Université 8 mai 45 de Guelma
Faculté des Mathématiques, D'Informatique
et des Sciences de la Matière
Département : MI

Niveau : 1ère Mathématique-Informatique
Matière : Structure Machine 1

Chapitre 2
Les Systèmes de Numération

I. Introduction

La numération universellement utilisée de nos jours est la numération décimale (introduite par les arabe depuis plus de 2000 ans avant Jésus-Christ) cette numération manipule des nombres constitués d'une suite de 10 chiffres (0, 1, 2, 3, 4, 5, 6, 7, 8, 9) et comprenant par fois des symboles (point ou virgule).

Vers la fin des années 30, **Claude Shannon** démontra qu'à l'aide de «contacteurs» (interrupteurs) fermés pour «vrai» et ouverts pour «faux» il était possible d'effectuer des opérations logiques en associant le nombre 1 pour «vrai» et 0 pour «faux». Ce codage de l'information est nommé **base binaire**. C'est avec ce codage que fonctionnent les ordinateurs. Il consiste à utiliser deux états (représentés par les chiffres 0 et 1) pour coder les informations.

II. Définition

Un système de numération est défini par sa base, c.a.d par le nombre de caractères utilisé pour écrire les nombres. Donc pour tout nombre **N** de base **B** et à **n** chiffres on a:

$$(N)_B = \sum_{i=0}^{n-1} a_i B^i$$

où

a_i : est un coefficient avec $0 \le a_i \le B-1$
i : est le rang du chiffre

Exemple :
N = $(2719)_{10}$ = 2000 + 700 + 10 + 9
= $2 . 10^3 + 7. 10^2 + 1. 10^1 + 9. 10^0$
Les coefficients a_i sont les suivants : $a_0 = 9$, $a_1 = 1$, $a_2 = 7$, $a_3 = 2$

III. Quelques systèmes de numération

III.1 Système Décimal

Utilise la base 10 et les chiffres de 0 à 9. Tout nombre dans ce système s'écrit :

$N = a_n 10^n + a_{n-1} 10^{n-1} + \ldots\ldots + a_1 10^1 + a_0 10^0 = (a_n a_{n-1} \ldots\ldots a_1 a_0)_{10}$

Exemple

$(329)_{10} = 3 . 10^2 + 2 . 10^1 + 9 . 10^0$

III.2 Système Binaire

Utilise la base 2 et les chiffres 0 et 1. Tout nombre dans ce système s'écrit :

$N = a_n 2^n + a_{n-1} 2^{n-1} + \ldots\ldots + a_1 2^1 + a_0 2^0 = (a_n a_{n-1} \ldots\ldots a_1 a_0)_2$

Exemple

$(1010)_2 = 1.2^3 + 0.2^2 + 1.2^1 + 0.2^0$

III.3 Système Octal

Utilise la base 8 et les chiffres de 0 à 7. Tout nombre dans ce système s'écrit :

$N = a_n 8^n + a_{n-1}8^{n-1} + \ldots\ldots + a_1 8^1 + a_0 8^0 = (a_n a_{n-1}\ldots\ldots a_1 a_0)_8$

Exemple

$(235)_8 = 2.8^2 + 3.8^1 + 5.8^0$

III.4 Système Hexadécimal

Utilise la base 16 et les chiffres de 0 à 15 mais comme les chiffres de 10 à 15 posent confusion avec les chiffres de 0 à 5 on les remplace par les lettres de A à F successivement. Donc les chiffres sont 0 à 9 et A à F. Tout nombre dans ce système s'écrit :

$N = a_n 16^n + a_{n-1}16^{n-1} + \ldots\ldots + a_1 16^1 + a_0 16^0 = (a_n a_{n-1}\ldots\ldots a_1 a_0)_{16}$

Exemple

$(A29)_{16} = A.(16)^2 + 2.(16)^1 + 9.(16)^0$

IV. Conversions entre systèmes de numération :

IV.1 Conversion d'un nombre de base P en un nombre décimal

La valeur décimale d'un nombre N écrit dans une base P s'obtient par sa forme polynomiale :

$(N)_P = (a_n\ a_{n-1}\ \ldots\ a_0)_P \rightarrow (N)_{10} = a_n P^n + a_{n-1}P^{n-1} + \ldots\ldots + a_0 P^0$

IV.1.1 Conversion Binaire-Décimal

Elle se fait simplement en additionnant les puissances de 2 correspondant aux bits de valeurs 1

Exemple

$(110)_2 = 1.2^2 + 1.2^1 + 0.2^0$
$= 4 + 2$
$= (6)_{10}$

IV.1.2 Conversion Octal-Décimal

La conversion se réduit à une addition de puissances de 8.

