

CODAGE INFORMATIQUE

INFORMATIQUE I

Author: CHERIEF Houria

Département de science et de la matière

houria.cherief1@univ-dbkm.dz

Table of Contents

- 1 INTRODUCTION
- 2 Systèmes de Numération
- 3 Conversion du système Décimal vers une base quelconque
- 4 Conversion du système Binaire vers une base quelconque
- 5 Conversion du système Octal vers une base quelconque
- 6 Conversion du système Hexadécimal vers une base quelconque
- 7 Les opérations Arithmétiques
- 8 Les opérateurs et les Expressions

INTRODUCTION

- Quelle que soit la nature de l'information traitée par un ordinateur (image, son, texte, vidéo), elle l'est toujours sous la forme d'un ensemble de nombres écrits en base 2, par exemple 01001011.
- Nombreux systèmes de numération sont utilisés en technologie numérique. Les plus utilisés sont les systèmes :Décimal (base 10), Binaire (base 2), Octal (base 8) et Hexadécimal (base 16).

Systèmes Informatiques!

Systèmes de Numération

Il existe quatre systèmes de numération qui sont :

Système	Base	Symboles Utilisee
Décimal	base 10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Binaire	base 02	0, 1
Octal	base 08	0, 1, 2, 3, 4, 5, 6, 7
Hexadécimal	base 16	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

Nous utilisons le système décimal (base 10) dans nos activités quotidiennes, et En informatique, on utilise très fréquemment le système binaire (base 2).

On utilise aussi très souvent le système hexadécimal (base 16) du fait de sa simplicité d'utilisation et de représentation pour les mots machines (il est bien plus simple d'utilisation que le binaire).

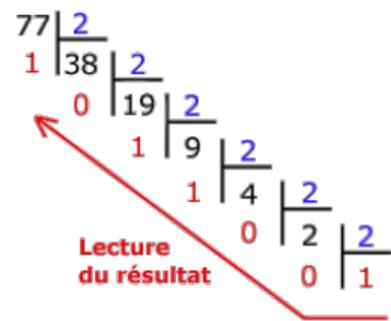
Conversion du système Décimal vers une base quelconque

1 **Décimal / Binaire** : Pour convertir un nombre décimal en un nombre binaire, diviser continûment le nombre décimal par la valeur 2, retenir la suite des restes de chaque division en commençant de bas en haut pour avoir le nombre équivalent.

Exemple :
convertir $(77)_{10}$

Solution :
77 s'écrit donc en base 2 : 1001101.

Figure: Méthode de conversion.



Conversion du système Décimal vers une base quelconque

- Décimal / Octal** : Pour convertir un nombre décimal en un nombre octal, diviser continûment le nombre décimal par la valeur 8, retenir la suite des restes de chaque division en commençant de bas en haut pour avoir le nombre équivalent.

Exemple :

convertir $(77)_{10}$

Solution :

77 s'écrit donc en base 8 : 115.

Methode :

$$\frac{77}{8} = ((9 \times 8) + 5)$$

$$\frac{9}{8} = ((1 \times 8) + 1)$$

$$\frac{1}{8} = ((0 \times 8) + 1)$$

Conversion du système Décimal vers une base quelconque

- sur Décimal / Hexadécimal :** Pour convertir un nombre décimal en un nombre hexadécimal, diviser continûment le nombre décimal par la valeur 16, retenir la suite des restes de chaque division en commençant de bas en haut pour avoir le nombre équivalent.

Exemple :

convertir $(77)_{10}$

Solution :

77 s'écrit donc en base 16 : 4D.

Methode :

$$\frac{77}{16} = ((4 \times 16) + 13)$$

$$\frac{1}{8} = ((0 \times 16) + 4)$$

Pour faire la conversion d'un nombre d'une base quelconque B1 vers une autre base B2 il faut passer par la base 10. Mais si la base B1 et B2 s'écrivent respectivement sous la forme d'une puissance de 2 on peut passer par la base 2 (binaire) :

- Tétrale (base 4) : $4 = 2^2$ chaque chiffre tétral se convertit tout seul sur 2 bits.
- Octale (base 8) : $8 = 2^3$ chaque chiffre octal se convertit tout seul sur 3 bits.
- Hexadécimale (base 16) : $16 = 2^4$ chaque chiffre hexadécimal se convertit tout seul sur 4 bits.

Chiffre Octal	Chiffre Hexadécimal	Nombre binaire		Nombre Décimal
		Pour Octal	Pour Hexadécimal	
0	0	000	0000	0
1	1	001	0001	1
2	2	010	0010	2
3	3	011	0011	3
4	4	100	0100	4
5	5	101	0101	5
6	6	110	0110	6
7	7	111	0111	7

	8		1000	8
	9		1001	9
	A		1010	10
	B		1011	11
	C		1100	12
	D		1101	13
	E		1110	14
	F		1111	15

Fig. 1. - Correspondances entre chiffres Octaux, Hexadécimaux et respectivement triplets et quartets binaires.

