6.4 Data Types

Table 6-1 lists the size, representation, and range of each scalar data type for the C28x compiler. Many of the range values are available as standard macros in the header file limits.h.

The storage and alignment of data types is described in Section 7.1.7.

Table 6-1. TMS320C28x C/C++ COFF and EABI Data Types

			Range	
Type	**Size**	**Representation**	**Minimum**	**Maximum**
char, signed char	16 bits	ASCII	-32 768	32 767
unsigned char	16 bits	ASCII	0	65 535
_Bool	16 bits	Binary	0 (false)	1 (true)
short	16 bits	Binary	-32 768	32 767
unsigned short	16 bits	Binary	0	65 535
int, signed int	16 bits	Binary	-32 768	32 767
unsigned int	16 bits	Binary	0	65 535
long, signed long	32 bits	Binary	-2 147 483 648	2 147 483 647
unsigned long	32 bits	Binary	0	4 294 967 295
long long, signed long long	64 bits	Binary	-9 223 372 036 854 775 808	9 223 372 036 854 775 807
unsigned long long	64 bits	Binary	0	18 446 744 073 709 551 615
enum (1)	16 bits	Binary	-32 768	32 767
float (2)	32 bits	IEEE 32-bit	1.19 209 290e-38(3)	3.40 282 35e+38
double (COFF)	32 bits	IEEE 32-bit	1.19 209 290e-38(3)	3.40 282 35e+38 (COFF)
double (EABI)	64 bits	IEEE 64-bit	2.22 507 385e-308(3)	1.79 769 313e+308
long double	64 bits	IEEE 64-bit	2.22 507 385e-308(3)	1.79 769 313e+308
pointers (4)	32 bits	Binary	0	0xFFFFFFFF

(1) For details about the size of an enum type, see Section 6.4.1.
(2) It is recommended that 32-bit floating point values for COFF be declared as `float`, not as `double`.
(3) Figures are minimum precision.
(4) Even though pointers are 32-bits, the compiler assumes that the addresses of global variables and functions are within 22-bits.

The wchar_t type is implemented as int (16 bits) for COFF and long (32 bits) for EABI.

Negative values for signed types are represented using two's complement.

NOTE: TMS320C28x Byte is 16 Bits

By ANSI/ISO C definition, the sizeof operator yields the number of bytes required to store an object. ANSI/ISO further stipulates that when sizeof is applied to char, the result is 1. Since the TMS320C28x char is 16 bits (to make it separately addressable), a byte is also 16 bits. This yields results you may not expect; for example, size of (int) = = 1 (not 2). TMS320C28x bytes and words are equivalent (16 bits). To access data in increments of 8 bits, use the __byte() and __mov_byte() intrinsics described in Section 7.6.

6.4.1 Size of Enum Types

An enum type is represented by an underlying integer type. The size of the integer type and whether it is signed is based on the range of values of the enumerated constants.

In strict C89/C99/C11 mode, the compiler allows only enumeration constants with values that will fit in "int" or "unsigned int".