

# 第二章 信息的表示和处理

2025年5月5日 20:06

## 2.1 信息的表示和处理

### 2.1.1 位、整数

一个字节由8位组成 (1Byte=8bit)。

将给定的二进制数字转换为十六进制时，若位总数不为4的倍数，在左边补0。

将十进制转化为十六进制：

整数：

整数部分不断除以基数k(2、8或16)，并记下余数，直到商为0为止

由最后一个余数起，逆向取各个余数，则为转换成的数

(二进制用后缀字母B,十六进制数用后缀字母H)

小数：

不断乘基数k，记录整数部分，直到小数部分为0

(可能无法乘尽，可选取精度)

布尔代数：与(&)、或(|)、非(~)、异或(^)，按位操作

$\sim x + 1 == -x$ ， $-x$ 为x的补码

C语言运算：所有非0值视为逻辑真(True)

$x \ll y$ ：x向左移动y位，右侧补0

$x \gg y$ ：逻辑右移：x向右移动y位，左侧补0

算术右移：复制左侧最高位y次

类型转换：

T2B (Two's Complement to Binary)：二进制补码转换为普通二进制

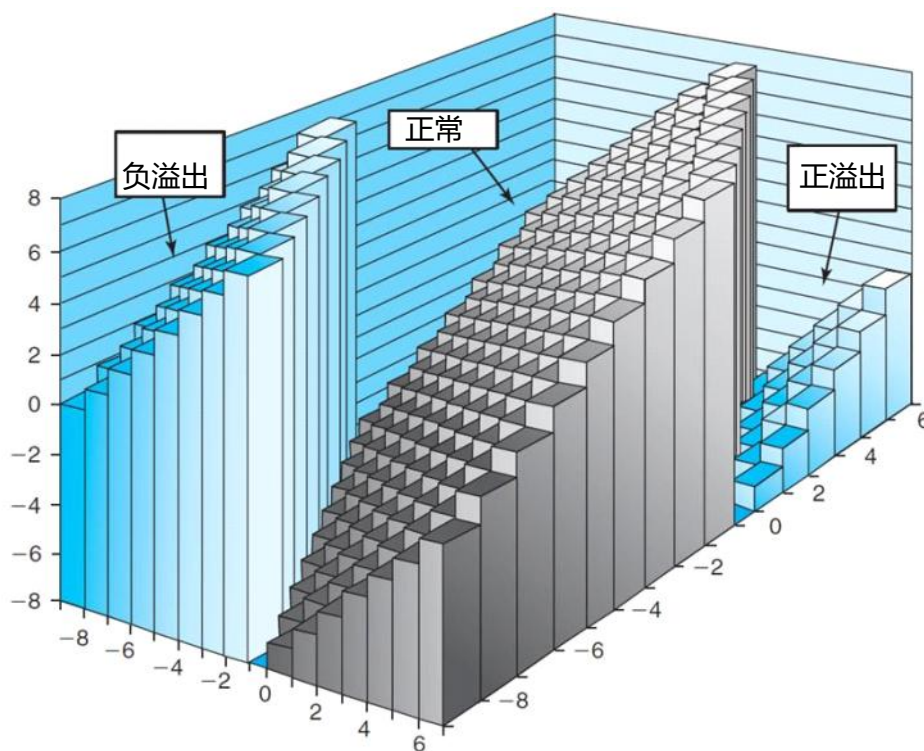
B2U (Binary to Unsigned)：二进制转换为无符号数

表达式中有符号和无符号数混用时，有符号数隐式转换为无符号数

扩展：将最高有效位复制至最左侧直至到达要求位数（无符号数填充0）

截断：丢弃多出的最高位直至到达要求的位数

无符号数加法：可能发生溢出，操作数最高位w位，真实和可能为w+1位，溢出时丢弃最高有效位



### 2.1.2 浮点数

浮点数的表示形式:  $(-1)^s \cdot M \cdot 2^E$

s: 符号, 0或1

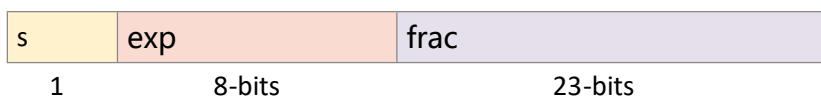
M: 尾数, (frac字段编码), 二进制小数,  $M \in [1.0, 2.0)$

