

让我们为参数 x 和 y 定义运算 $+_w$, 其中 $0 \leq x, y < 2^w$, 该操作是把整数和 $x+y$ 截断为 w 位得到的结果, 再把这个结果看做是一个无符号数。这可以被视为一种形式的模运算, 对 $x+y$ 的位级表示, 简单丢弃任何权重大于 2^{w-1} 的位就可以计算出和模 2^w 。比如, 考虑一个 4 位数字表示, $x=9$ 和 $y=12$ 的位表示分别为 [1001] 和 [1100]。它们的和是 21, 5 位的表示为 [10101]。但是如果丢弃最高位, 我们就得到 [0101], 也就是说, 十进制值的 5。这就和值 $21 \bmod 16=5$ 一致。

我们可以将操作 $+_w$ 描述为:

原理: 无符号数加法

对满足 $0 \leq x, y < 2^w$ 的 x 和 y 有:

$$x +_w y = \begin{cases} x + y, & x + y < 2^w \\ x + y - 2^w, & 2^w \leq x + y < 2^{w+1} \end{cases} \quad (2.11)$$

图 2-22 说明了公式 (2.11) 的这两种情况, 左边的和 $x+y$ 映射到右边的无符号 w 位的和 $x+_w y$ 。正常情况下 $x+y$ 的值保持不变, 而溢出情况则是该和数减去 2^w 的结果。

推导: 无符号数加法

一般而言, 我们可以看到, 如果 $x+y < 2^w$, 和的 $w+1$ 位表示中的最高位会等于 0, 因此丢弃它不会改变这个数值。另一方面, 如果 $2^w \leq x+y < 2^{w+1}$, 和的 $w+1$ 位表示中的最高位会等于 1, 因此丢弃它就相当于从和中减去了 2^w 。

说一个算术运算溢出, 是指完整的整数结果不能放到数据类型的字长限制中去。如等式 (2.11) 所示, 当两个运算数的和为 2^w 或者更大时, 就发生了溢出。图 2-23 展示了字长 $w=4$ 的无符号加法函数的坐标图。这个和是按模 $2^4=16$ 计算的。当 $x+y < 16$ 时, 没有溢出, 并且 $x+_w y$ 就是 $x+y$ 。这对应于图中标记为“正常”的斜面。当 $x+y \geq 16$ 时, 加法溢出, 结果相当于从和中减去 16。这对应于图中标记为“溢出”的斜面。

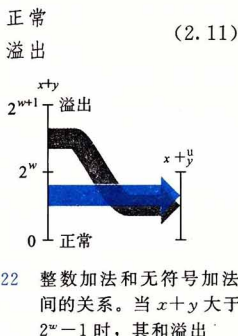


图 2-22 整数加法和无符号加法间的关系。当 $x+y$ 大于 2^w-1 时, 其和溢出

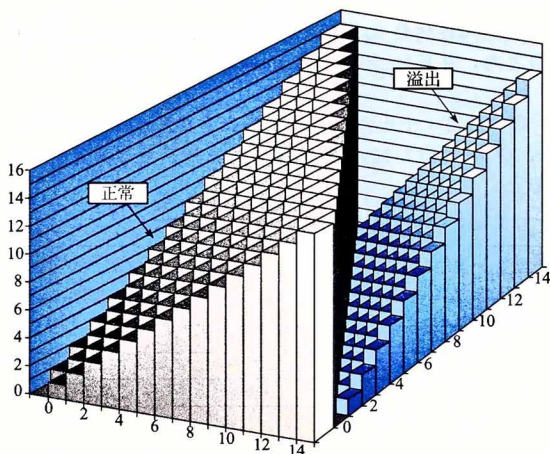


图 2-23 无符号加法 (4 位字长, 加法是模 16 的)