**Representing Negative Numbers**

Always think of negative numbers as – (#); for example, -450 as – (450).

We will look at three ways of representing negative numbers in computers, Sign Magnitude, One’s Complement, and Two’s complement.

**Sign Magnitude:**

Change the first bit to 1.

For example: 0001 0001 00002 becomes

1001 0001 00002

**One’s Complement:**

In the binary form, exchange, or invert, the 0’s for 1’s and the 1’s for 0’s.

For example: 0001 1001 00012 becomes

1110 0110 11102

To convert to hexadecimal:

1110 0110 11102

E 6 E16

To convert to decimal:

1. Convert to positive binary.
2. Convert to decimal and make the number negative.

**Two’s Complement notation of integers**

Two's complement is a common way computers choose to represent integers. To get the two's complement negative notation of an integer, you write out the number in binary. You then invert the digits, and add one to the result.

For example: -28 = 1110 0100

1. 0001 1100 becomes 1110 0011
2. 1110 0011

+ 1

1110 0100

A leading 1 means the number is negative, a leading 0 means the number is 0 or positive in two’s complement.

**Conversion from Two's Complement**

1. Reverse the sign of this number by simply inverting the bits (0 goes to 1, and 1 to 0).
2. Add one to the resulting number.
3. Make the number negative.

For example:

1111 1111 1111 1111 1111 1111 1111 1111 = -1

1. The inversion of that binary number:

0000 0000 0000 0000 0000 0000 0000 0000

1. Then add one.

0000 0000 0000 0000 0000 0000 0000 0001