



Neuron C Data Element Storage General Information

Neuron_C_storage.doc

Ver. 2.0, 2000-10-30, by Greg Neff

Note: this file may be freely distributed.

Introduction

All C compilers have implementation dependencies, and different processors manage data in different ways. The following information is extracted from the Echelon® Neuron® C Reference Guide, and from empirical measurements using a logic analyzer:

Data type limits:

Neuron C data type sizes are atypical:

Type	Bits	Format	Minimum Value	Maximum Value
boolean	1	Binary	0	1
char	8	2's complement	-128	127
short	8	2's complement	-128	127
int	8	2's complement	-128	127
unsigned char	8	Binary	0	255
unsigned short	8	Binary	0	255
unsigned int	8	Binary	0	255
long	16	2's complement	-32,768	32,767
unsigned long	16	Binary	0	65,535
s32_type	32	Structure	-2,147,483,647	+2,147,483,647
float_type	32	Structure	-1E38	1E38

Data storage order:

The Neuron Chips use big-endian ordering. Specifically, when a Neuron Chip writes or reads 16-bit values to or from memory, it first accesses the most significant byte at address n , and then accesses the least significant byte at address $n+1$.

Structures and Bit Fields:

All fields declared in a structure are stored most significant byte first and then most significant bit first.

```
typedef struct {
    unsigned bit0    :1;    // most significant bit in most significant byte
    unsigned bit1    :1;
    unsigned bit2    :1;
    unsigned bit3    :1;
    unsigned bit4    :1;
    unsigned bit5    :1;
    unsigned bit6    :1;
    unsigned bit7    :1;    // least significant bit in most significant byte
    unsigned bit8    :1;    // most significant bit in least significant byte
    unsigned bit9    :1;
    unsigned bit10   :1;
    unsigned bit11   :1;
    unsigned bit12   :1;
    unsigned bit13   :1;
    unsigned bit14   :1;
    unsigned bit15   :1;    // least significant bit in least significant byte
} example_struct;
```

Note that bit fields are commonly named starting with bit0 at the top, but this is a label only. This does not correlate to the bit position in the byte. With respect to signals on the Neuron 3150® Chip, D0 is the least significant bit, and A0 is the least significant address line.

Unions:

```
union {
    example_struct eu_struct;    // example_struct defined above
    unsigned eu_array[2];    // index 0 = msb, index 1 = lsb
    struct {
        unsigned msb;    // most significant byte first
        unsigned lsb;    // least significant byte last
    } union_struct;
} example_union;
```

In this example_union:

most significant byte of example_union.eu_struct = example_union.eu_array[0] = example_union.union_struct.msb
least significant byte of example_union.eu_struct = example_union.eu_array[1] = example_union.union_struct.lsb

Echelon, Neuron and 3150 are trademarks of Echelon Corporation registered in the United States and other countries. Other marks belong to their respective holders.