

## 单目按位操作符

（数据都是在内存中处理的所以都变成补码再运算）

与  $\&$  一假即假，有一个0全0

或  $|$  一真即真，有一个1全1

异或  $\wedge$  相同即0，不同即1

## 原码反码补码

内存中存储的是补码

打印在屏幕上的是原码

正整数原，反，补三码相同

原码转换为反码：符号位不变，数值位分别“按位取反”

反码转换为原码也是一样：符号位不变，数值位分别“按位取反”

原码转换为补码：符号位不变，数值位按位取反，末位再加1

补码转换为原码：符号位不变，数值位按位取反，末位再加1。

即补码的补码等于原码

## 移位操作符

<< 左移操作符

>> 右移操作符

右移操作符 移位规则：

首先右移运算分两种：

1. 逻辑移位 左边用0填充，右边丢弃
  2. 算术移位 左边用原该值的符号位填充，右边丢弃 比特
- eg:

正整数:

原，

反，

补

$$n = 15$$

binary:

[illegible]
$$15 = 2^3 + 2^2 + 2^1 + 2^0$$
$$n \gg 2 = ?$$
[illegible]

所以  $n \gg 2 = 7$