

Architecture des Ordinateurs 1



AMGHAR Djazia

Département informatique
université TLEMCEN

djaziaamghar@gmail.com

1

Table des matières

I - Système de numération(number system)	3
1. objectifs	3
2. Introduction	3
3. Codage de l'information	3
3.1. ALPHABET A (SET OF SYMBOLS OR DIGITS)	4
3.2. Des règles d'écritures des nombres (Juxtaposition de symboles)	4
3.3. Système de numération positionnel pondéré à base b	4
4. Bases de numération (Binaire, Octale et Hexadécimale)	5
4.1. Système binaire	5
4.2. Système Octale	5
4.3. Système Hexadécimale	5
5. Codage des données:	5
5.1. Transcodage ou Conversion de bases	5
5.2. Opérations dans le système binaire :	8
6. Exercice	11

I Système de numération (number system)

1. objectifs

Expliquer comment sont représentés les nombres dans les systèmes de numération.

Effectuer des conversions entre bases différentes.

Expliquer comment est représentée l'information dans le système binaire et comment sont réalisés les calculs dans ce système.

2. Introduction

Les informations traitées par les ordinateurs sont de différentes natures :

- Nombres, texte,
- Images, sons, vidéo,
- Programmes, ...

Dans un ordinateur, elles sont toujours représentées sous forme binaire (**BIT : Binary digIT**) : une suite de 0 et de 1 .

Par exemple: 01001011.

Le terme bit (b minuscule dans les notations) signifie « binary digit », c'est-à-dire 0 ou 1 en numérotation binaire.

- Il s'agit de la plus petite unité d'information manipulable par une machine numérique.
Il est possible de représenter physiquement cette information binaire par un phénomène physique (un signal électrique, magnétique..), qui, au-delà d'un certain seuil, correspond à la valeur 1.

3. Codage de l'information

permet d'établir une correspondance qui permet sans ambiguïté de passer d'une représentation (dite externe) d'une information à une autre représentation (dite interne : sous forme binaire) de la même information, suivant un ensemble de règles précises.

Exemple

Le nombre 12 :

- 12 est **la représentation externe** du nombre douze
- 1100 est **la représentation interne** de 12 dans la machine

3.1. ALPHABET A (SET OF SYMBOLS OR DIGITS)

Un système de numération décrit la façon avec laquelle les nombres sont représentés. Il est défini par :

- Un alphabet A (*Set of Symbols or Digits- A*) : L'alphabet d'un système de numération est l'ensemble de symboles ou de chiffres qui sont utilisés pour représenter les nombres.

Exemple

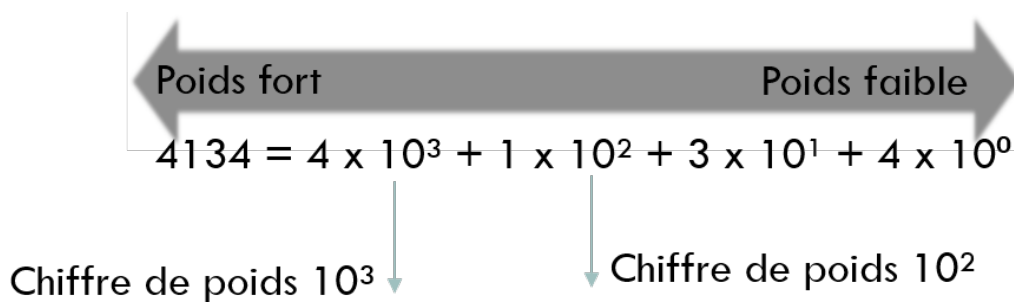
le système décimal (decimal system) utilise les chiffres(number) de 0 à 9 comme son alphabet.

3.2. Des règles d'écritures des nombres (Juxtaposition de symboles)

Les règles d'écriture définissent comment les symboles de l'alphabet sont combinés pour représenter des nombres. La position des chiffres dans un nombre est cruciale et détermine la valeur du chiffre. Cela suit généralement une règle de puissances de la base du système.

Exemple

le nombre 4134 s'écrit comme suit :


$$4134 = 4 \times 10^3 + 1 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$$

Chiffre de poids 10^3 Chiffre de poids 10^2

3.3. Système de numération positionnel pondéré à base b

Un système de numération positionnel pondéré à base b est défini sur un alphabet de b chiffres :

Soit :

représentation en base b sur n chiffres

- : est un chiffre de l'alphabet de poids i (position i).
- : chiffre de poids 0 appelé le chiffre de poids faible.
- : chiffre de poids n-1 appelé le chiffre de poids fort.

Exemple

$$4134 = 4 \times 10^3 + 1 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$$

$$\sum_{i=0}^{n-1} a_i b^i$$

Exemple

base10={1,2,3,4,5,6,7,8,9}

$$N = 4 \times 10^3 + 1 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$$

4. Bases de numération (Binaire, Octale et Hexadécimale)

4.1. Système binaire

Système binaire (binary system) (b=2) utilise deux chiffres : {0,1} C'est avec ce système que fonctionnent les ordinateurs .

