

扩展第4讲 机器是怎样处理符号与小数点的？ - -机器数与定浮点数

战 德 臣

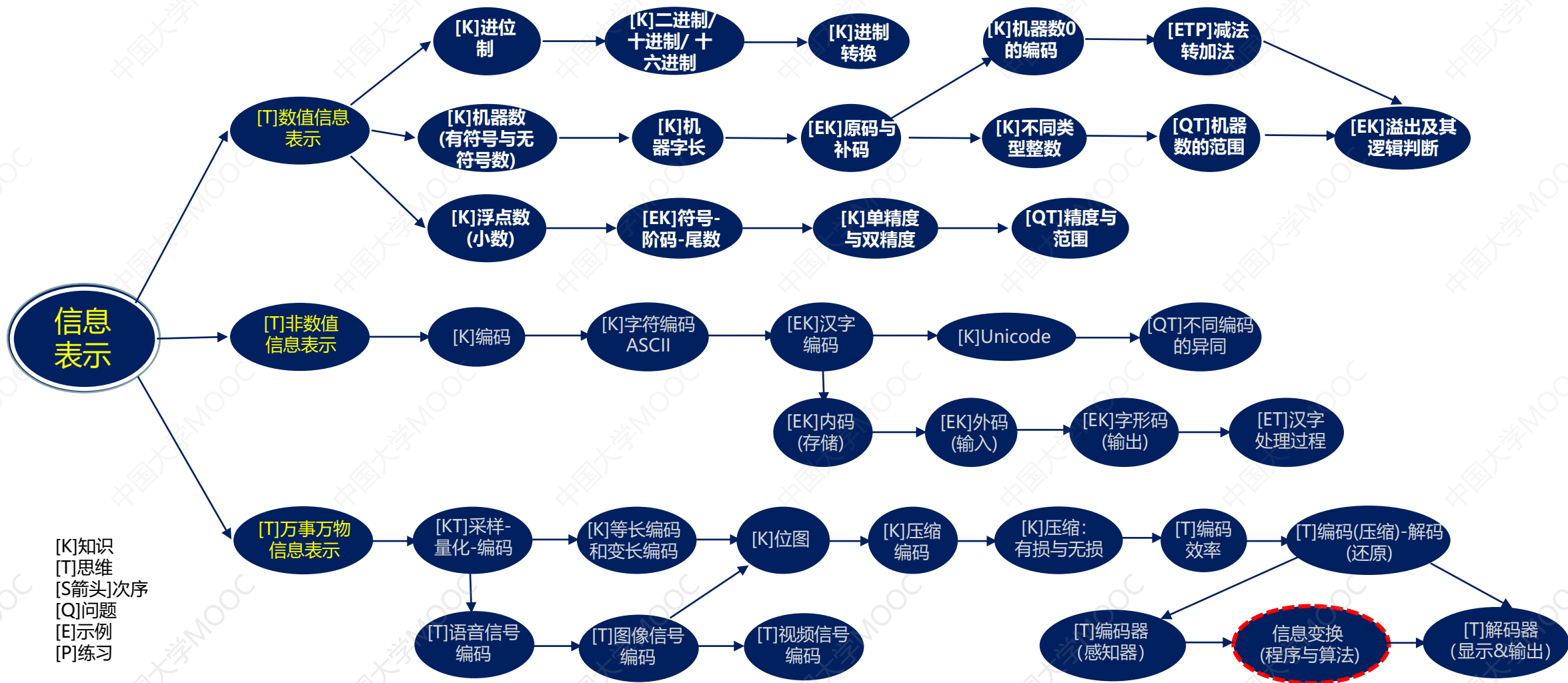
哈尔滨工业大学计算学部教学委员会主任
国家教学名师

18686783018, dechen@hit.edu.cn

信息在计算机中的基本表示方法

2

概览



计数制

3

十进制、二进制、十六进制

日常使用

十进制

- 有0,1,2,3,4,5,6,7,8,9共十个数码
- 数码的位置规定了数码的等级“权/数位”： 10^i
- 逢十进一、借一当十。高数位的1相当于低数位的10
- 表示法：128、 $(128)_+$ 、 $(128)_D$

机器内部
使用

二进制

- 有0,1共两个数码
- 数码的位置规定了数码的等级“权/数位”： 2^i
- 逢二进一、借一当二。高数位的1相当于低数位的2
- 表示法： $(01001000)_2$ 、 $(01001000)_B$

