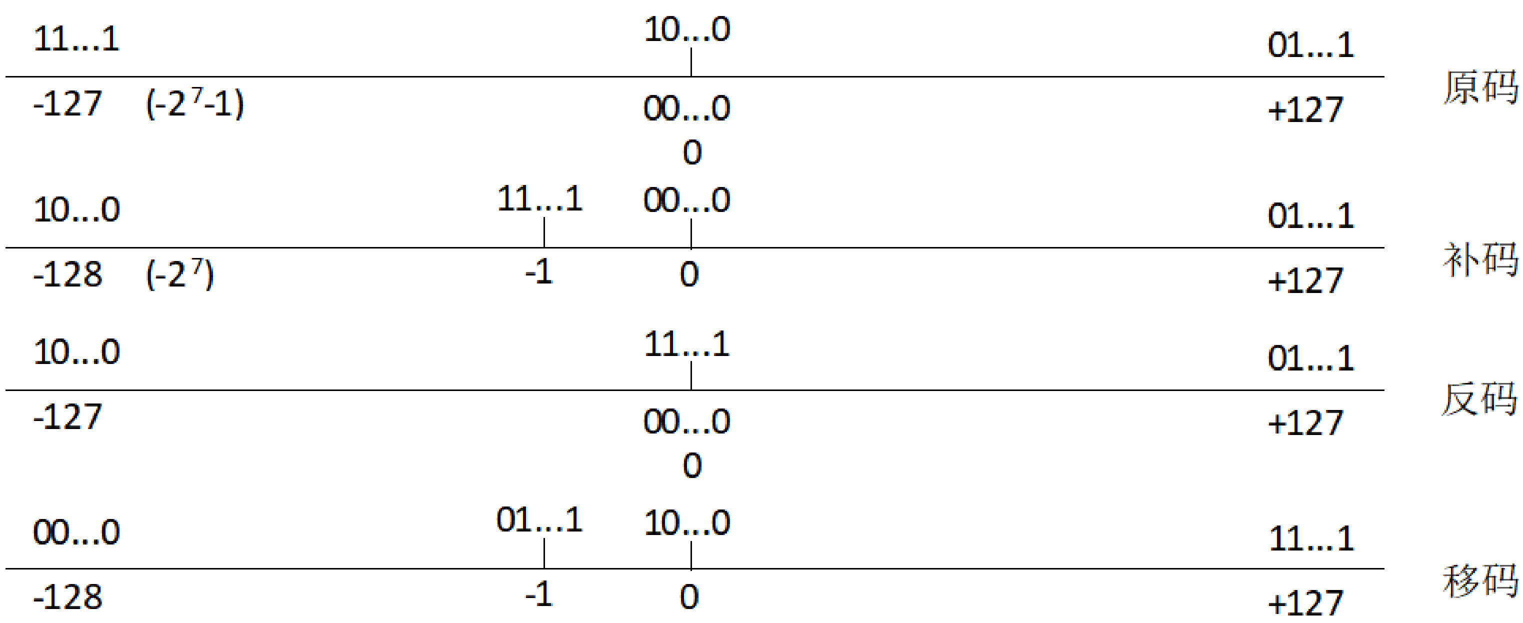
A．$x_1=0$，$x_2=0$，$x_3=1$ ==➔ $[x]_{反}=1.001$ $x=-0.75$

B．$x_1=1$ ==➔ $[x]_{反}=1.1x_2x_3$ $x<-0.5$

C．$x_1=0$，且x_2，x_3不全为0

D．$x_1=0$，$x_2=0$，$x_3=0$ ==➔ $[x]_{反}=1.000$ $x=-0.875$

总结



课后习题：P67 3.4 3.5 3.6 3.12

有问题欢迎随时跟我讨论

办公地点：西校区信息馆423
邮　　箱：gddu@ysu.edu.cn