

算术/逻辑指令

逻辑运算

- **按位运算(bitwise operation)**:组合两个二进制位串, 在各个列上应用基本运算, 产生一个输出串。
- $10011010 \text{ AND } 11001001 = 1001000$
- AND运算的一个主要用途是将位模式的一部分设为0, 而不影响另一部分。
 - 可以用于过滤RGB格式表示的数字图像的某些颜色。
 - AND运算的一个应用是一个称为**屏蔽(masking)**,这里, 一个称为**掩码(mask)**的操作数决定另一个操作数的那部分会影响结果。
 - AND运算在操作其他类型的**位映射(bit map)**时也很有用。
 - Eg.(1)假设一个八位储存单元被用作一个位映射, 我们希望查明从高端位算起第三位相连系的对象是否存在, 我们只需要将整个字节与掩码00100000进行AND运算, 若本身为0, 则当且仅当该位射从高端位算起第三位为0时, 其他位皆为0.
 - Eg.(2)如果该位映射从高端位算起第一位为1时, 我们不想在破坏其他位的情况下将其改为0, 那么可以将其与掩码11011111进行AND运算。
- OR也可以用于复制一个串的一部分, 但方法是将不复制的地方换为1.
- XOR运算的一个主要作用是将形成一个串的补码(及相对的负数)。任意一个位串与全为1的掩码进行XOR运算, 即可得到其补码。一般用于反转一个GRB图像, 将浅色转化成深色。

循环移位运算及移位运算

- **循环移动(circular shift)**:将右端移出的位置放置在左端的位置上。
- **逻辑运算(logical shift)**:丢弃移出的位, 并用0来填充空位。
- **算术移位(arithmetic shift)**:右移时留出空位(符号位位置), 用他原来的值来填补, 保留符号位不变。

算术运算

- 一些小型CPU只会加法或减法, 因为乘法和除法只是加法和减法的替换。