COURS 3

SYSTEME DE NUMERATION

CLASSE : 1° S .

DATE : / / 09

1. Base d'un système de numération

1.1 Système décimal.

C'est le système de base 10 que nous utilisons tous les jours. Il comprend dix symboles différents : **0, 1, 2, 3, 4, 5, 6, 7, 8 et 9.**

Exemple du nombre 2356 de ce système : nous l'écrivons $N=(2356)_{10}$. L'indice 10 indique la base dans laquelle le nombre est écrit.

Ce nombre N peut être écrit sous la forme suivante :

$\mathbf{N = 2.10^3 + 3.10^2 + 5.10^1 + 6.10^0 = 2000+300+50+6=2356.}$

1.2 Système binaire.

Ce système dit de base 2 comprend **deux symboles différents : 0 et 1**. Chacun d'eux est aussi appelé bit *(**B**inary dig**IT** = BIT).*

Exemple : $N= (10110)_2$. Ce nombre N peut être écrit sous la forme suivante :

$N = (10110)_2 = \mathbf{1.2^4 + 0.2^3 + 1.2^2 + 1.2^1 + 0.2^0 =(16+4+2)_{10}= (22)_{10}}.$

En utilisant n bits, on peut former $\mathbf{2^n}$ **nombres différents** et le plus grand d'entre eux **est égal à $\mathbf{2^n-1}$**. Par exemple avec un dispositif à **8 bits (n = 8), on peut représenter $\mathbf{2^8}$ = 256 nombres différents dont le plus grand est $\mathbf{(11111111)_2 = (255)_{10}}$.**

1.3 Système hexadécimal.

Ce système dit de base 16 comprend **seize symboles différents : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.**

Exemple : $N= (AC53)_{16}$. Ce nombre N peut être sous la forme suivante :

$N = (AC53)_{16} = \mathbf{A.16^3 + C.16^2 + 5.16^1 + 3.16^0 = 10.16^3 + 12.16^2 + 5.16^1 + 3.16^0 = (44115)_{10}}$

1.4 Correspondance entre nombres de différentes bases.

Décimal	Binaire	Hexadécimal	Décimal	Binaire	Hexadécimal
0	0000	0	**8**	1000	8
1	0001	1	**9**	1001	9
2	0010	2	**10**	1010	A
3	0011	3	**11**	1011	B
4	0100	4	**12**	1100	C
5	0101	5	**13**	1101	D
6	0110	6	**14**	1110	E
7	0111	7	**15**	1111	F

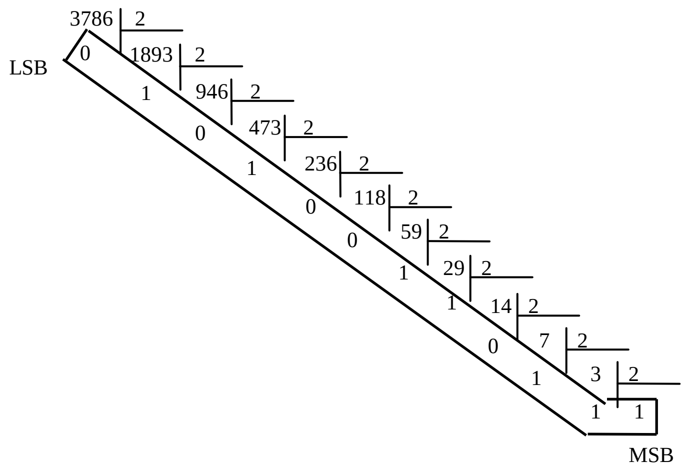
2. Changement de base.

2.1 Conversion d'un nombre décimal en un nombre d'une autre base

Méthode : diviser le nombre décimal à convertir par la base b et conserver le reste de la division. Le quotient obtenu est divisé par b et conserver le reste. Il faut répéter l'opération sur chaque quotient obtenu.

Les restes successifs sont écrits, en commençant par le dernier, de la gauche vers la droite pour former l'expression de $(N)_{10}$ dans le système de base b. Cette méthode est dite « Méthode de la division successives ».

