Chapter One

1.1 Introduction:

Numbers are used to express quantities. There are many numerations systems used in the field of digital electronics, one of the most important being the binary system of numeration on which is based the computer science. Each of the various numerations systems and codes has its advantages but also inconvenient. The aim of this chapter is to present and explain the most common numeration systems and codes used in the conception of digital circuits.

1.2 Digital versus Analogue representation:

There are two basic ways we can represent quantities: Analogue representation and digital representation. With analogue representation, the quantity is symbolised in a way that is infinitely divisible. With digital representation, the quantity is symbolised in a way that is discretely packaged.

Example 1.1:

- The height of the red column which indicates the temperature measured by a thermometer is an analogue representation.
- An electronic watch whose digits changes second after second, minute after minute, shows a digital representation.

The comparison between digital and analogue representations can be given as in the following chart:

Analogue representation	Digital representation
Infinitely divisible	Discrete (Step by step)
Prone to errors of precision	Absolute precision

1.3 Systems of numeration:

To represent quantities in the different systems of numeration, specific symbols are used, which are also called **ciphers**.

1.3.1 Decimal numeration system:

Decimal system is the most common numeration system for daily uses. It is constituted by 10 symbols or ciphers: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9. Each cipher represents an integer quantity and each place from right to left in a decimal notation represents a weight for each integer quantity.

Example 1.2:

Let's consider the decimal notation 1253. This number can be broken into its constituent weight-products as such:

$$1253 = 1000 + 200 + 50 + 3$$

$$1253 = 1\times1000 + 2\times100 + 5\times10 + 3\times1$$

$$1253 = 1\times10^3 + 2\times10^2 + 5\times10^1 + 3\times10^0$$

We can easily notice that the cipher 1 is more weighted than the cipher 2 which in his turn is more weighted than the cipher 5. The cipher 3 is the less weighted. In the decimal numeration system, each cipher is called a **digit**. Each weight or place value is ten that of the one to the immediate right. The less weighted cipher carries the One place, the cipher at the immediate left carries the Tens place, the follower carries the Hundreds place, thousands place, and so on…

1.3.2 Binary numeration system:

The binary numeration system uses only two ciphers instead of ten as the decimal numeration system. Those two ciphers are "0" and "1". In binary system of numeration, ciphers are called bit (Binary Digit). Cipher are arranged right to left in doubling values of weight (instead of multiplying the weight by 10 as in the case of decimal system).

Example 1.3:

Let's consider the following binary number

Weights

5 4 3 2 1 0

$A = 1\ 0\ 1\ 1\ 0\ 1_2$

Base 2

$$A = 1\times 2^5 + 0\times 2^4 + 1\times 2^3 + 1\times 2^2 + 0\times 2^1 + 1\times 2^0$$
$$A = 32 + 0 + 8 + 4 + 1$$
$$A = 45_{10}$$

Each weight is 2 that of the one in the immediate right. The less weighted cipher carries the Ones place (2^0), the cipher at the immediate left carries the twos place (2^1), the following cipher carries the fourth place (2^2)…

Exercise 1.1:

Convert the following binary numbers to decimal numbers:

A = 110101 B = 100110101 C = 11110111101 D = 101100001111

1.3.3 Binary versus decimal numeration system:

Let us count from 0 to 15 using binary and decimal systems of numeration

Binary				Decimal
D(MSB)	C	B	A(LSB)	
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

It is obvious that the representation of a quantity in binary numeration system takes mores ciphers than in decimal system. We can therefore ask ourselves why the binary system is preferred to decimal system in computer sciences. The reason is that in electronics, it is easier to materialise two quantities “0” and “1” (by two different voltages for example) than to materialises 10 different quantities “0” ,”1” ,”2”,”3”,”4”,”5”,”6”,”7”,”8”, and ”9” (by 10 different voltages). In fact, in digital circuits, 0 and 1 are materialised by specific ranges of voltages or current; this will be discussed later.

Remark 1.1:

With n bits we can represent 2^n different binary numbers. The higher H number is given using the following formula.

$$H = 2^n - 1 \qquad (1)$$

Example 1.4:

With 4 bits we can represent $2^4 = 16$ different binary numbers (from 0 to 15), and the higher number is $H = 2^4 - 1 = 15$.

Remark 1.2: Conversion from binary to decimal

To convert a number written in binary numeration system to its equivalent in decimal, we just have to calculate the products of the bits with their respective weights, as in example 1.3 above. For binary numbers with "binary point" (equivalent of decimal point for decimal numbers), the conversion is done as follow.

$$\begin{array}{lcccccc} & 2 & 1 & 0 & -1 & -2 & -3 \\ A= & 1 & 0 & 1. & 1 & 0 & 1 \end{array}$$

$A= 1\times2^{2} + 0\times2^{1}+1\times2^{0} +1\times2^{-1}+0\times2^{-2}+1\times2^{-3}$

$A = 5.625_{10}$

Exercise 1.2:

Convert from binary to decimal:

A = 10110.01   B = 11110111.1011   C = 111.111   D = 10110101101.111101

1.3.4 Octal numeration system:

The octal numeration system is a place weighted system with a base of eight. Valid ciphers include the symbols "0","1","2","3","4","5","6", and "7".

