Chapter 2: Numeration Systems and Number Coding

ISTA Constantine1 University

Dr. ASSABAA Mohamed

27/08/2023

1.0

Table des matières

Objectifs

At the end of this course the learner will be able to:

• Make a conversion between different number bases.

• Process arithmetic operations and make calculations in appropriate bases.

• Do the coding of natural integers and signed integers to complement 2

• Familiarity with various coding systems.

Prerequisites:

Mathematics

Introduction

For digital information to be processed by a circuit, it must be represented in a suitable format for that circuit. To achieve this, a base B number system (B a natural number ≥ 2) needs to be chosen. Several number systems are used in digital technology, with the most commonly used systems are Decimal (base 10), Binary (base 2), Octal (base 8) and Hexadecimal (base 16).

In general, the expression of a number in base B is of the form:

$(N)B=a^n a^{n-1} \ldots a^0, a^{-1} \ldots a^{-m}$...............(1)

Where each coefficient ai is a digit whose value is between 0 and (B-1).

Any number N can be decomposed into a sum of powers of the base of its number system. This decomposition is called the polynomial form of the number N and which is given by:

$(N)_B=a_n B^n + a_{n-1} B^{n-1} + \ldots + a_1 B^1 + a_0 + a_{-1} B^{-1} + \ldots a_{-m} B^{-m}$ ....(2)

Numeration Systems

I

1. Decimal System

This is the usual numeration system in everyday life. In this system, any number N is expressed using the ten digits: 0, 1, 2, ..., 9. Thus, the base of this numeration system is B=10.

Exemple : Example :

The number 1356.724 corresponds to:

1356,724=1000+300+50+6+0.7+0.02+0.004

1356,724=$10^3+3*10^2+5*10^1+6*10^0+7*10^{-1}+2*10^{-2}+4*10^{-3}$

So: $(1356,724)_{10}=1*B^3+3*B^2+5*B^1+6*B^0+7*B^{-1}+2*B^{-2}+4*B^{-3}$

with B=10.

Conversion from Decimal system to any base

To convert a number from base 10 to any base B, it is necessary to make successive divisions by B and retain each time the rest until obtaining a quotient lower than the base B. in this case the number is written from left to right starting with the last quotient going to the first remainder.

Binary decimal conversion

Méthode : Method of successive divisions

The method of successive divisions consists of dividing successively by 2, the decimal number to be converted, until the result of the division is a zero. The corresponding binary number will be the sequence of the obtained remainders. The most significant bit of this number is the remainder of the last division.

Example :

$(27)_{10}=(11011)_2$

27 :	2	=	13 :	2	=	6 :	2	=	3 :	2	=	1 :	2	=	0
	1			1			0			1			1		

←

The remainder of the last division will be written first (reading direction).

Méthode : Method of successive subtractions

To use this method, we must first determine the successive values 2^i (i=0,...,n). Then, we will determine between which successive values of 2^i the number to be converted lies. The lower bound is then subtracted from the number. We proceed in the same way with the rest obtained, until we obtain zero as the remainder of the subtraction. The binary value will be 1 at the position of the weight used in the subtraction, and 0 at the position of unused weights.

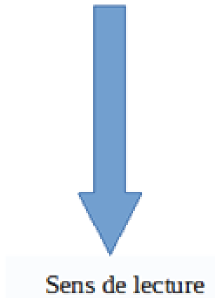
Remarque : Remark:

For decimal to another number system conversion, the process is the same. We always establish the successive values of Bi first, where B is the base of the given number system.

Exemple : Example :

Using the method of successive subtractions, convert the decimal number $(230)_{10}$ to binary:

De	230	On peut retirer	128	reste	102	**1**
De	102	On peut retirer	64	reste	38	**1**
De	38	On peut retirer	32	reste	6	**1**
De	6	On ne peut pas retirer	16	reste	6	**0**
De	6	On ne peut pas retirer	8	reste	6	**0**
De	6	On peut retirer	4	reste	2	**1**
De	2	On peut retirer	2	reste	0	**1**
De	0	On ne peut pas retirer	1	reste	0	**0**



Définition : Conversion of the fractional part of a number

We multiply successively by 2 the fractional part until we get an integer, we stop the calculations. At each multiplication, only the integer part obtained is taken into account.