Exemple

$(1567)_8 = 1.8^3 + 5.8^2 + 6.8^1 + 7.8^0$
$= 512 + 320 + 48 + 7$
$= (887)_{10}$

IV.1.3 Conversion Hexadécimal-Décimal

La conversion se réduit à une addition de puissances de 16.

Exemple

$(BAC)_H = B.(16)^2 + A.(16)^1 + C.(16)^0$
$= 11.(16)^2 + 10.(16)^1 + 12.(16)^0$
$= (2988)_{10}$

IV.2 Conversion d'un nombre décimal en un nombre de base P

L'algorithme de conversion est le suivant :

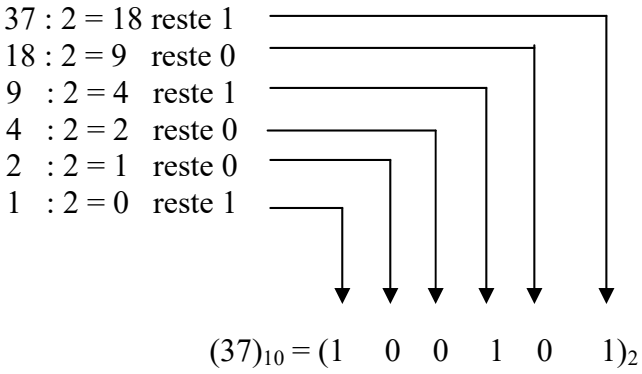
1. Diviser (division entière) le nombre décimal par la base P en premier lieu.
2. Prendre le résultat de la division et le diviser (division entière) de nouveau par la base P.
3. Reprendre l'étape 2 jusqu'à obtenir un résultat nul.
4. Le nouveau nombre équivalent s'obtient en prenant les restes des divisions de la dernière à la première et en les écrivant de gauche à droite successivement.

IV.2.1 Conversion Décimal –Binaire

La conversion s'effectue par des divisions entières successives par 2. Le teste d'arrêt correspond à un quotient nul. Le nombre binaire est obtenu en lisant les restes du dernier au le premier.

Exemple : Convertir le nombre $(37)_{10}$ en binaire.

Solution

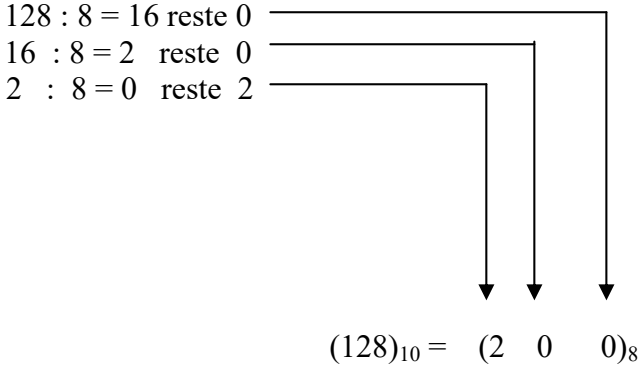


IV.2.2 Conversion Décimal –Octal

La conversion correspond à des divisions entières successives par 8. Le nombre octal est obtenu en prenant les différents restes du dernier au premier.

Exemple : Convertir le nombre $(128)_{10}$ en octal.

Solution



IV.2.3 Conversion Décimal -Hexadécimal

La conversion correspond à des divisions entières successives par 16. Le nombre hexadécimal est obtenu en prenant les différents restes du dernier au premier.

Exemple : Convertir le nombre $(128)_{10}$ en hexadécimal.