To convert from binary to octal numeration system, we just have to divide the number into groups of binary numbers having 3 bits each. And each group of 3 bits is replaced by its equivalent in octal.

Example 1.5:

Let's convert the following binary numbers in octal:

A = 10110101

B = 11010111.01

$A = \underbrace{010}_{2}\,\underbrace{110}_{6}\,\underbrace{101}_{5_8}$

$\boxed{10110101_2 = 265_8}$

The bits are grouped from the right to the left. A zero has been added to the two firs bits to form a group of 3 bits. That zero is called an implied zero.

$B = \underbrace{011}_{3}\,\underbrace{010}_{2}\,\underbrace{111}_{7}\,.\,\underbrace{010}_{2_8}$

$\boxed{11010111.01_2 = 327.2_8}$

Two implied zeros have been added to the number to form groups of 3 bits.

1.3.5 Hexadecimal numeration system:

The hexadecimal numeration system is a place weighted system with a base of sixteen. Valid ciphers include the normal decimal symbols “0”,”1”,”2”,”3”,”4”,”5”,”6”,”7”;”8”;9” plus six alphabetical characters A, B, C, D, E, and F. The following table summarizes the equivalence between decimal, binary, octal and hexadecimal systems.

Decimal	Binary	Octal	Hexadecimal
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

To convert from binary to hexadecimal numeration, we group bits in fours. Each group of four bit is replaced by its hexadecimal equivalent.

Example 1.6:

Convert the following binary numbers in hexadecimal.

A = 1101011101

B = 11101011101.11

As explained above, we just have to group the binary number in groups of four bits each:

A = **00**11 $\underbrace{0101}$ 1101

3 5 D_{16}

$\boxed{1101011101_2 = 35D_{16}}$

The binary number has been grouped is groups of four bits each, from the right to the left two implied zeros have been added at the extreme left. In the same way the number B can also be converted.

$B = \underbrace{0111}_{7}\ \underbrace{0101}_{5}\ \underbrace{1101}_{D}\ .\underbrace{1100}_{C}\,{}_{16}$

$\boxed{11101011101.11_2 = 75DC_{16}}$

1.4 Changing of base:

We have already seen in the previous section how to change from binary to decimal, octal or hexadecimal systems of numeration. The present section is intended to show how to move from a given system of numeration to any other system.

1.4.1 From octal and hexadecimal to binary and decimal:

The octal and hexadecimal systems are actually used by computer engineer just to obtain a "shorthand" representation of binary numbers (because octal and hexadecimal representations take a few numbers of ciphers or symbols as compared to binary system). It should therefore be understood that only binary system is implemented in the electronic circuits of digital systems (through two levels of voltages or currents: high (1) and low (0)), the others systems being used by engineers just for simplification issues.

However, we sometimes have the need to convert either of those systems to binary or decimal forms.

1.4.1.1 Octal and hexadecimal to binary:

It is obvious that, to convert from octal to binary, we just have to convert each octal cipher to its binary equivalent in 3 bits. In the same way, to convert from hexadecimal to binary, we should convert each hexadecimal symbol into its binary equivalent in 4 bits.

Example 1.7:

a) Convert the following octal number to digital 523_8.

b) Convert the following hexadecimal number to binary $4DC2_{16}$.

$523_8 = \underbrace{101}_{5}\ \underbrace{010}_{2}\ \underbrace{011}_{3}\,{}_2$

$\boxed{523_8 = 101010011_2}$

$4DC2_{16} = \underbrace{0100}_{4}\ \underbrace{1101}_{D}\ \underbrace{1100}_{C}\ \underbrace{0010}_{2}\,{}_2$

$\boxed{4DC2_{16} = 100110111000010_2}$

1.4.1.2 Octal to decimal:

Because octal is a base of eight numeration system, each place weight value differs from either adjacent place by factor of eight.

Example 1.8:

Let us convert the following octal number to decimal: A = 264.74_8

2 1 0 -1 -2

A = 2 6 4. 7 4_8

$$A = 2 \times 8^2 + 6 \times 8^1 + 4 \times 8^0 + 7 \times 8^{-1} + 4 \times 8^{-2}$$
$$A = 180.9375_{10}$$

Exercise 1.3:

Convert the following octal number to decimal:

A = 4562.36_8 B = 523411.232_8 C= 264.365_8 D = 451632_8

Is the number 12586 an octal number?

1.4.1.3 Hexadecimal to decimal:

The technique for converting hexadecimal notation to decimal is the same as the one used above, except that each successive place weight changes by a factor of sixteen.

