$TMin_w \sim TMax_w$ 的数，结果得到一个 $0 \sim UMax_w$ 的值，这里两个数有相同的位模式，除了参数是无符号的，而结果是以补码表示的。类似地，对于 $0 \sim UMax_w$ 之间的值 x，定义函数 $U2T_w$ 为 $U2T_w(x) \doteq B2T_w(U2B_w(x))$。生成一个数的无符号表示和 x 的补码表示相同。

继续我们前面的例子，从图 2-15 中，我们看到 $T2U_{16}(-12\ 345)=53\ 191$，并且 $U2T_{16}(53\ 191)=-12\ 345$。也就是说，十六进制表示写作 0xCFC7 的 16 位位模式既是 $-12\ 345$的补码表示，又是 53 191 的无符号表示。同时请注意 $12\ 345+53\ 191=65\ 536=2^{16}$。这个属性可以推广到给定位模式的两个数值(补码和无符号数)之间的关系。类似地，从图 2-14 我们看到 $T2U_{32}(-1)=4\ 294\ 967\ 295$，并且 $U2T_{32}(4\ 294\ 967\ 295)=-1$。也就是说，无符号表示中的 $UMax$ 有着和补码表示的 -1 相同的位模式。我们在这两个数之间也能看到这种关系：$1+UMax_w=2^w$。

接下来，我们看到函数 $U2T$ 描述了从无符号数到补码的转换，而 $T2U$ 描述的是补码到无符号的转换。这两个函数描述了在大多数 C 语言实现中这两种数据类型之间的强制类型转换效果。

练习题 2.19 利用你解答练习题 2.17 时填写的表格，填写下列描述函数 $T2U_4$ 的表格。

x	$T2U_4(x)$
−8	
−3	
−2	
−1	
0	
5	

通过上述这些例子，我们可以看到给定位模式的补码与无符号数之间的关系可以表示为函数 $T2U$ 的一个属性：

原理：补码转换为无符号数

对满足 $TMin_w \leqslant x \leqslant TMax_w$ 的 x 有：

$$T2U_w(x) = \begin{cases} x+2^w, & x<0 \\ x, & x \geqslant 0 \end{cases} \tag{2.5}$$

比如，我们看到 $T2U_{16}(-12\ 345)=-12\ 345+2^{16}=53\ 191$，同时 $T2U_w(-1)=-1+2^w=UMax_w$。

该属性可以通过比较公式(2.1)和公式(2.3)推导出来。

推导：补码转换为无符号数

比较等式(2.1)和等式(2.3)，我们可以发现对于位模式 $\vec{x}$，如果我们计算 $B2U_w(\vec{x})-B2T_w(\vec{x})$之差，从 0 到 $w-2$ 的位的加权和将互相抵消掉，剩下一个值：$B2U_w(\vec{x})-B2T_w(\vec{x})=x_{w-1}(2^{w-1}-(-2^{w-1}))=x_{w-1}2^w$。这就得到一个关系：$B2U_w(\vec{x})=x_{w-1}2^w+B2T_w(\vec{x})$。我们因此就有

$$B2U_w(T2B_w(x)) = T2U_w(x) = x + x_{w-1}2^w \tag{2.6}$$

根据公式(2.5)的两种情况，在 x 的补码表示中，位 x_{w-1}决定了 x 是否为负。 ■

比如说，图 2-16 比较了当 $w=4$ 时函数 $B2U$ 和 $B2T$ 是如何将数值变成位模式的。对补码来说，最高有效位是符号位，我们用带向左箭头的条来表示。对于无符号数来说，最高有效位是正权重，我们用带向右的箭头的条来表示。从补码变为无符号数，最高有效位