Remarque : Remark:

- When successive multiplication by 2 does not give 1 after several multiplications, the calculations are stopped.
- For the integer part, we proceed by divisions as for an integer.

Exemple : Example :

To convert the number $(462.625)_{10}$ to base 2, proceed as follows:

Convert the integer part (462).

Convert the fractional part by making successive multiplications by 2 and each time retaining the digit that becomes an integer.

$(462,625)_{10}$= (?)$_2$ $(462)_{10}$=$(111001110)_2$

0,625*2=1,25

0, 25*2= 0,5

0, 5*2= 1,0

The result is: $(462, 625)_{10}$ =$(111001110,101)_2$

$(12,15)_{10}$= $(?)_2$

$(12)_{10}$=$(1100)_2$

0,1 5*2= 0,3

0 ,3*2=0,6

0 ,6*2=1,2

0 ,2*2=0,4

0 ,4*2=0,8

0 ,8*2=1,6

0 ,6*2=1,2

The result is:

$(12, 15)_{10}$ = $(1100, 001001...)_2$

2. Binary System

In this numeration system, all numbers are expressed using the digits 0 and 1; these two digits are called bits (a contraction of Binary digiT).

For the Binary system, the base of numeration is B=2.

The coefficients: a_i : 0, B-1 →0,1

This base is very practical in digital electronics to distinguish between two logic states. We write :

$(a_n a_{n-1} ... a_0)_2=a_n2^n+a_{n-1}2^{n-1}+ ... +a_1 2^1+a_0 2^0$

The right side of the equation gives the decimal value of the binary number written on the left.

a_0 : The rightmost bit is the least significant bit or the least significant bit (LSB : Low Significant Bit).

a_n : The leftmost bit is the most significant bit or the most significant bit (MSB : Most Significant Bit).

Exemple : Example:

$(1011,01)_2=1*2^3 +0* 2^2+1*2^1+1* 2^0+0*2^{-1}+1*2^{-2}$

The passage from the binary system to the decimal system is called a decoding, and the passage from the decimal system to the binary system is called coding, globally: The passage from a system X to a system Y is called a transcoding.

Decimal Binary Conversion

To convert a binary number to decimal, simply use the relation (2) by by setting B=2:

$(N)_{10}=a_nB^n +a_{n-1}B^{n-1}+ ... +a_1B^1+a_0+ a_{-1}B^{-1} + ... a_{-m} B^{-m}$

$(N)_{10}=a_n 2^n + a_{n-1} 2^{n-1} + \ldots + a_1 2^1 + a_0 + a_{-1} 2^{-1} + \ldots a_{-m} 2^{-m}$

Exemple : Example :

Convert the binary number $(11010)_2$ to decimal.

$(11010)_2 = 1*2^4+1*2^3+0*2^2+1*2^1+0*2^0$

$(11010)_2 = 16+8+2$

$(11010)_2 = (26)_{10}$

$(110001,001)_2 = 1*2^5+1*2^4+0*2^3+0*2^2+0*2^1+1*2^0+0*2^{-1}+0*2^{-2}+1*2^{-3}$

$(110001,001)_2 = 32+16+1+0.125$

$(110001,001)_2 = (49,125)_{10}$

3. Exercice

Quelle est la valeur décimale qui correspond à la valeur binaire 110001 ?

Veuillez choisir une réponse :

- 87
- 49
- 32

4. Other conversions

To convert a number from any base B_1 to another base B_2, it is necessary to go through the base 10. However, if both bases B_1 and B_2 can be written in the form of a power of 2, then it is possible to go through base 2 (binary):

Tetral base (base 4): $4=2^2$ each tetral digit is converted directly on 2 bits.

Octal base (base 8): $8=2^3$ each octal digit is converted directly on 3 bits.

Hexadecimal base (base 16): $16=2^4$ each hexadecimal digit is converted directly on 4 bits.

There is an equivalence between a number expressed in the base $B=2^n$ and a group of n bits of a number expressed in base 2. For the integer part, the bits will be grouped from right to left, while for the fractional part, they will be grouped from left to right.

