Binary Addition and Negative Numbers

Author: Jared Moore

Edited by Nathan Bowman

$$\begin{array}{r} 10_{10} \\ +\,12_{10} \\ \hline \end{array} \qquad \begin{array}{r} 5_{10} \\ +\,1_{10} \\ \hline \end{array} \qquad \begin{array}{r} 0110_{2} \\ +\,1001_{2} \\ \hline \end{array} \qquad \begin{array}{r} 0100_{2} \\ +\,0001_{2} \\ \hline \end{array}$$

$$\begin{array}{r} 19_{10} \\ +\,12_{10} \\ \hline \end{array} \qquad \begin{array}{r} 7_{10} \\ +\,6_{10} \\ \hline \end{array} \qquad \begin{array}{r} 0110_{2} \\ +\,1011_{2} \\ \hline \end{array} \qquad \begin{array}{r} 1100_{2} \\ +\,1001_{2} \\ \hline \end{array}$$

Negative Numbers

Binary numbers we've talked about are unsigned.

(All positive.)

How could we represent negative numbers?

Negative Numbers

How about: -1010_2?

That works with mathematical binary numbers, but we are interested in numbers stored on a machine

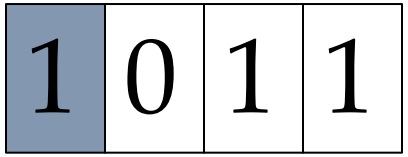
First system is doing essentially this, but in a way we can store in 1s and 0s

Sign/Magnitude

Most significant bit represents sign

- Similar to decimal in that the sign has a fixed position.

0 is positive, 1 is negative



Are there any issues with this system?

Sign/Magnitude

0000_2

1000_2

Sign/Magnitude

$$\begin{array}{r} 0100_2 \\ +\ 0001_2 \\ \hline \end{array} \qquad \begin{array}{r} 1100_2 \\ +\ 1001_2 \\ \hline \end{array} \qquad \begin{array}{r} 0100_2 \\ +\ 1001_2 \\ \hline \end{array}$$

Sign/Magnitude

$$\begin{array}{r} 0100_2 \\ +\ 0001_2 \\ \hline 0101_2 \end{array}$$

$$\begin{array}{r} 1100_2 \\ +\ 1001_2 \\ \hline 0101_2 \end{array}$$

$$\begin{array}{r} 0100_2 \\ +\ 1001_2 \\ \hline 1101_2 \end{array}$$

Should be

5

-5

3

Fixed number of bits

Made implicit assumption that we work with *fixed number of bits*

- Very reasonable assumption when working with computers

Otherwise, what would it mean for "first bit" to be 1?

In standard, non-machine-based binary numbers

$$1000_2 = 01000_2$$

In sign/magnitude representation, those are different numbers

In fact, doesn't really make sense to compare them

Fixed number of bits (continued)

With negative binary number systems, numbers will be represented differently depending on number of bits

-2 decimal = 110_2 (in three bits)

0110_2 (in four bits) = 6 decimal

Takeaway: you need to know how many bits you are working with to use negative numbers in binary

This will be true of both representations we discuss