NUMBER BASE SYSTEM

A computer or any digital system works in a binary manner. The main number systems used in digital hardware are as follows.

DECIMAL NUMBER SYSTEM: The decimal number system (base 10) number system has ten as it base. It uses various symbols called digit for ten distinct value (0,1,3,4,5,6,7,8 and 9) to represent numbers. It requires 10 different types of electronic pulse.

The decimal system is a position number system. It has position for unit, tens, hundred e.t.c The position of each digit conveys the multiplier (a power of ten) to be used with the digit- each position has value to ten time of a position to its right. For example:

275 = (2×100) + (7×10) + (5×1)

2x 102 + 7 + 101 + 5 x 100

BINARY NUMBER

The binary number (base 2) number system represents values using symbols typically 0 and 1. In other words, the binary number system is a position number system with a power of two (2). Owing to its relatively straightforward implementation in electronic circuitry, the binary is used internally by virtually in all modern computers.

The numerals 0 and 1 have the same meaning in the decimal system, but a different interpretation is placed on the position occupied by a digit.

In the binary number system, the individual digit represent the coefficient of power 2 rather than 10 as in the decimal number system. For example, the decimal number system 19 is written in the binary representation as 10011

14030211102 = 1 x 24 +0 x 23 + 0 x 22 + 1 x 21 + 1 x 20

= 16 + 0 + 0 + 2 + 1.

Let’s look at base-two, or binary, numbers. How would you write, for instance, 1210 (“twelve, base ten”) as a binary number? You would have to convert to base-two columns, the analogue of base-ten columns. In base ten, you have columns or “places” for 100 = 1, 101 = 10, 102 = 100, 103 = 1000, and so forth. Similarly in base two, you have columns or “places” for 20 = 1, 21 = 2, 22 = 4, 23 = 8, 24 = 16, and so forth.

The first column in base-two math is the units column. But only “0” or “1” can go in the units column. When you get to “two”, you find that there is no single solitary digit that stands for “two” in base-two math. Instead, you put a “1” in the twos column and a “0” in the units column, indicating “1 two and 0 ones”. The base-ten “two” (210) is written in binary as 102.

A “three” in base two is actually “1 two and 1 one”, so it is written as 112. “Four” is actually two-times-two, so we zero out the twos column and the units column, and put a “1” in the fours column; 410 is written in binary form as 1002. Here is a listing of the first few numbers:

decimal (base 10)	binary (base 2)

0	0	0 ones
1	1	1 one
2	10	1 two and zero ones
3	11	1 two and 1 one
4	100	1 four, 0 twos, and 0 ones
5	101	1 four, 0 twos, and 1 one
6	110	1 four, 1 two, and 0 ones
7	111	1 four, 1 two, and 1 one
8	1000	1 eight, 0 fours, 0 twos, and 0 ones
9	1001	1 eight, 0 fours, 0 twos, and 1 one
10	1010	1 eight, 0 fours, 1 two, and 0 ones
11	1011	1 eight, 0 fours, 1 two, and 1 one
12	1100	1 eight, 1 four, 0 twos, and 0 ones
13	1101	1 eight, 1 four, 0 twos, and 1 one
14	1110	1 eight, 1 four, 1 two, and 0 ones
15	1111	1 eight, 1 four, 1 two, and 1 one
16	10000	1 sixteen, 0 eights, 0 fours, 0 twos, and 0 ones

Converting between binary and decimal numbers is fairly simple, as long as you remember that each digit in the binary number represents a power of two.

Convert 1011001012 to the corresponding base-ten number.

I will list the digits in order, and count them off from the RIGHT, starting with zero:

digits:	1 0 1 1 0 0 1 0 1
numbering:	8 7 6 5 4 3 2 1 0

The first row above (labelled “digits”) contains the digits from the binary number; the second row (labelled ” numbering”) contains the power of 2 (the base) corresponding to each digits. I will use this listing to convert each digit to the power of two that it represents:

1×28 + 0×27 + 1×26 + 1×25 + 0×24 + 0×23 + 1×22 + 0×21 + 1×20

= 1×256 + 0×128 + 1×64 + 1×32 + 0×16 + 0×8 + 1×4 + 0×2 + 1×1

= 256 + 64 + 32 + 4 + 1

= 357

DECIMAL	BINARY
9	1001
10	1010
11	1011
12	1100
13	1101

14	1110
15	1111
100	0100100
512	1000000000

= 19

DECIMAL	BINARY
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000

OCTAL NUMBER SYSTEM

The octal number system is a base 8 number system, and uses the digits from 0 to 7. Programs often display in an octal format because it can be translate relatively in binary format, each digit in the octal number system represents a power of base 8. For example the binary representation for decimal 74 is 1001010, which group into 1001010, so the octal representation is 112

1128 = 1 x 82 + 1 x 81 + 2 x 80

= (1 x 64) + (1 x 8) + (2 x 1)

= 64 + 8 + 2

= 74.

So, the decimal equivalent of octal number 1128 is 7410. Since there are only 8 digit (0-8) in the octal number system, 3 bits are sufficient to represent an octal number in a binary digits.

OCTAL	BINARY
0	000
1	001

2	101
3	011
4	100
5	101
6	110
7	111

With this table, it is easy to translate octal and binary system for example

658 = 110 1012

178 = 001 1112

HEXADECIMAL NUMBER SYSTEM

In the hexadecimal number system is a number with a base of 16, usually written using symbols 0-9 and A-F. for example, the decimal number 79 whose binary representation is 01001111 can be written as 4F in hexadecimal (4 = 0100, F = 1111) for example 1FF16 = 1 x 132 + F x 161 + F x 160

= 1 x 256 + 15 x 16 + 16 x 1

= 511.

Thus, the decimal equivalent of hexadecimal number 1FF16 is 51110. Since there are only 16 digits in the hexadecimal number system, 4 bits are sufficient to represent any hexadecimal number in binary.

The current decimal number system was first introduced to the computing world in 1963 by international business machine (IBM). An early version that used the digit 0-9 and u-2 was introduced in 1956, in the Bendix G-15 computer

The tale given below displays the binary and decimal equivalent of some hexadecimal numbers

HEXADECIMAL	BINARY	DECIMAL
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10

B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15