Exemple : Example :

$(1\ 1001\ 1101,1101\ 0011)_2 = (19D,D3)_{16}$

$(1231)_4 = (01\ 10\ 11\ 01)_2$ $(1231)_8 = (001\ 010\ 011\ 001)_2$

5. Octal System

The octal system or base 8 includes eight digits which are: 0, 1, 2, 3, 4, 5, 6, 7. The digits 8 and 9 do not exist in this base (B=8).

Exemple : Example:

Let's write the numbers 45278 and 1274.6328:

$(4527)_8 = 4*8^3 + 5*8^2 + 2*8^1 + 7*8^0$

$(1274.632)_8 = 1*8^3 + 2*8^2 + 7*8^1 + 4*8^0 + 6*8^{-1} + 3*8^{-2} + 2*8^{-3}$

6. Hexadecimal System

The use of the base B = 16 results from the development of microcomputers. The symbols used in this base include the ten digits from 0 to 9, supplemented by the letters A (for 10), B (for 11), C (for 12), D (for 13), E (for 14), and F (for 15).

Exemple : Examples :

$(4210)_{16} = 4*16^3 + 2*16^2 + 1*16^1 + 0*16^0$

$(2A4E)_{16} = 2*16^3 + 10*16^2 + 4*16^1 + 14*16^0$

$(C1B.D5)_{16} = 12*16^2 + 1*16^1 + 11*16^0 + 13*16^{-1} + 5*16^{-2}$

Coding of numbers

II

1. Pure binary code

Corresponds to the binary decimal conversion of the number.

2. "8421" Code

It allows you to code decimal numbers between 0 and 15: 16 combination.

Decimal number	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Table 1. Code 8 4 2 1

3. BCD Code (Binary Coded Decimal)

This code preserves the advantages of both the Decimal system and the binary code. It is used by calculating machines. The Binary Coded Decimal (BCD) code involves representing each digit of a decimal number by its binary equivalent in 4 bits. Thus, we have:

Decimal code 0 1 2 3 4 5 6 7 8 9

BCD code 0000 0001 0010 0011 0100 0101 0110 0111 1000 1001

Exemple : Example :

$(19)_{10} = (00011001)_{BCD}$

$(421)_{10} = (0100\ 0010\ 0001)_{BCD}$

Remarque : Remark:

- In BCD code, a number of n digits always occupies 4n bits.
- Binary possibilities from 10 to 15 are not used.

4. Hexadecimal code

The Hexadecimal system or base 16 contains sixteen elements which are {0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F}.

The numbers A, B, C, D, E represent 10, 11, 12, 13, 14 and 15 respectively.

Exemple : Example :

$(D62C)_{16} = (13*16^3 + 6*16^2 + 2*16^1 + 12*16^0)_{10} = (54828)_{10}$

$(A2B,E1)_{16} = 10*16^2 + 2*16^1 + 11*16^0 + 14*16^{-1} + 1*16^{-2} = (2603,8789)_{10}$

$(468)_{10} = (1D4)_H$

468 :	16	=	29:	16	=	1:	16	=	0
	4			13			1		
	4			D			1		

Hexadecimal Number	Binary			
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
A	1	0	1	0
B	1	0	1	1
C	1	1	0	0
D	1	1	0	1
E	1	1	1	0
F	1	1	1	1

Table 2. Code Hexadécimal

Arithmetic operations

III

Arithmetic operations are performed in any base B using the same methods as in base 10.

Carry or Borrow occurs when reaching or exceeding the value B of the base.

1. Addition

It is sufficient to know that:

0+0=0

0+1=1

1+0=1

1+1=0 Carry 1

$$\begin{array}{r} 1\,1\,0\,1\,1\,1 \\ +\quad\quad\quad\;\; \\ 1\,1\,1\,0\,1\,0 \\ \hline = 1\,1\,1\,0\,0\,0\,1 \end{array}$$

In hexadecimal :

$$\begin{array}{r} 1\,A \\ + \quad \\ C\,7 \\ \hline = E\,1 \end{array} \equiv \begin{array}{r} 0001\;1010 \\ + \quad\quad\quad\quad \\ 1100\;0111 \\ \hline = \; 1110\;0001 \end{array}$$

BCD Addition :

Exemple : Example 1 :

```
  13            0001 0011
+          ≡  +
  26            0010 0110
-----         -----------
= 39          = 0011 1001
```

