

符号	类型	含义
$B2T_w$	函数	二进制转补码
$B2U_w$	函数	二进制转无符号数
$U2B_w$	函数	无符号数转二进制
$U2T_w$	函数	无符号转补码
$T2B_w$	函数	补码转二进制
$T2U_w$	函数	补码转无符号数
$TMin_w$	常数	最小补码值
$TMax_w$	常数	最大补码值
$UMax_w$	常数	最大无符号数
$+^t_w$	操作	补码加法
$+^u_w$	操作	无符号数加法
$*^t_w$	操作	补码乘法
$*^u_w$	操作	无符号数乘法
$-^t_w$	操作	补码取反
$-^u_w$	操作	无符号数取反

图 2-8 整数的数据与算术操作术语。下标 w 表示数据表示中的位数

2.2.1 整型数据类型

C 语言支持多种整型数据类型——表示有限范围的整数。这些类型如图 2-9 和图 2-10 所示，其中还给出了“典型”32 位和 64 位机器的取值范围。每种类型都能用关键字来指定大小，这些关键字包括 `char`、`short`、`long`，同时还可以指示被表示的数字是非负数（声明为 `unsigned`），或者可能是负数（默认）。如图 2-3 所示，为这些不同的大小分配的字节数根据程序编译为 32 位还是 64 位而有所不同。根据字节分配，不同的大小所能表示的值的范围是不同的。这里给出来的唯一一个与机器相关的取值范围是大小指示符 `long` 的。大多数 64 位机器使用 8 个字节的表示，比 32 位机器上使用的 4 个字节的表示的取值范围大很多。

C数据类型	最小值	最大值
<code>[signed]char</code>	-128	127
<code>unsigned char</code>	0	255
<code>short</code>	-32 768	32 767
<code>unsigned short</code>	0	65 535
<code>int</code>	-2 147 483 648	2 147 483 647
<code>unsigned</code>	0	4 294 967 295
<code>long</code>	-2 147 483 648	2 147 483 647
<code>unsigned long</code>	0	4 294 967 295
<code>int32_t</code>	-2 147 483 648	2 147 483 647
<code>uint32_t</code>	0	4 294 967 295
<code>int64_t</code>	-9 223 372 036 854 775 808	9 223 372 036 854 775 807
<code>uint64_t</code>	0	18 446 744 073 709 551 615

图 2-9 32 位程序上 C 语言整型数据类型的典型取值范围