要熟悉和习
惯二进制和
十六进制

辅助阅读
机器数据

十六
进制

- 有0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F共16个数码
- 数码的位置规定了数码的等级“权/数位”： 16^i
- 逢十六进一、借一当十六。高数位的1相当于低数位的16
- 与二进制转换简单：小数点开始向左向右分别处理，1位十六进制数对应为4位二进制数，或4位二进制数对应1位十六进制数。
- 表示法： $(15FE)_{+六}$ 、15FEH、**0x15FE** → **0001 0101 1111 1110**

数值性信息与机器数

机器数：机器字长

【机器字长】是指机器内部进行数据处理、信息传输等的基本单元所包含的二进制位数，通常是8位、16位、32位、64位等。

例：(78)_十转换成二进制数。不同字长的表示，结果是不同的

$$(1) \quad (78)_{十} = (1001110)_二$$

$$(2) \quad (78)_{十} = (01001110)_二$$

$$(3) \quad (78)_{十} = (00000000 \ 01001110)_二$$

$$(4) \quad (78)_{十} = (00000000 \ 00000000 \ 00000000 \ 01001110)_二$$

未考虑机器字长

8位机器字长

16位机器字长

32位机器字长

机器
字长

数的表
示范围

溢出

数值性信息与机器数

5

机器数：有符号数和无符号数

假设：16位机器字长

无符号数

无符号机器数的表示范围： $0 \leq X < 2^{16}$ ，即0至65,535

(00000000 00000000) = ——— (11111111 11111111) =

数的符号也用0和1表达，0表“+”号，1表“-”号

有符号数

有符号机器数的表示范围： $-2^{15} < X < 2^{15}$ ，即-32,767至+32,767

(**0**0000000 00000000) = (01111111 11111111) =

(**1**0000000 00000000) = (11111111 11111111) =

有符号数
的大小？

有符号数的
表示范围？

不同的机器数表示方法

带符号的机器数：原码、反码和补码

十进制数	正数 + 244	- 244 负数
二进制数	+ 1 1 1 1 0 1 0 0	- 1 1 1 1 0 1 0 0
机器数-原码	0 1 1 1 1 0 1 0 0	1 1 1 1 1 0 1 0 0
机器数-反码	0 1 1 1 1 0 1 0 0	1 0 0 0 0 1 0 1 1
		加 0 0 0 0 0 0 0 1
机器数-补码	0 1 1 1 1 0 1 0 0	1 0 0 0 0 1 1 0 0

不同的机器数表示方法

机器数：原码、反码和补码

正数

原码、反码和补码是一样的

负数

原码、反码和补码是不同的表示

原码、反码和补码的表示范围也是不同的

真实数值 (带符号的 n 位 二进制数)	十进制数	机器数($n+1$ 位二进制数,其中第 $n+1$ 位表符号,0 表示正号,1 表示负号)		
		原码	反码	补码
+ 11...11	$+(2^n-1)$	0 11...11	0 11...11	0 11...11
+ 10...00	$+2^{n-1}$	0 10...00	0 10...00	0 10...00
+ 00...00	+0	0 00...00	0 00...00	0 00...00
- 00...00	-0	1 00...00	1 11...11	0 00...00
- 10...00	-2^{n-1}	1 10...00	1 01...11	1 10...00
- 11...11	$-(2^n-1)$	1 11...11	1 00...00	1 00...01
-100...00	-2^n	-	-	1 00...00
		正数的原码、反码同补码形式是一样的。最高位为 0 表示正数		
		负数的最高位为 1 表示负数。其余同真实数值的二进制数。	负数的最高位为 1 表示负数。其余在真实数值的二进制数基础上逐位取反。	负数的最高位为 1 表示负数。其余在反码基础上最低位加 1 后形成的。它的负数不包括 0, 但包括 -2^n
		机器数由于受到表示数值的位数的限制,只能表示一定范围内的数。超出此范围则为“溢出”		

为什么要有补码:(1) 0只有一种表示,可多表示一个数;(2) 运算时符号位可一起参与运算;(3)减法可当作加法进行

减法可用加法实现

8

补码的加减法：符号位与数值位一起参与运算

5位机器字长（1位符号位，4位数值位）

$$(+7) + (+3) = (+10)$$

$$\begin{array}{r} \text{---} \\ 0 \ 0111 \\ +) 0 \ 0011 \\ \hline 0 \ 1010 \end{array}$$

$$(10) + (-3) = (7)$$

$$\begin{array}{r} \text{---} \\ 0 \ 1010 \\ +) 1 \ 1101 \\ \hline 0 \ 0111 \end{array}$$

$$(-5) + (-7) = (-12)$$

$$\begin{array}{r} \text{---} \\ 1 \ 1011 \\ +) 1 \ 1001 \\ \hline 1 \ 0100 \end{array}$$