4.2. Système Octale

Système Octale (octal system) (b=8) utilise huit chiffres : {0,1,2,3,4,5,6,7} Utilisé il y a un certain temps en Informatique, il permet de coder 3 bits par un seul symbole.

4.3. Système Hexadécimale

Système Hexadécimale(hexadecimal system) (b=16) utilise 16 chiffres :

{0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F}

Où : A=, , , et

Cette base est très utilisée dans le monde de la micro informatique. Elle permet de coder 4 bits par un seul symbole.

5. Codage des données:

On distingue 2 types de données:

- non numériques (codage assez simple car aucune opération arithmétique ou logique ne sera appliquée sur ces données, une table de correspondance suffit)
- numériques (codage complexe qui doit faciliter la mise en place de circuits réalisant les opérations arithmétiques).

5.1. Transcodage ou Conversion de bases

Le transcodage (ou conversion de base) est l'opération qui permet de passer de la représentation d'un nombre exprimé dans une base à la représentation du même nombre mais exprimé dans une autre base.

Les conversions possibles sont les suivantes:

- Décimale vers Binaire, Octale et Hexadécimale
- Binaire vers Décimale, Octale et Hexadécimale
- Octale vers Décimale, Binaire et Hexadécimale
- Hexadécimale vers Décimale, Binaire et Octale

Correspondance entre les systèmes les plus utilisés

Décimal	Octal	Binaire	Hexadécimal
0	0	0	0
1	1	1	1
2	2	10	2
3	3	11	3
4	4	100	4
5	5	101	5
6	6	110	6
7	7	111	7
8	10	1000	8
9	11	1001	9
10	12	1010	A
11	13	1011	B
12	14	1100	C
13	15	1101	D
14	16	1110	E
15	17	1111	F
16	20	10000	10

5.1.1. Binaire, octal ou hexadécimal vers décimal:

Exemple

trouvez la représentation décimal des Nombre

Base 2:

:

$$= 3 + 16 + 64$$

=

Base16:

trouvez la représentation décimal des nombre

5.1.2. Du Décimal vers une Base b(2,8,16)

La règle à suivre est les divisions successives :

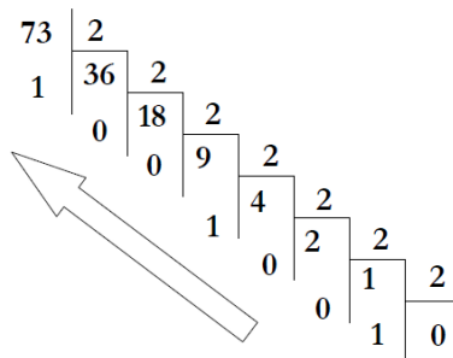
- On divise le nombre par la base b puis le quotient par la base et ainsi de suite jusqu'à obtenir 0.
- La suite des restes correspond aux symboles de la base visée.
- On obtient en premier le chiffre de poids
- faible et en dernier le chiffre de poids fort.

a) Décimal vers Binaire :

Décimal vers Binaire

Exemple

Soit un nombre représenté en base décimale 10 Trouver sa représentation binaire :



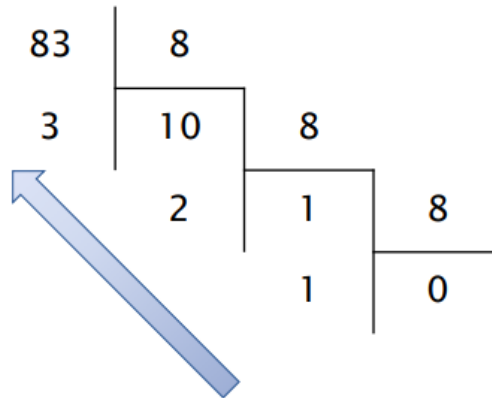
$N = (1001001)_2$

b) Décimal vers Octal :

🔗 Exemple

Soit un nombre représenté en base décimale 10

Trouver sa représentation en octal :

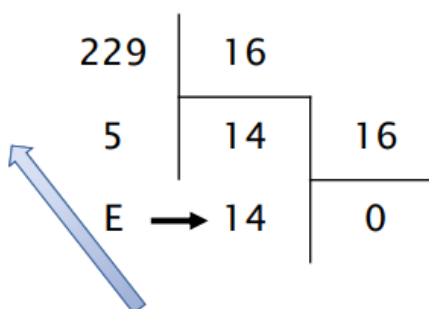


c) Décimal vers Hédécimal

🔗 Exemple

Soit un nombre représenté en base décimale 10

Trouver sa représentation en hexadécimal :



5.1.3. Binaire vers octal ou hexadécimal :

Première solution :

convertir le nombre en base binaire vers la base décimale puis convertir ce nombre (en base 10) vers la base souhaitée (8 ou 16).