128 : 16 = 8 reste 0
8 : 16 = 0 reste 8

$(128)_{10} = (8 \quad 0)_{16}$

IV.3 Conversion des nombres fractionnaires

Nombres fractionnaires

Se sont les nombres qui comportent une partie inférieure à un.

Exemple :
32.125, 2/3, 0.138

IV.3.1 Conversion d'un nombre fractionnaire de base P en un nombre décimal fractionnaire

La valeur décimale d'un nombre N fractionnaire écrit dans une base P s'obtient par sa forme polynomiale avec les rangs négatifs et de façon décroissante.

Exemple : convertir le nombre $(18.053)_{16}$ en décimal

$(18.053)_{16} = 8 . (16)^0 + 1 . (16)^1 + 0 . (16)^{-1} + 5 . (16)^{-2} + 3 . (16)^{-3}$
$= (24 , 0839)_{10}$

IV.3.2 Conversion d'un nombre décimal fractionnaire en un nombre fractionnaire de base P

L'algorithme de conversion consiste à :

1. Multiplier le nombre décimal fractionnaire par la base P en premier lieu.
2. Reprendre la partie fractionnaire du résultat et la multiplier de nouveau par la base.
3. Reprendre l'étape 2 jusqu'à obtenir une partie fractionnaire nulle ou arriver à une limite de chiffres après la virgule.
4. Le nouveau nombre équivalent s'obtiens en prenant les parties entières des résultats des multiplications de la première à la dernière et en les écrivant de gauche à droite.

Exemple : convertir le nombre 0.125 en binaire

Solution :

0.125 x 2 = 0.25 ⟶ 0
0.25 x 2 = 0.5 ⟶ 0
0.5 x 2 = 1.00 ⟶ 1

$(0.125)_{10} = (0.001)_2$

Remarque

Si on veut convertir un nombre constitué d'une partie entière et d'une partie fractionnaire, on convertie la partie entière à part et la partie fractionnaire à part.

Exemple : convertir du décimal au binaire le nombre 15.625

Solution :

15 : 2 = 7 reste 1
7 : 2 = 3 reste 1
3 : 2 = 1 reste 1
1 : 2 = 0 reste 1

$\Longrightarrow$ $(15)_{10} = (1111)_2$

0.625 x 2 = 1.25 $\longrightarrow$ 1
0.25 x 2 = 0.5 $\longrightarrow$ 0
0.5 x 2 = 1.00 $\longrightarrow$ 1

$(0.625)_{10} = (0.101)_2$

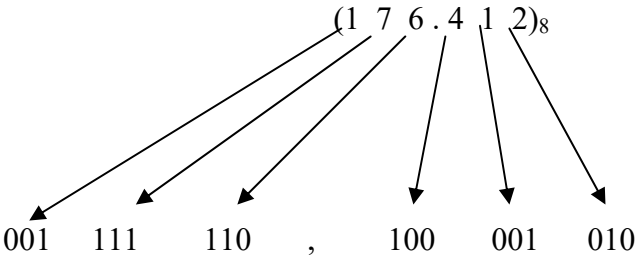
Donc :

$(15{,}625)_{10} = (1111{,}101)_2$

IV.3.3 Conversion Binaire – octal et vice versa

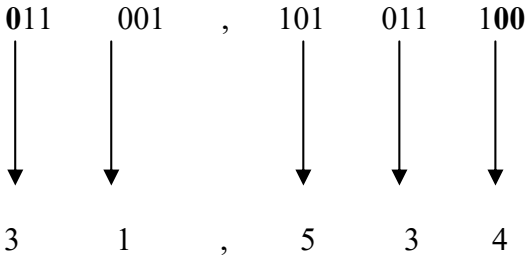
Les chiffres du système octal sont de 0 à 7, ces chiffres nécessitent 3 bits pour être codés en binaire 0 $(000)_2$ et 7 $(111)_2$.
Donc pour convertir un nombre octal en binaire il suffi de remplacer chaque chiffre par son équivalent binaire sur 3 bits.