Exemple : Example 2 :

```
  18            0001 1000
+             +
  29            0010 1001
-----         -----------
= 47          = 0100 0001
              +
                     0110
              -----------
              = 0100 0111
```

We correct by adding 6 to the quartet >9

Exemple : Example 3 :

```
  65            0110 0101
+             +
  68            0110 1000
-----         -----------
= 133         = 1100 1101
              +
                0110 0110
              -----------
              =1 0011 0011
```

2. Substraction

It is sufficient to know that:

0-0=0

0-1=1 Borrow 1

1-0=1

1-1=0

Exemple : Example :

10-9=+1

$$\begin{array}{rcccc} & 1 & 0 & 1 & 0 \\ - & 1 & 0 & 0 & 1 \\ \hline = \underline{0}| & 0 & 0 & 0 & 1 \end{array}$$

9-10=-1

$$\begin{array}{rcccc} & 1 & 0 & 0 & 1 \\ - & 1 & 0 & 1 & 0 \\ \hline = \underline{1}| & 1 & 1 & 1 & 1 \end{array} \rightarrow \text{(-1) (CP2)}$$

Définition : 2's complement subtraction

The advantage of 2's complement subtraction is the ability to transform subtraction into addition.

By definition, the 2's complement of a binary number is the complement bit by bit of this number +1, with 1 added to the least significant bit. A-B=A+CP_2(B).

$$CP_2(B) = \bar{B} + 1$$

Exemple

9-10=9+CP_2(10)

$(10)_{10}=(1010)_2$

CP2(10)=

9-10=9+CP_2(10)

$(10)_{10}=(1010)_2$

CP_2(10)=

$$\begin{array}{rcccc} & 0 & 1 & 0 & 1 \\ + & & & & 1 \\ \hline = & 0 & 1 & 1 & 0 \end{array}$$

Then:

$$\begin{array}{rcccc} & 1 & 0 & 0 & 1 \\ + & 0 & 1 & 1 & 0 \\ \hline = \underline{0}| & 1 & 1 & 1 & 1 \end{array}$$

No carry, thus the result < 0.

$CP_2(1111)=0000$

$$
\begin{array}{rcccc}
 & 0 & 0 & 0 & 0 \\
+ & & & & \\
 & & & & 1 \\
\hline
= & 0 & 0 & 0 & 1
\end{array}
$$

3. Multiplication

Binary multiplication is very simple, here is the multiplication table:

0×0=0

0×1=0

1×0=0

1×1=1

Exemple : Example 1 :

6×7=42

$$
\begin{array}{ccccccc}
 & & & & 1 & 1 & 0 \\
 & & & \times & & & \\
 & & & & 1 & 1 & 1 \\
\hline
= & & & & 1 & 1 & 0 \\
 & & & 1 & 1 & 0 & . \\
 & & 1 & 1 & 0 & . & . \\
\hline
= & 1 & 0 & 1 & 0 & 1 & 0
\end{array}
$$

Exemple : Example 2 :

10,75×3=32,25

```
      1  0  1  0, 1  1
   ×
                  1  1
   -------------------
   =  1  0  1  0, 1  1
   1  0  1  0, 1  1  .
-----------------------
=  1  0  0  0  0, 0  1
```

4. Division

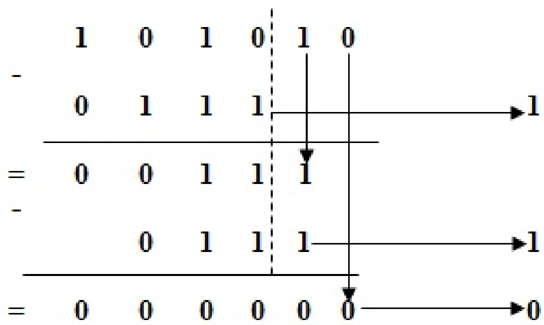
The principle of binary division is similar to that of decimal division, but simpler due to the fact that each partial quotient is either 1 (division possible) or 0 (division not possible). Division is the inverse operation of multiplication, in the sense that a number is repetitively subtracted from another until it is no longer possible, with each time a shift to the right.