受字长限制自动舍弃

补码再求补
即为原码

$$(-7) + (-12) = \text{溢出}$$

$$\begin{array}{r} \text{---} \\ 1 \ 1001 \\ +) 1 \ 0100 \\ \hline 0 \ 1101 \end{array}$$

小数点的处理

定点数：纯小数 vs. 纯整数

$(+ 245.25)_+$

+ 1 1 1 1 0 1 0 1 . 0 1 0 0
0 1 1 1 1 0 1 0 1 . 0 1 0 0

0 1 1 1 1 0 1 0 1 0 1 0 0

小数点默认在此位置(无需用符号表示), 机器中全部是小数

0 1 1 1 1 0 1 0 1 0 1 0 0

小数点默认在此位置(无需用符号表示), 机器中全部是整数

定点数

小数点的处理

10

浮点数：小数点位置的处理

$$(+ 245.25)_+ = 0.24525 \times 10^3$$

+ 11110101.0100

0 11110101.0100

0 0.111101010100 $\times 2^8$

0 **0.**111101010100 \times **2⁰**1000

0 **01000**

111101010100

符号

指数/阶码

?多少位; ?数的范围

尾数

?多少位; ?精度

浮点数的
指数又被称为
阶码

浮点数

小数点的处理

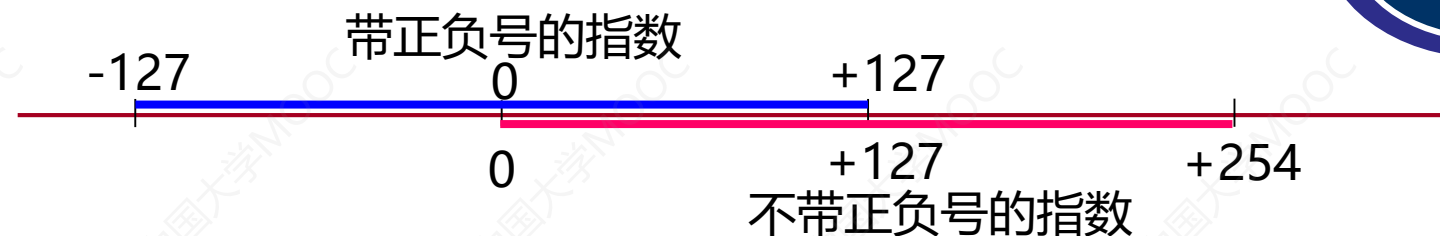
11

浮点数：是否还能提高精度



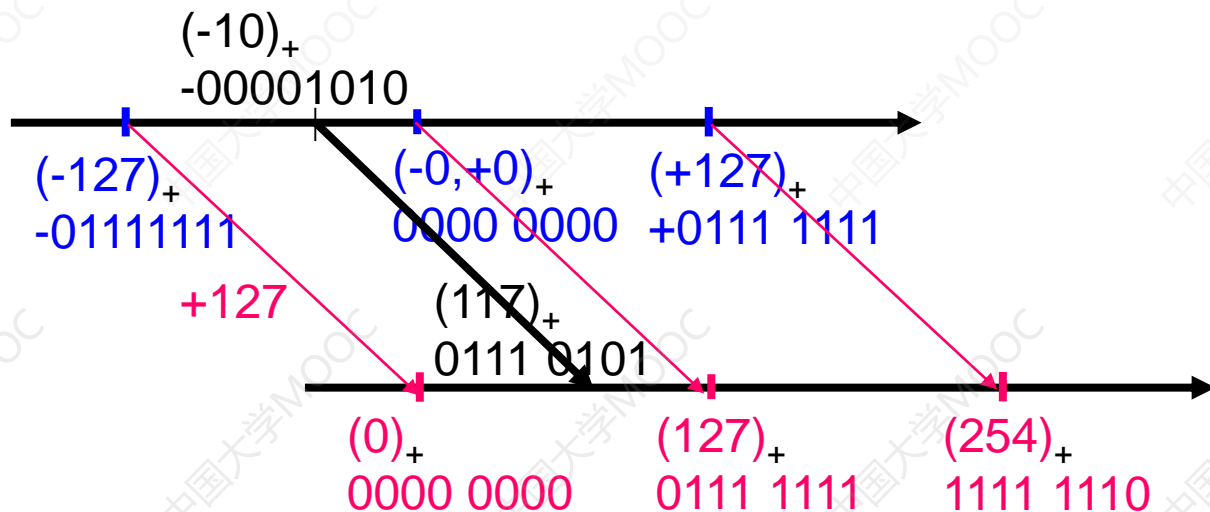
能否去掉指数的符号

是否可多表示1位尾数？



将指数平移 (均加上 2^7-1) : -127--+127的范围, 变为0—254, 避免指数符号与整个数符号的混淆

$$\begin{aligned} & -0.11101 \times 2^{-1001} \\ & -1.1101 \times 2^{-1010} \end{aligned}$$