Conversion du système Binaire vers une base quelconque

Le passage du système binaire au système décimal s'appelle un décodage, et le passage du système décimal au système binaire s'appelle codage.

sur Binaire / Décimal : Pour convertir un nombre binaire en décimal, il suffit d'utiliser la relation (2) en posant $B=2$:

$$(N)_{10} = a_n B_n + a_{n-1} B_{n-1} + \dots + a_1 B_1 + a_0 + a_{-1} B_{-1} + \dots a_{-m} B_{-m}$$

Exemple 1 :

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

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

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

Exemple 2 :

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

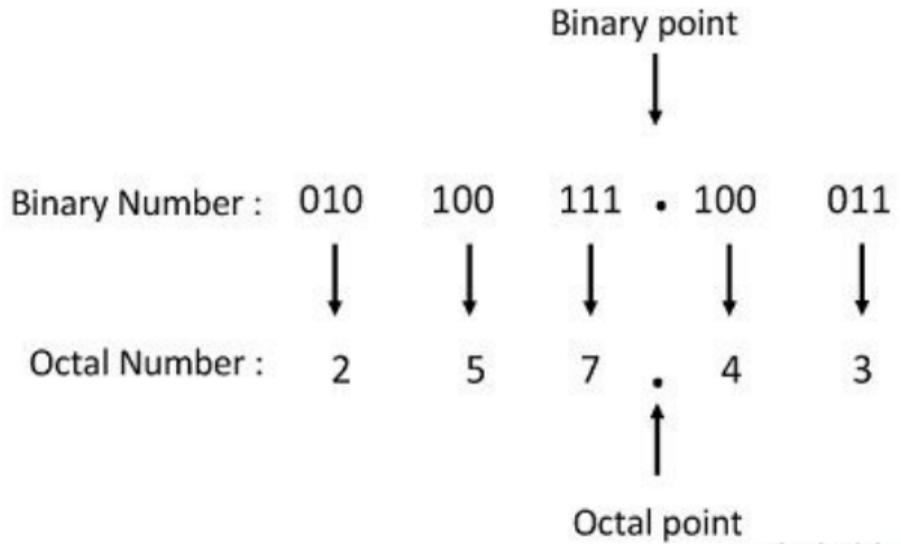
$$(110001,001)_2 = 32 + 16 + 1 + 0.125$$

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

Conversion du système Binaire vers une base quelconque

sur Binaire / Octal : Pour convertir un nombre binaire en octal, chaque chiffre octal se convertit tout seul sur 3 bits.

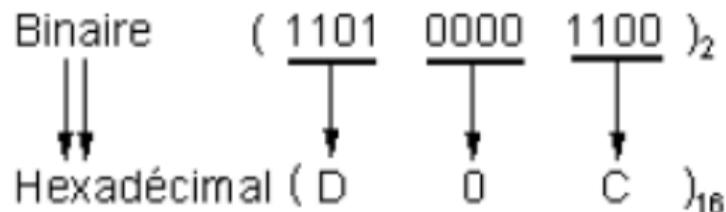
Exemple :



Conversion du système Binaire vers une base quelconque

- **Binaire / Hexadécimal** : Pour convertir un nombre binaire en hexadécimal, chaque chiffre hexadécimal se convertit tout seul sur 4 bits.

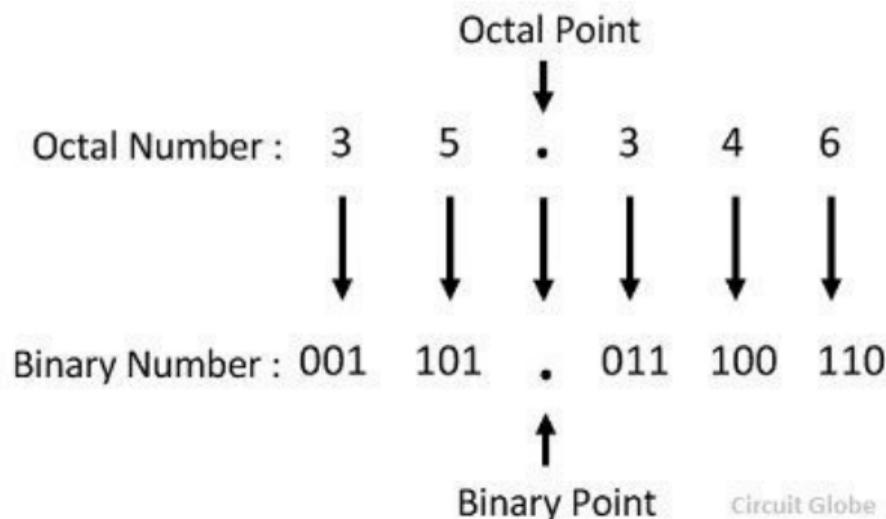
Exemple :



Conversion du système Octal vers une base quelconque

sur **Octal / Binaire** : Pour convertir un nombre octal en binaire, chaque 3 bits convertit en un chiffre octal.

