

原码反码补码

在计算机中，数据通常以二进制形式表示，二进制又可以用原码、反码和补码方式来表示。如果只有正数和加法运算，使用原码即可，反码、补码的出现是为了解决负数和减法所带来的问题。计算机信息存储的最小单位是byte（8位）。为了便于理解，接下来用带符号位的四位二进制数举例说明。

0. 真值

真值是指在计算机中实际表示的数值，它可以通过原码、反码或补码转化得到。

对于无符号整数，真值与原码相同；

对于有符号整数，真值可通过补码表示。

1. 原码

定义：原码是一种直接表示数字符号和大小的二进制编码方法。

符号位：最高位为符号位，0 表示正数，1 表示负数。

数值部分：剩下的位表示数值大小（绝对值）。

特点：

正数的原码和其二进制形式相同。

负数的原码：符号位为 1，数值部分和正数的绝对值一致。

例子：

+5 的原码：00000101 （8 位表示）。

-5 的原码：10000101 （8 位表示）。

2. 反码

定义：反码是在原码的基础上对数值部分（非符号位）取反（0 变 1，1 变 0）。

正数的反码和原码相同。

负数的反码：符号位保持 1，数值部分逐位取反。

特点：

正数的反码不变。

负数的反码可看作表示一种“对称”结构。

例子：

+5 的反码：00000101 （与原码相同）。

-5 的反码：11111010 （符号位 1，数值部分取反）。

为什么引入反码？（了解）

反码主要是为了引入负数的二进制表示对称性，便于机器理解和处理正负数。

在原码表示中，正数和负数不对称。

比如：+5 的原码是 00000101，-5 的原码是 10000101。

问题 1：双零问题：原码中存在“正零”（00000000）和“负零”（10000000），这对于计算机来说是不便处理的。

问题 2：加减法复杂：使用原码进行加减法需要特别处理符号位，逻辑复杂。

通过引入反码，负数的表示更自然，形成了正负数的“对称性”，减少了加减法运算中的一些逻辑问题。

3. 补码

定义：补码是在反码的基础上，最后一位加 1。

正数的补码和原码相同。

负数的补码：符号位保持 1，数值部分先取反（反码），然后加 1。

特点：

补码是计算机中表示有符号数的主要方式，因为它简化了加减法运算。

补码表示的范围比原码和反码大 1。

补码唯一代表零（00000000），而原码和反码有两个零（正零和负零）。

例子：

+5 的补码：00000101 （与原码相同）。

-5 的补码：11111011 （反码基础上加 1）。

为什么引入补码？（了解）

反码虽然解决了对称性问题，但反码仍然有**双零问题**：00000000 和 11111111 分别表示“正零”和“负零”，这在某些计算中依然是麻烦的。

补码解决的问题

双零问题：在补码表示中，零只有一种表示形式，即 00000000。没有“负零”。

加减法运算统一：补码可以让减法转换成加法，从而统一了加法和减法的运算逻辑。

4、总结对比

数值	原码	反码	补码
5	00000101	00000101	00000101
-5	00000101	11111010	11111011

反码的主要作用是解决负数表示的不对称性，为计算机加减法提供更简单的规则。

补码是在反码基础上进一步优化，解决了双零问题，并简化了计算逻辑，使得现代计算机能够高效地处理有符号数的运算。