Exemple

Deuxième solution : (Utiliser les tables de correspondance)

- Binaire vers octale : regroupement des bits en des groupes de trois bits de droite à gauche puis remplacer chaque groupe par le symbole correspondant dans la base 8.
- Binaire vers Hexadécimale : regroupement des bits en des groupes de quatre bits de droite à gauche puis remplacer chaque groupe par le symbole correspondant dans la base 16.

Exemple

$$(1011100)_2 = \frac{001}{1} \frac{011}{3} \frac{100}{4} = (134)_8$$

$$(1011100)_2 = \frac{0101}{5} \frac{1100}{C} = (5C)_{16}$$

5.2. Opérations dans le système binaire :

- Addition +
- Soustraction -
- Multiplication *
- Division /

5.2.1. Addition :

On écrit les nombres sur des lignes successives en les mettant en colonne, en partant de la droite ; ensuite on additionne les chiffres de chaque colonne en commençant par celle de droite.

On suit les règles suivantes :

$$0 + 0 = 0 ,$$

$$1 + 0 = 1 ,$$

$$0 + 1 = 1 ,$$

$$1 + 1 = 0 \text{ et on retient } 1, \text{ soit } 10$$

Exemple

additionner les nombres

$$\begin{array}{r}
 \begin{array}{ccccccc}
 & & 1 & & 1 & & 1 & & 1 \\
 & & 1 & 1 & 0 & 1 & 1 & & \\
 + & 1 & 0 & 1 & 0 & 1 & & & \\
 \hline
 1 & 1 & 0 & 0 & 0 & 0 & & &
 \end{array}
 \end{array}$$

Attention

: si on dépasse le nombre de bits autorisé alors on appelle ça un dépassement de capacité ou **overflow**

Remarque

L'entier maximal pouvant être codé dépendra du nombre de bits que l'on réserve pour coder un nombre.

En général les entiers sont codés sur un mot e.g., pour un ordinateur 32 bits : $2^{32} - 1 = 4\,294\,967\,295$.

Dépassement de capacité overflow

Lorsque par exemple le résultat d'une opération sur des nombres produit un nombre plus grand que la taille du mot prévu pour représenter ces nombres (e.g., bit de retenue).

5.2.2. Soustraction :

On écrit les nombres sur des lignes successives en les mettant en colonne, en partant de la droite ; ensuite on soustrait les chiffres de chaque colonne en commençant par celle de droite. On suit les règles suivantes :

$$0 - 0 = 0$$

$$1 - 0 = 1$$

$$1 - 1 = 0,$$

$0 - 1 = 1$ et on retient 1 (dans ce cas où la soustraction est impossible donc on emprunte 1 puis on le retourne dans la deuxième colonne)

Exemple

effectuer l'opération suivante :

$$\begin{array}{r}
 \begin{array}{ccccccc}
 & 1 & & 0 & & 0 & & 1 & & 1 \\
 - & 1 & & 1 & & 1 & & 1 & & 0 \\
 \hline
 0 & & 0 & & 1 & & 0 & & 1
 \end{array}
 \end{array}$$

5.2.3. Multiplication :

On écrit les nombres sur des lignes successives en les mettant en colonne, en partant de la droite ; ensuite on effectue la multiplication comme le

décimal. On suit les règles suivantes :

$$0 * 0 = 0 ,$$

$$1 * 0 = 0 ,$$

$$1 * 1 = 1 .$$

Exemple

effectuer l'opération suivante :

$$\begin{array}{r}
 1101 \\
 * \quad 101 \\
 \hline
 1101 \\
 0000. \\
 1101. . \\
 \hline
 1000001
 \end{array}$$

5.2.4. Division :

la division s'effectue selon les règles arithmétiques traditionnelles et elle basée sur la soustraction en suivant les règles suivantes :

soustraction possible le résultat est 1, et 0 dans le cas contraire

$$0 / 0 \text{ indéterminé}$$

$$1 / 0 \text{ indéterminé}$$

$$0 / 1 = 0$$

$$1 / 1 = 1$$

Exemple

effectuer les opérations suivantes :

$$\begin{array}{r}
 100100 \\
 - 1100 \\
 \hline
 001100 \\
 - 1100 \\
 \hline
 0000
 \end{array}
 \quad
 \begin{array}{r}
 1100 \\
 \hline
 11
 \end{array}
 \qquad
 \begin{array}{r}
 111100 \\
 - 1101 \\
 \hline
 001000 \\
 - 001000 \\
 \hline
 000000
 \end{array}
 \quad
 \begin{array}{r}
 1101 \\
 \hline
 100
 \end{array}$$

6. Exercice

Exercice

Donner la valeur décimale des entiers suivants, la base dans laquelle ces entiers sont codés

étant précisé

A1BE en hexadécimal (base 16) ;

☐ 41 406

☐ 41523