k	>>k（二进制）	十进制	$-12340/2^k$
0	1100111111001100	−12340	−12340.0
1	*1*110011111100110	−6170	−6170.0
4	*1111*110011111100	−772	−771.25
8	*11111111*11001111	−49	−48.203125

图 2-29 进行算术右移（这个例子说明了算术右移类似于除以 2 的幂，除了是向下舍入，而不是向零舍入）

推导：除以 2 的幂的补码除法，向下舍入

设 x 为位模式$[x_{w-1}, x_{w-2}, \cdots, x_0]$表示的补码整数，而 k 的取值范围为 $0\leqslant k<w$。设 x'为 $w-k$ 位$[x_{w-1}, x_{w-2}, \cdots, x_k]$表示的补码数，而 x''为低 k 位$[x_{k-1}, \cdots, x_0]$表示的无符号数。通过与对无符号情况类似的分析，我们有 $x=2^k x'+x''$，而 $0\leqslant x''<2^k$，得到 $x'=\lfloor x/2^k\rfloor$。进一步，可以观察到，算术右移位向量$[x_{w-1}, x_{w-2}, \cdots, x_0]$ k 位，得到位向量

$$[x_{w-1}, \cdots, x_{w-1}, x_{w-1}, x_{w-2}, \cdots, x_k]$$

它刚好就是将$[x_{w-1}, x_{w-2}, \cdots, x_k]$从 $w-k$ 位符号扩展到 w 位。因此，这个移位后的位向量就是$\lfloor x/2^k\rfloor$的补码表示。 ■

我们可以通过在移位之前“偏置(biasing)”这个值，来修正这种不合适的舍入。

原理：除以 2 的幂的补码除法，向上舍入

C 变量 x 和 k 分别有补码值 x 和无符号数值 k，且 $0\leqslant k<w$，则当执行算术移位时，C 表达式 (x+(1<<k)−1)>>k 产生数值$\lceil x/2^k\rceil$。

图 2-30 说明在执行算术右移之前加上一个适当的偏置量是如何导致结果正确舍入的。在第 3 列，我们给出了 −12 340 加上偏量值之后的结果，低 k 位（那些会向右移出的位）以斜体表示。我们可以看到，低 k 位左边的位可能会加 1，也可能不会加 1。对于不需要舍入的情况($k=1$)，加上偏量只影响那些被移掉的位。对于需要舍入的情况，加上偏量导致较高的位加 1，所以结果会向零舍入。

k	偏量	−12 340 + 偏量	>>k（二进制）	十进制	$-12340/2^k$
0	0	1100111111001100	1100111111001100	−12340	−12340.0
1	1	110011111100110*1*	*1*110011111100110	−6170	−6170.0
4	15	110011111101*1011*	*1111*110011111101	−771	−771.25
8	255	11010000*11001011*	*11111111*11010000	−48	−48.203125

图 2-30 补码除以 2 的幂（右移之前加上一个偏量，结果就向零舍入了）

偏置技术利用如下属性：对于整数 x 和 $y(y>0)$，$\lceil x/y\rceil=\lfloor (x+y-1)/y\rfloor$。例如，当 $x=-30$ 和 $y=4$，我们有 $x+y-1=-27$，而$\lceil -30/4\rceil=-7=\lfloor -27/4\rfloor$。当 $x=-32$ 和 $y=4$ 时，我们有 $x+y-1=-29$，而$\lceil -32/4\rceil=-8=\lfloor -29/4\rfloor$。

推导：除以 2 的幂的补码除法，向上舍入

查看$\lceil x/y\rceil=\lfloor (x+y-1)/y\rfloor$，假设 $x=qy+r$，其中 $0\leqslant r<y$，得到$(x+y-1)/y=q+(r+y-1)/y$，因此$\lfloor (x+y-1)/y\rfloor=q+\lfloor (r+y-1)/y\rfloor$。当 $r=0$ 时，后面一项等于 0，而当 $r>0$ 时，等于 1。也就是说，通过给 x 增加一个偏量 $y-1$，然后再将除法向下舍入，当 y 整除 x 时，我们得到 q，否则，就得到 $q+1$。