E: 阶码, (exp字段编码), 采用偏置值编码:  $E = \text{Exp} - \text{Bias}$

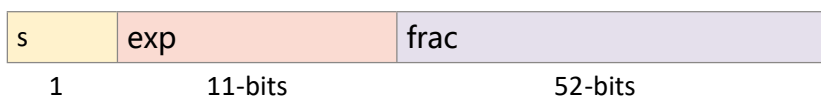
Exp: exp字段的无符号数值

Bias:  $\text{Bias} = 2^{k-1} - 1$ , k为阶码的位数

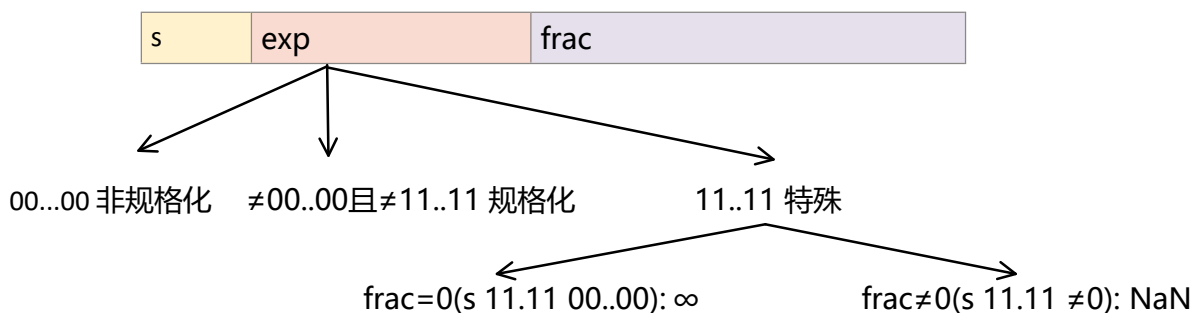
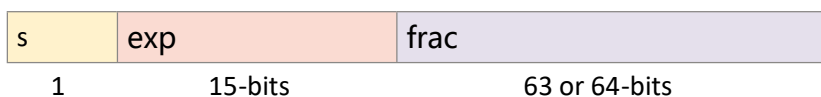
单精度(32bits):

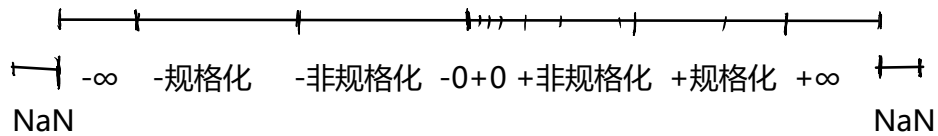


双精度(64bits):



扩展精度(80bits):





分布：靠近零点处密集

规格化编码示例：float F = 15213.0

$$15213_{(10)} = 11\ 1011\ 0110\ 1101_{(2)} = 1.1101101101101_2 \times 2^{13}$$

$$\because \text{exp 为 8 位} \quad \therefore \text{Bias} = 2^7 - 1 = 127, \quad \text{Exp} = 127 + 13 = 140 = 10001100_2$$

$$\therefore 0\ 10001100\ 110110110110100000000000$$

向偶数舍入：

当要舍去的部分严格小于（大于）中间值（0.5）时：向下（向上）舍入。

当要舍去的部分恰好等于中间值（0.5）时，检查保留的最后一位数字：

为偶数（0），向下舍入；为奇数（1），向上舍入。

浮点乘法：  $s = s_1 \wedge s_2$ ;  $M = M_1 \times M_2$ ;  $E = E_1 + E_2$ ; (E超出范围则溢出)

修正：  $M \geq 2$ : M右移一位,  $E + 1$