Numbering systems

Positional notation:

a numeral system where the value represented by a digit depends on its value, multiplied by the base to the power of the digit's position, and the number is equal to the sum of the products of each digit. The first digit is in position 0, the second digit is in position 1, and so on.

For example: 986456 in base 10 =

$9 \times 10^5 + 8 \times 10^4 + 6 \times 10^3 + 4 \times 10^2 + 5 \times 10^1 + 6 \times 10^0$

900000 + 80000 + 6000 + 400 + 50 + 6

= 986456

Binary, Octal and hexadecimal numbering systems:

Differences:

Binary has base 2 and has two digits (0 and 1)

Octal has base 8 and has eight digits (0, 1, 2, 3, 4, 5, 6, 7)

Hexadecimal has base 16 and has sixteen digits (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F)

4-digit decimal to binary example:

Decimal to binary conversion can be done by dividing the given decimal number recursively by 2. Then, the remainders are noted down until 0 is the final quotient. Then, these remainders are written in reverse order to get the binary value of the decimal number, with the first remainder being the last digit(the least valuable digit), and the last remainder being the first digit(the most valuable digit).

For example: 2468 in base 10 –

2468 / 2 = 1234 rem <u>0</u>

1234 / 2 = 617 rem <u>0</u>

617 / 2 = 308 rem <u>1</u>

308 / 2 = 154 rem <u>0</u>

154 / 2 = 77 rem <u>0</u>

77 / 2 = 38 rem <u>1</u>

38 / 2 = 19 rem <u>0</u>

19 / 2 = 9 rem <u>1</u>

9 / 2 = 4 rem <u>1</u>

4 / 2 = 2 rem <u>0</u>

2 / 2 = 1 rem <u>0</u>

1 / 2 = 0 rem <u>1</u>

=100110100100 in base 2

10-digit binary to decimal example

Binary to decimal conversion can be done by using the positional notation method. This is where the value of a digit in a number is based on its position in the number. This is achieved by multiplying each digit times the base, 2, to the power of depending upon the position of the digit in the number. The sum of each value obtained for each digit gives the equivalent value of the given binary number in the decimal system.

For example: 1000111100 in base 2-

1(9), 0(8), 0(7), 0(6), 1(5), 1(4), 1(3), 1(2), 0(1), 0(0).

~ position of each digit in the number

$1.2^9 + 0.2^8 + 0.2^7 + 0.2^6 + 1.2^5 + 1.2^4 +1.2^3 + 1.2^2 + 0.2^1 + 0.2^0 =$

512 + 0 + 0 + 0 + 32 + 16 + 8 + 4 + 0 + 0

= 572 in base 10

4-digit decimal to octal example

There are two methods to convert decimal to octal. The first method is to convert the number from decimal to another numbering system such as binary or hexadecimal, and then convert again to octal. The second method involves converting the number directly from decimal to octal.in this example, I will use the second example, and convert the number directly from decimal to octal. This is done by dividing the number by 8 continuously until the number is less then 8, whilst keeping records of the remainders and organising them backwords to get your new number in octal, with the first remainder being the last digit (least valuable digit), and the last remainder being the first digit (most valuable digit).

For example, 1184 in base 10=

1184/8= 148 rem <u>0</u>

148/8= 18 rem <u>4</u>

18/8= 2 rem <u>2</u>
2/8= 0 rem <u>2</u>
=2240 base 8

4-digit octal to decimal example

Conversion of a number from octal to decimal is done similarly as any other numbering system to decimal. One must take each individual digit of the number, and multiply each digit by its base (8) to the power of the position of the digit in the number, and then all the products must be added together for the final number in decimal. For example, 654 base 8 in base 10 would be
$6 \times 8^2 + 5 \times 8^1 + 4 \times 8^0$.
For example – 4368 in base 8=
$4 \times 8^3 + 3 \times 8^2 + 6 \times 8^1 + 8 \times 8^0$
4 x 512 + 3 x 64 + 6x8 + 8x1
2048 + 192 + 48 + 8
= 2296 base 10

4-digit decimal to hexadecimal example

To convert a decimal number to hexadecimal, you must divide the decimal number by 16 and then divide the quotient by 16 again until the quotient is equal to zero. Keep note of all the remainders while the division takes place. Then, the reverse order pattern must be followed to arrange the values of the remainders into the hexadecimal number, Bearing in mind that A, B, C, D, E, and F all represent 10, 11, 12, 13, 14 and 15.
For example, 3674 base 10 =
3674/16 = 229 rem <u>10</u>
229/16 = 14 rem <u>5</u>
14/16 = 0 rem <u>14</u>
10=A, 5=5, 14=E

=E 5 A

4-digit hexadecimal to decimal example

To convert a hexadecimal number to a decimal number , firstly one must convert any letters in the hexadecimal numbers to their numbers equivalent. A, B, C, D, E and F all correspond to 10, 11, 12, 13, 14 and 15 in hexadecimal. Once that step is complete, we then multiply each individual digit by the base of the number (16) to the power of the digit's position in the number. Once all the individual values are found , we add all the results together to get the full decimal number.

For example, 7 B 5 A in base 16 =

7, 11, 5, 10

$7 \times 16^3 + 11 \times 16^2 + 5 \times 16^1 + 10 \times 16^0 =$

7 x 4096 + 11 x 256 + 5 x 16 + 10 x 1

28672 + 2816 + 80 + 10

= 31578 base 10

8-digit binary to octal example

To convert a binary number to octal, we must first divide the binary number into groups of 3 starting from the right hand side of the number (least significant digit). We can use this chart to identify each 3 digit binary group and its corresponding octal number, (note- there are only 8 digits in octal, 0-7, therefore there are no values over 7 in octal).

binary	octal
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

If there are any numbers on the left (most significant digit) not in a group of 3, add in 0(s) until the pair of 3 is complete. Then the remaining digits are the octal number.

For example- 10011011

010 011 011

2 3 3
= 233 in base 8

8-digit octal to binary example

To convert an octal number to binary we must first take the octal number, and then convert each induvial digit into its 3 bit binary counterpart, as seen in the binary to octal example above. Once every octal digit is replaced with the 3 bit binary group all the groups are put together to get the full binary number.

For example- 23445443=

010 011 100 100 101 100 100 011

= 010011100100101100100011

Addition of two binary numbers works using two 8-bit binary numbers

While adding two binary numbers together, there are certain rules that must be followed. As the only digit values possible in binary are 0 and 1, 1 + 1 cannot =2. Therefore, addition is done normally with 0 + 0, or either variation of 0 + 1, but if 1 + 1 occurs, the result must be 0, whilst the 1 is carried over to the left. If 1 + 1 + 1 occurs, the result must be 1, with an additional 1 carrying over to the left.

For example- 10110011 + 01011101 in base 2

```
011111111
  10110011
+01011101
 0100010000
```

Subtraction of two binary numbers works using two 8-bit binary numbers

Subtraction of two binary numbers is also done normally, from right to left, until 0-1 is encountered. In that case , one power of the base (2) must be borrowed from the next left digit. If the next left digit is 0, keep shifting to the left until a 1 is found.

For example- 10110110 – 01101011

02 02022
10110110
\- 01101011
<u>01001011</u>