Exemple : Convertir $N = (3786)_{10}$ en binaire $N = (111011001010)_2$.



2.2 Conversion d'un nombre hexadécimal en binaire.

Chaque symbole du nombre écrit dans le système hexadécimal est remplacé par son équivalent écrit dans le système binaire.

Exemple : Convertir $N = (ECA)_{16} = (\underbrace{1110}_{E}\ \underbrace{1100}_{C}\ \underbrace{1010}_{A})_2$.

2.3 Conversion d'un nombre binaire en hexadécimal.

C'est l'inverse de la précédente. Il faut donc regrouper les 1 et les 0 du nombre par 4 en commençant par la droite, puis chaque groupe est remplacé par le symbole hexadécimal correspondant.

Exemple : Convertir $N = (1\ 1000\ 0110\ 1111)_2 = (\underset{0001}{\underline{1}}\ \underset{1000}{\underline{8}}\ \underset{0110}{\underline{6}}\ \underset{1111}{\underline{F}})_{16}$.

3. Les autres systèmes de codage.

3.1 Code gray ou binaire réfléchi.

C'est le système de codage qui, contrairement au code binaire pur est arrangé de manière à ne faire changer d'état qu'une variable à la fois d'une ligne à l'autre.

Code binaire pur				
Nombre$_{(10)}$	2^3	2^2	2^1	2^0
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

Code binaire réfléchi				
Nombre$_{(10)}$	**a**	**b**	**c**	**d**
0	0	0	0	0
1	0	0	0	1
2	0	0	1	1
3	0	0	1	0
4	0	1	1	0
5	0	1	1	1
6	0	1	0	1
7	0	1	0	0
8	1	1	0	0
9	1	1	0	1

3.2 Le binaire signé : Représentation des nombres positifs et négatif.

Dans ce type de représentation, sur un format de 8 bits par exemple, il reste 7 bits significatifs ; Le 8eme bit indique donc le signe du nombre codé.

Exemple *: Sur 8 bits on peut représenter des nombres signés de -128 à +127.*

➩ *Lorsque le bit le plus à gauche est un 0, le nombre est positif.*

0	1	1	1	1	1	1	1

Ici le résultat est +127 comme en binaire pur.

➩ *Lorsque le bit le plus à gauche est un 1, le nombre est négatif.*

1	1	1	1	1	1	1	1

Ici le résultat est -128.

3.3 Le code ASCII.

Historique : Dans les premières années du télégraphes, il fut défini un code international pour communiquer des messages écrits. Le code ASCII (Américan Standard Code of Information Interchange) est maintenant généralisé pour tous les traitements informatiques. Il permet de coder tout caractère alphanumérique en binaire. Par exemple le code ASCII est le code utilisé pour coder les touches du clavier d'un micro-ordinateur.

L'ensemble des codes ASCII correspondant à chaque caractère alphanumérique est contenue dans un tableau de conversion *(Voir documentation technique).*

4. Quelques définitions.

4.1 Bit.

➩ Le bit est une unité élémentaire d'information ne pouvant prendre que deux valeurs distinctes *(Notées 0 ou 1).*

4.2 Mot binaire.

➩ En informatique, l'unité de traitement de l'information est le mot binaire.

Nota : - Un ensemble de 4 bits (Ou Mot de 4 bits) = Quartet
- Un ensemble de 8 bits (Ou mot de 8 bits) = Octet.

4.3 Octet.

➩ Un octet est composé de 8 bits :

1 1 0 1 0 1 0 1

b7 b6 b5 b4 b3 b2 b1 b0

MSB LSB

On distingue :

- Le bits de poids fort b7 *(MSB : Most Significant Bit).*
- Le bits de poids faible b0 *(LSB : Least Significant Bit).*

4.4 Kilo-octet *(Koctet)* :

➩ Un Kilo-octet est composé de 1024 octets *(2^{10} = 1024)*