Exemple : Convertir le nombre suivant de l'octal au binaire : $(176.412)_8$



De façon inverse, pour convertir un nombre binaire en octal on regroupe les bits (chiffre binaires) en groupe de trois de droite à gauche pour le partie entière (on ajoute des zéros à gauche si c'est nécessaire) et de gauche à droite pour la partie fractionnaire (on ajoute des zéros à droite si c'est nécessaire) en suite on remplace chaque groupe par son équivalent en octal.

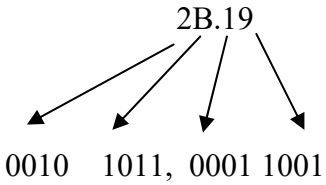
Exemple : Convertir le nombre suivant en octal : $(11001.1010111)_2$



IV.3.4 Conversion Binaire – Hexadécimal et vise versa

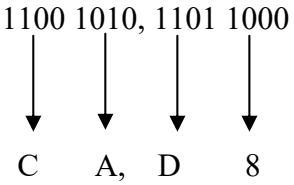
Les chiffres du système décimal sont de 0 à 9 et de A à F, ces chiffres nécessite 4 bits pour être codés en binaire : 0 $(0000)_2$ et F$(1111)_2$.
Ainsi pour convertir un nombre hexadécimal en binaire il suffit de remplacer chaque chiffre par son équivalent binaire sur 4 bits.

Exemple : Convertir le nombre suivant en binaire : $(2B.19)_{16}$
Solution :



De façon inverse pour convertir un nombre hexadécimal, on regroupe les bits en groupe de quatre de droite à gauche pour la partie entière (on ajoute des zéros à gauche si c'est nécessaire) et de gauche à droite pour la partie fractionnaire (on ajoute des zéros à droite si c'est nécessaire) en suite on remplace chaque groupe par son équivalent hexadécimal.

Exemple : Convertir le nombre suivant en Hexadécimal : $(1100\ 1010, 1101\ 1000)_2$
Solution :



V. Opérations arithmétiques de base

V.1 Opération arithmétique en binaire

Pourquoi l'arithmétique binaire ?

Un ordinateur (ou micro-ordinateur) ne comprend que le code binaire (Hexadécimal, Octal) donc au lieu d'effectuer l'arithmétique habituelle l'ordinateur effectue l'arithmétique binaire.

a. Addition binaire :

Les règles de base de l'addition binaire sont :

0 + 0 = 0
0 + 1 = 1
1 + 0 = 1
1 + 1 = 0 retenue 1
1 + 1 + 1 = 1 retenue 1

Exemple : effectuer l'opération d'addition suivante en binaire

$(18)_{10}$ ⟶ $(10010)_2$
$+\ (20)_{10}$ ⟶ $+\ (10100)_2$
$= (38)_{10}$ $(100110)_2$

b. Soustraction binaire :

Les règles de base de la soustraction binaire sont :

0 - 0 = 0
1 - 0 = 1
1 - 1 = 0
0 - 1 = 1 retenue 1
1 – 0 – 1 = 0
1– 1 – 1 = 1 retenue 1
0 – 1 – 1 = 0 retenue 1

Exemple : Effectuer l'opération de soustraction suivante en binaire

$(20)_{10}$ ⟶ $(10100)_2$
$-\ (18)_{10}$ ⟶ $-\ (10010)_2$
$=(02)_{10}$ $(00010)_2$

c. Multiplication binaire :

Les règles de base de la multiplication binaire sont :

0 x 0 = 0
0 x 1 = 0
1 x 0 = 0
1 x 1 = 1

Exemple : effectuer l'opération de multiplication suivante en binaire

11 ⟶ 1011
x 5 ⟶ x 101
55 1011
\+ 0000 .
1011 . .
110111

d. Division binaire :

Les règles de base de la division binaire sont :
0 / 1 = 0
1 / 1 = 1

Exemple : effectuer l'opération de division suivante en binaire

175	5
25	35
0	

➔

10101111	101
111	100011
101	
0101	
0	

Remarque :
Les opérations arithmétiques en octal, en hexadécimal et dans d'autres bases seront étudier dans des exercices de travaux dirigés.