默认存在, 但不存储

$$-1.1101 * 2^{-1010}$$

加 (2^7-1) 后得到



小数点的处理

12

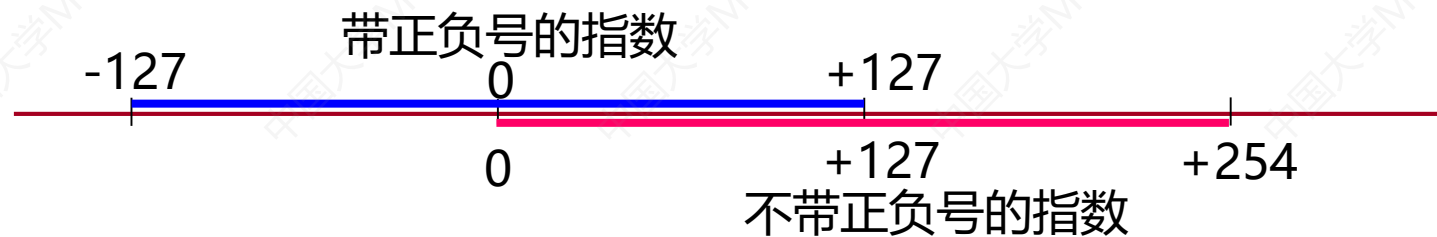
浮点数：单精度数 vs. 双精度数

S	指数(8位)	尾数(后23位)
---	--------	----------

浮点数，32位表示单精度数(相当于科学计数法 $1.x \times 2^y$)
(S为符号位, x为23位尾数, y为8位指数/阶码)

S	指数(11位)	尾数(后52位)
---	---------	----------

浮点数，64位表示双精度数(相当于科学计数法 $1.x \times 2^y$)
(S为符号位, x为52位尾数, y为11位指数/阶码)



将指数平移：-127---+127的范围，变为0—254，避免指数符号与整个数符号的混淆

小数点的处理

13

浮点数：小结—同学仍需要延展学习与思考一下

数值的小数点的处理：定点数与浮点数

S (全为小数)

定点数，小数点位置固定(默认在符号位S的后面)

S (全为整数)

定点数，小数点位置固定(默认在尾部)

S 指数(8位) 尾数(后23位)

浮点数，32位表示单精度数(相当于科学计数法 $1.x \times 2^y$)
(S为符号位，x为23位尾数，y为8位指数)

S 指数(11位) 尾数(后52位)

浮点数，64位表示双精度数(相当于科学计数法 $1.x \times 2^y$)
(S为符号位，x为52位尾数，y为11位指数)

深入：浮点数的
0真的是0吗？
例如 “x==0”
是真还是假？

深入：单精度和
双精度数的表示
范围与表示精度
究竟是多少？

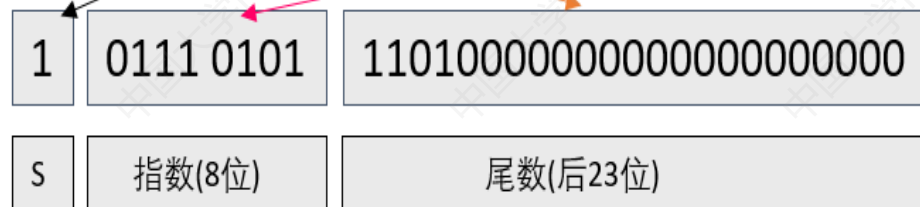
IEEE754有
很多设计特
色与细节

细节：指数值全
0或全1时表示什
么含义（同学自
主学习）

默认存在，但不存储

- 1.1101 * 2^{-1010}

加(2^7-1)后得到



数值性信息表示

14

为什么要学习数值性信息表示？

- 高级语言编程时，为什么要声明变量的类型？
- 例如：C语言
 - 占用存储单元数目-机器字长。不同类型的字长是不同的
 - 数值的表示方法，不同类型的数值表示是不同的
 - 不同类型的变量在处理时会有一些约束（例如与0比较时）

