Binary, Decimal, and Hexadecimal

Binary, Decimal, and Hexadecimal

Overview

Binary, Decimal, and Hexadecimal are all different means of representing numbers. The different types are used depending on the level of abstraction when it comes to computational hardware and each type has its place.

Decimal

Decimal is the "standard" nomenclature of representing a number. Officially, it is called base 10 and typically will be abreviated with a subscript 10 at the end of the number when presented alongside binary or hexadecimal values, however some sources choose to ignore the subscript for base 10 only. The reason it is called base 10 is because each digit has 10 distinct values (0-9).

Examples

8_{10} 124_{10} 2048_{10}

Binary

Binary is a way of representing numbers in base 2 (two), where each digit can have two values (0, 1). Binary is typically used close to hardware where a 0 can be representing by one voltage and a 1 can be represented by another voltage, however different encoding schemes exist where 0s and 1s could be decoded from the phase of a signal, frequency of a signal, voltage transfer over a single bit period of a signal, etc. Regardless, once it is converted to binary it can represent data and numbers in a logical form. When written, it is convention that the right-most digit (or bit) represents the least significant value (typically 0/1). Sometimes underscores are appened in between groups of 4 or 8 bits for readability (similar to commas in decimal notation). Each bit represents 2 to the power of the position of that bit from right to left, starting with 0. A collection of 8 bits is called a byte.

Examples

Bits value table

Bit Position	8	7	6	5	4	3	2	1	0
Value	2^8^	2^7^	2^6^	2^5^	2^4^	2^3^	2^2^	2^1^	2^0^
Decimal	256	128	64	32	16	8	4	2	1

Binary and Decimal numbers

$1000_2 = 8_{10}$ $0111\_1100_2 = 124_{10}$ $1000\_0000\_0000_2 = 2048_{10}$

Converting from Decimal

To go to binary from decimal, find the largest power of 2 which can fit into the signal, add a 1 in the position of that value, then subtract that value from the decimal number. Continuously repeat this while adding 0s to the signal to pad values which cannot fit into the remainder.

Example

Step	Current Decimal Remainder	Step	Current Binary Value
0	86	-64	1000000
1	22	-16	1010000
2	6	-4	1010100
3	2	-2	1010110
3	0	-0	1010110

Converting to Decimal

To convert from binary to decimal, multiply each value with a 1 in it with it's position's weight and sum the weights

Bit Position	6	5	4	3	2	1	0
Value	1	0	1	0	1	1	0
Weight	64	0	16	0	4	2	0
Sum: 64 + 16 + 4 + 2 = 86							

Hexadecimal

Hexadecimal (often abbreviated as Hex) is effectively only used as a human-readable format of binary. It is base-16 which allows for four digits in binary to be represented as 1 digit in hexadecimal. It is often used when representing long binary numbers in a display for a human to read. Each byte can be represented as 2 digits in Hex. Hex numbers can either be noted by having the subscript 16 after, or by being prefaced with "0x". Since decimal only offers digits 0-9, hexadecimal represents the remaining 6 digits with letters A-F. A table is attached below to show the correlation.

Hex Value	Decimal Value	Binary Value
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011

Hex Value	Decimal Value	Binary Value
C	12	1100
D	13	1101
E	14	1110
F	15	1111

Examples

Decimal Value	Binary Value	Hex Value
56	0011_1000	0x38
12	0000_1100	0x0C
15	0000_1111	0x0F

Negative Numbers

Overview

Negative numbers are represented in binary in either two different ways: a sign bit, or 2's complement.

Sign Bit

A sign bit is when a specific bit of the value is assigned to represent the sign, often with a 0 representing positive and 1 representing negative. Typically the sign bit is the most significant bit (MSB). This was one of the first ways to represent negative numbers in binary, and has flaws as it makes computation such as addition and subtraction harder to do in hardware, and implements two different 0 values, a positive and negative (+0 / -0).

Examples

Decimal Value	Binary Value

0	0000_0000 OR 1000_0000
12	0000_1100
-12	1000_1100

2's Complement

Two's complement is the more unanimous method for representing binary numbers. In order to represent a negative number in binary, take the positive number in binary, flip all of the bits, and add 1. Reversing the process to convert a negative to a positive is just a matter of doing the same process of flipping all the bits and adding 1. To check if a number is positive or negative, the most significant bit can be checked for being a 1 (negative) or 0 (positive). This format is popular because it maintains having only one 0 value, and allows direct computation using the same circutry as normal, positive binary numbers.

Examples

Deicmal Value	Binary Value
0	0000_0000
12	0000_1100
-12	1111_0100

Revision #2
Created 2026-07-30 20:24:54 UTC by iantech
Updated 2026-07-30 20:25:30 UTC by iantech