链滴

计算机基础知识回顾：原码、反码、补码、移码

作者：angels

原文链接：https://ld246.com/article/1496287434596

来源网站：链滴

前言：

- 所有信息在机器上都是以0、1的方式进行存储的。
- 原码、反码、补码事机器存储的一个具体数字的编码方式。

原码：

- 如果机器字长为n，那么一个数的原码就是用一个n位的二进制数，其中最高位为符号位：正数为，负数为1。剩下的n-1位表示概数的绝对值。一个字节（byte）由8个比特（bit）构成。如果位数不可在符号位后进行补全；
  - X=+00 0001，[X]原= 0000 0001
  - X=-00 00001, [X]原= 1000 0001

位数不够的用0补全。

反码：

- 反码：很好理解，就是原来的数据位上的数取反，但要注意，正数不变，仅负数取反，注意只数据位哦。

为什么有了原码还要有反码的出现呢?

场景： 1-1=0 =》1+（-1）=0

计算机设计之初为了尽量把逻辑电路简单化，机器只有两个数的加操作，而没有减操作，那么机器是何计算两数相减的呢，答案是根据二进制左边第一位的符号位，计算机将符号位也纳入计算范围内：

用原码进行计算：

1-1=1+（-1）=[0000 0001] + [1000 0001] = [1000 0010] = -2(因为左边第一位为符号为)，所以入机器用原码进行减法计算时就会出错。

- X=00 0001，[X]原= 0000 0001 ， [X]反= 0000 0001
- X=-00 0001，[X]原= 1000 0001 ， [X]反= 1111 1110

补码：

- 补码：同样很好理解，正数的补码等于原码等于反码，负数的补码等于负数的反码+1
- X=00 0001，[X]原= 0000 0001 ， [X]反=0000 0001，[X]补=0000 0001
- X=-00 0001，[X]原= 1000 0001 ， [X]反=1111 1110，[X]补=1111 1111
- 0的补码是唯一的，如果机器字长为8那么[0]补=00000000。

移码：

- 移码：补码的符号位取反。
  - X=00 0001 , [X]原= 0000 0001 ， [X]反=0000 0001，[X]补=0000 0001，[X]移=1000 0001
  - X=-00 0001 , [X]原= 1000 0001 ， [X]反=1111 1110，[X]补=1111 1111，[X]移=0111 111