Exemple :



Conversion du système Octal vers une base quelconque

Sur Octal / Décimal : Pour convertir un nombre octal en décimal, suit l'exemple suivant :

Exemple :

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

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

Sur Octal / Hexadécimal : Pour convertir un nombre octal en hexadécimal, on passe par **la conversion d'un nombre octal au un nombre binaire puis on convertit le dernier à un nombre hexadécimal.**

Conversion du système Hexadécimal vers une base quelconque

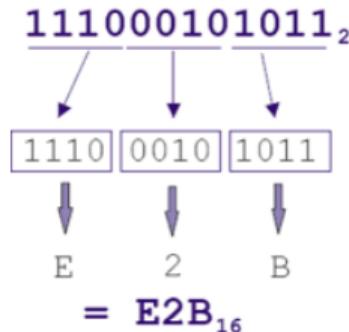
Hexadécimal/ Décimal : Pour convertir un nombre hexadécimal en décimal, suit l'exemple suivant :

Exemple :

$$(4A27)_{16} = 4 * 16^3 + 10 * 16^2 + 2 * 16^1 + 7 * 16^0$$

Hexadécimal / Binaire : Pour convertir un nombre Hexadécimal en binaire, chaque 4 bits convertit en un chiffre Hexadécimal.

Exemple :



Les opérations Arithmétiques

Les opérations arithmétiques s'effectuent en base quelconque B avec les mêmes méthodes qu'en base 10. Une retenue ou un report apparaît lorsque l'on atteint ou dépasse la valeur B de la base.

Sur Addition : Il suffit de savoir que : $0 + 0 = 0 / 0 + 1 = 1 / 1 + 0 = 1 / 1 + 1 = 0$
Retenue1

Exemple :

$$\begin{array}{r}
 00011001 \\
 + 00011011 \\
 \hline
 00110100
 \end{array}$$

Les opérations Arithmétiques

- Sur Soustraction en complément à 2 :** L'intérêt de la soustraction en complément à 2 est la possibilité de transformer la soustraction en addition. Par définition le complément à 2 d'un nombre binaire c'est le complément bit à bit de ce nombre +1, le 1 étant rajouté au bit le moins significatif.

$$A - B = A + (\bar{B} - 1)$$

Exemple :

			1	1	
		1	1	0	1
-		0	1	1	1
		1	1		
$=_2$		1	1	0	

Les opérateurs et les Expressions

- **Une expressions** : est une combinaison cohérente éventuellement parenthésée, d'opérandes et opérateurs qui sont évalués pour donner une valeur.
 - **Opérande** : peut prendre les formes suivantes (**Valeur, Constante, Variable, Fonction**).
 - **Opérateurs** : peut être sous ses types (**Arithmétiques, Relationnels, Logique**)
- **Opérateurs Arithmétiques** : effectuent divers calculs arithmétiques comme l'addition (+), soustraction (-), Multiplication (*), Division (/), puissance.
- **Opérateurs Relationnels (de Comparaison)** : ces opérateurs comparent les valeurs de chaque côté de l'opérande et déterminent la relation entre elles ($\geq, \leq, >, <, =, \neq$).

Les opérateurs et les Expressions

Opérateurs Logiques : sont

- **Conjonction** AND (&& ET)
- **Disjonction** OR (|| OU)
- **Négation** NOT (! NON)

<i>A</i>	<i>B</i>	<i>A ou B</i>	<i>A</i>	<i>B</i>	<i>A et B</i>
0	0	0	0	0	0
0	1	1	0	1	0
1	0	1	1	0	0
1	1	1	1	1	1

L'ordre de priorité

sur **L'ordre de priorité** : la priorité des opérateurs détermine quels opérateurs doivent être évalués en premier, l'évaluation fait selon l'ordre de priorité suivants:

- La parenthèse plus interne ().
- Les fonctions.
- NOT (!), signe (-).
- *, /, Div, Mod.
- +, -.
- \geq , \leq , $>$, $<$.
- ==, !=.
- AND (&& ET)
- OR (|| OU)