The first step is to subtract, starting from the left, the divisor from the dividend. If subtraction is not possible, the divisor is shifted one position to the right, then the subtraction is carried out. The next subtraction takes place between the result of the previous subtraction, augmented on the right of the next bit of the dividend, following the previously stated rule. This step is repeated until all the bits of the dividend are exhausted. At each subtraction, we write 1 to the result, otherwise we put 0.

Exemple : Example1 :

Do the following division:

42/7=6

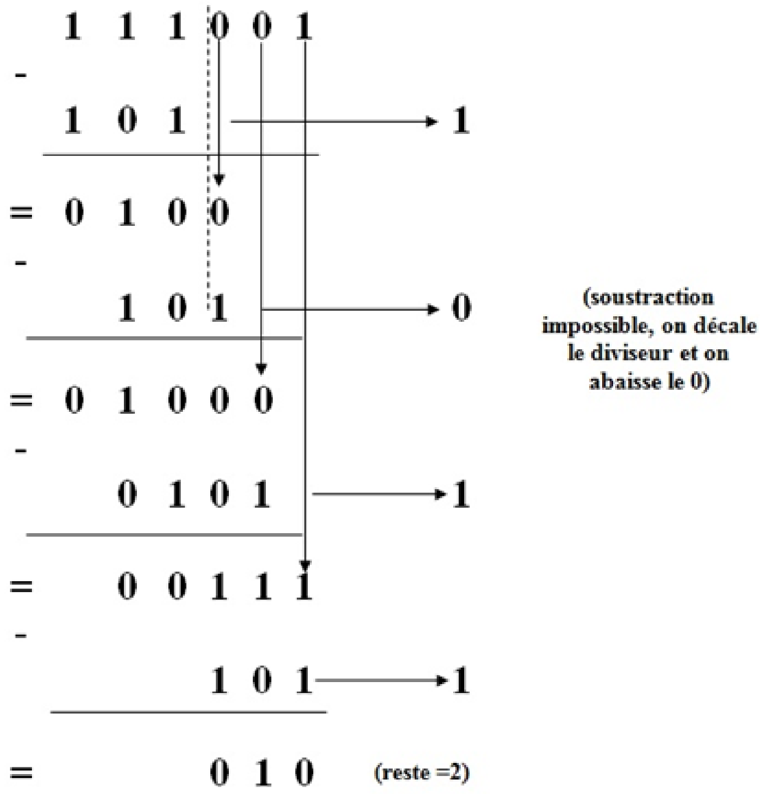


The result is thus 110 with a remainder=0 (6 and rest=0)

Exemple : Example2 :

Do the following division:

57/5=11 reste=2


1 1 1 0 0 1
-
1 0 1
1
= 0 1 0 0
-
1 0 1
0
(soustraction impossible, on décale le diviseur et on abaisse le 0)
= 0 1 0 0 0
-
0 1 0 1
1
= 0 0 1 1 1
-
1 0 1
1
= 0 1 0
(reste =2)

IV

Exercises:

1. Exercise 01 :

Convert the following numbers to decimal:

$(01001011)_2$, $(1245)_8$, $(3C5)_{16}$, $(1001\ 1000)_{BCD}$.

2. Exercice 02 :

Convert the following numbers to binary:

$(1523)_{10}$, $(74)_8$, $(A94)_{16}$, $(124)_7$.

3. Exercice 03 :

Convert decimal numbers:

$(108)_{10}$ en octal in octal

$(1023)_{10}$ in hexadecimal

$(12{,}524)_{10}$ in binary

$(51{,}225)_{10}$ in base 7

4. Exercice 04 :

Perform the following operations in binary:

$(254+36)_{10}$,

$(A049+0AFC)_{16}$,

$(104-111)_{10}$,

$(255\times127)_{10}$,

$(294/14)_{10}$,

$(57/5)_{10}$

Bibliographie

Y. Granjon, B. Estibals, S. Weber, Electronique : tout le cours en fiches, IUT, Licence, Ecoles d'Ingénieurs, DUNOD, Paris, 2015

P. Horowitz, W. Hill, Traité de l'électronique analogique et numérique - Volume 2, Publitronic - Elektor, 1997

T. Ndjountche, Electronique numérique 1 : Circuits logiques combinatoires, ISTE editions, 2016

N. Mansouri, "Les Systèmes Logiques", Tome 1, EUMC