Example 1.9:

Let us convert the following hexadecimal number to decimal: A = $34DF.AC2_{16}$

3 2 1 0 -1-2-3

A = 3 4 D F.A C 2_{16}

$$A = 3 \times 16^3 + 4 \times 16^2 + 13 \times 16^1 + 15 \times 16^0 + 10 \times 16^{-1} + 12 \times 16^{-2} + 2 \times 16^{-3}$$
$$A = 12288 + 1024 + 208 + 15 + 0.625 + 0.046875 + 0.000488281$$
$$A = 13535.67236_{10}$$

Exercise 1.4:

Convert from hexadecimal to decimal.

X = $A23C.DF_{16}$ Y = $7D3E_{16}$ Z= $D96EC.FA_{16}$

1.4.2 Conversion from decimal numeration system to others systems:

The conversion from decimal numeration system to others systems of numeration is an important task for everyone dealing with computer science, because it permits to move from daily world to digital world.

To convert a number from decimal numeration system to binary, octal or hexadecimal, we use repeated cycles of divisions to break the decimal numeration down into multiples of binary, octal or hexadecimal place weight values.

In the first cycle of division, we take the original decimal number and divide it by the base of the numeration system that we are converting to: It meant that for binary, we should divide by 2, for octal we should divide by 8, for hexadecimal we should divide by 16. Then we take the whole number portion of the division result and divide it by the result again, and so on, until we end up with a quotient of less than the base value.

1.4.2.1 Decimal to binary conversion:

Let us convert the decimal number 87_{10} to binary, using the principle described above. It meant that the decimal number should be repeatedly divided by 2.

87	2	
43	2	**1**
21	2	**1**
10	2	**1**
5	2	**0**
2	2	**1**
1		**0**

The coloured ciphers are the reminders of repeated division of the decimal number by 2. To obtain the binary number, we just have to take those reminders, beginning with the last one, as indicated by the arrow. Then we have:

$87_{10} = 1010111_2$

In short, the binary bits are assembled from the reminders of the successive division steps, beginning with the LSB (Least Significant Bit) and proceeding to the MSB (Most significant Bit).

Exercise 1.5:

Convert the following decimal numbers to binary

A = 153_{10} B = 255_{10} C = 46_{10} D = 38_{10}

1.4.2.2 Conversion of decimal numbers less than 1 to binary:

For converting a decimal number less than 1 to binary, we use repeated multiplication by 2, taking the integer portion of the product in each step as the next

digit of our converted number. Let us convert the decimal number 0.375_{10} to binary:

0.375x2 = **0**.75	Integer portion of the product = **0**
0.75x2 = **1**.5	Integer portion of the product = **1**
0.5x2 = **1**	Integer portion of the product = **1** (we stop when the product is a pure integer)

Each step gives us the next bit further away from the binary point, so the binary number is obtained taking the bits from up to down.

$0.375_{10} = 0.011_2$

Remark 1.3:

With integer division, worked from the LSB to the MSB (down to up), but with repeated multiplication, we worked from up to down.

Exercise 1.6:

Convert from decimal to binary:

A = 0.8125_{10} C =0.875_{10} B= 0.625_{10} D = 0.40625_{10}

Remark 1.4:

To convert a decimal number greater than 1 with a less than 1 component, we should use both techniques, one at time. Let us convert the decimal number 23.125_{10} to binary.

Step one: repeated division for the integer portion 23_{10}.

23	2	
11	2	**1**
5	2	**1**
2	2	**1**
1		**0**

Partial answer:

$23_{10} = 10111_2$

Step two: repeated multiplication for the less than 1 portion 0.125_{10}

0.125x2 = 0.25	Integer portion of the product = 0
0.25x2 = 0.5	Integer portion of the product = 0
0.5x2 = 1	Integer portion of the product = 1

Partial answer:

$0.125_{10} = 0.001_2$

Complete answer:

$10111_2 + 0.001_2 = 10111.001_2$

Exercise 1.7:

Convert from decimal to binary

$A = 17.375_{10}$ $C = 27.875_{10}$

$B = 43.625_{10}$ $D = 49.40625_{10}$

1.4.2.3 Decimal to octal conversion:

Let us convert the number 123_{10} from decimal to octal numeration system. As explained before, we just have to divide the decimal number successively by 8.

123	8	
15	8	**3**
1		**7**

$123_{10} = 173_8$

The octal digits are determined by the reminders left over by each division step. These reminders are between 0 and 7.

Exercise 1.7:

Convert the following numbers from decimal to octal:

$A = 323_{10}$ $C = 128_{10}$ $B = 452_{10}$ $D = 99_{10}$

1.4.2.4 Decimal to hexadecimal conversion:

Let us convert the number 456_{16} from decimal to hexadecimal. This conversion is obtained by repeated division of the decimal number by 16.

456	16	
28	16	**8**
1		**12 (C_{16})**

$456_{16} = 1C8_{16}$

Exercise 1.8: Convert from decimal to hexadecimal:

$A = 4523_{10}$ $B = 867_{10}$ $C = 997_{10}$ $D = 1238_{10}$