

Example: Converting to Float

Convert the following single-precision IEEE 754 number into a floating-point decimal value.

1 10000001 10110011001100110011010

1. **First, put the bits in three groups.**

Bit **31** (the leftmost bit) shows the **sign** of the number.

Bits **23-30** (the next 8 bits) are the **exponent**.

Bits **0-22** (on the right) give the **fraction**

2. **Now, look at the sign bit.**

If this bit is a 1, the number is negative.

If it is 0, the number is positive.

This bit is **1**, so the number is negative.

3. **Get the exponent and the correct bias.**

The exponent is simply a positive binary number.

$10000001_{\text{bin}} = 129_{\text{ten}}$

Remember that we will have to subtract a bias from this exponent to find the power of 2. Since this is a single-precision number, the bias is 127.

4. **Convert the fraction string into base ten.**

This is the trickiest step. The binary string represents a fraction, so conversion is a little different.

Binary fractions look like this:

$$0.1 = (1/2) = 2^{-1}$$

$$0.01 = (1/4) = 2^{-2}$$

$$0.001 = (1/8) = 2^{-3}$$

So, for this example, we multiply each digit by the corresponding power of 2:

$$0.10110011001100110011010_{\text{bin}} = 1 \cdot 2^{-1} + 0 \cdot 2^{-2} + 1 \cdot 2^{-3} + 1 \cdot 2^{-4} + 0 \cdot 2^{-5} + 0 \cdot 2^{-6} + \dots$$

$$0.10110011001100110011010_{\text{bin}} = 1/2 + 1/8 + 1/16 + \dots$$

Note that this number is just an approximation on some decimal number. There will most likely be some error. In this case, the fraction is about

0.7000000476837158.

5. This is all the information we need. We can put these numbers in the expression:

$$(-1)^{\text{sign bit}} * (1 + \text{fraction}) * 2^{\text{exponent} - \text{bias}}$$

$$= (-1)^1 * (1.7000000476837158) * 2^{129-127}$$

$$= -6.8$$

The answer is approximately -6.8.

[Return to index ->](#)
