

Institut Préparatoire aux Études d'Ingénieurs de Tunis



Leila Ben Othman

Première année des classes préparatoires
MP-PC

Année universitaire: 2017- 2018

Représentation des nombres

Plan

Partie 1:

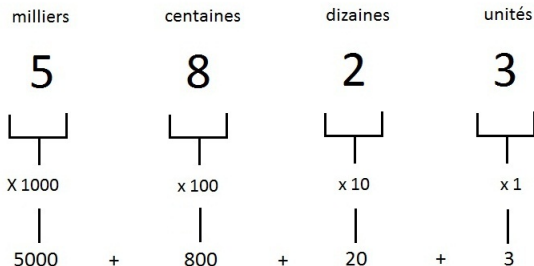
- ① Décomposition d'un nombre décimal
- ② Système de numération (décimal, binaire, octale, hexadécimal)
- ③ Transcodage
 - Conversion de la base 10 en base b
 - Méthode des divisions successives
 - Conversion d'une base b vers le décimale
 - Méthode naïve
 - Méthode de Horner
 - Conversion du binaire vers l'octal et inversement
 - Conversion du binaire vers l'hexadécimal et inversement
 - Conversion de l'octal vers hexadécimal et inversement

Plan

Partie : 2

Décomposition d'un nombre décimal (1)

décomposition du nombre décimal 5 823



$$5823 = 5 \times 10^3 + 8 \times 10^2 + 2 \times 10^1 + 3 \times 10^0$$

Décomposition d'un nombre décimal (1)

décomposition du nombre décimal 5 823

milliers

5

X 1000

1

5000

+

centaines

8

x 100

1

800

+

dizaines

2

x 10

1

20

+

unités

3

x 1

1

3

$$5823 = 5 \times 10^3 + 8 \times 10^2 + 2 \times 10^1 + 3 \times 10^0$$

Décomposition d'un nombre décimal (2)

De façon générale, le nombre N s'écrit sous la forme d'une suite de chiffres (juxtaposition de chiffres)

$$N = a_{n-1}a_{n-2}\dots a_1a_0$$

où :

- n désigne le nombre de chiffres de N
- a_i représente un chiffre composant le nombre N
- i représente la position du chiffre dans l'écriture de N , où $i \in \{0, 1, \dots, n-1\}$
 - a_0 est dit chiffre de poids faible.
 - a_{n-1} est dit chiffre de poids fort ($a_{n-1} \neq 0$).

Décomposition d'un nombre décimal (2)

De façon générale, le nombre N s'écrit sous la forme d'une suite de chiffres (juxtaposition de chiffres)

$$N = a_{n-1}a_{n-2}\dots a_1a_0$$

où:

- n désigne le nombre de chiffres de N
- a_i représente un chiffre composant le nombre N
- i représente la position du chiffre dans l'écriture de N , où $i \in \{0, 1, \dots, n-1\}$
 - a_0 est dit chiffre de poids faible.
 - a_{n-1} est dit chiffre de poids fort ($a_{n-1} \neq 0$).

Décomposition d'un nombre décimal (3)

N=5823

n=4

5	8	2	3
a_3 chiffre de poids fort	a_2	a_1	a_0 chiffre de poids faible

Décomposition d'un nombre décimal (4)

La valeur du nombre N est donnée par:

- $N = a_{n-1} \times 10^{n-1} + a_{n-2} \times 10^{n-2} + \dots + a_0 \times 10^0$

- ou encore $N = \sum_{i=0}^{i=n-1} a_i 10^i$

$\Rightarrow$ On parle d'un **système de numération décimal**, autrement, un **système en base 10**.

Système de numération décimal

Pour représenter le nombre N en base 10, on utilise:

- Dix chiffres (symboles): 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- Une base 10 qui sert à multiplier les chiffres selon leurs positions, afin de connaître la valeur de N

$$5823 = 5 \times 10^3 + 8 \times 10^2 + 2 \times 10^1 + 3 \times 10^0$$

Système de numération décimal

Pour représenter le nombre N en base 10, on utilise:

- Dix chiffres (symboles): 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- Une base 10 qui sert à multiplier les chiffres selon leurs positions, afin de connaître la valeur de N

$$5823 = 5 \times 10^3 + 8 \times 10^2 + 2 \times 10^1 + 3 \times 10^0$$

Qu'est-ce qu'un système de numération?

- Un **système de numération** décrit la façon avec laquelle les nombres sont représentés
- Un système de numération est défini par:
 - un alphabet: ensemble de **symboles** ou de **chiffres**
 - une **base b** qui correspond au nombre de symboles que possède le système et qu'on utilise pour calculer la valeur d'un nombre.
- **Notation:** On notera par $(N)_b$, le nombre N représenté en base b .
 - Par exemple, la notation $(5823)_{10}$ indique que le nombre 5823 est en base 10.

Qu'est-ce qu'un système de numération?

- Un **système de numération** décrit la façon avec laquelle les nombres sont représentés
- Un système de numération est défini par:
 - un alphabet: ensemble de **symboles** ou de **chiffres**
 - une **base b** qui correspond au nombre de symboles que possède le système et qu'on utilise pour calculer la valeur d'un nombre.
- **Notation:** On notera par $(N)_b$, le nombre N représenté en base b .
 - Par exemple, la notation $(5823)_{10}$ indique que le nombre 5823 est en base 10.

Système de numération en base b

La valeur du nombre $(N)_b = a_{n-1}a_{n-2}\dots a_1a_0$ est donnée par:

- $N = a_{n-1} \times b^{n-1} + a_{n-2} \times b^{n-2} + \dots + a_0 \times b^0$

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

$\Rightarrow$ On parle d'un **système de numération en base b**

Application

$$(10001)_2 =$$

$$(122)_3 =$$

$$(101)_4 =$$

$$(32)_5 =$$

$$(25)_6 =$$

$$(23)_7 =$$

$$(21)_8 =$$

$$(18)_9 =$$

Application

$$(10001)_2 = 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = (17)_{10}$$

$$(122)_3 = 1 \times 3^2 + 2 \times 3^1 + 2 \times 3^0 = (17)_{10}$$

$$(101)_4 = 1 \times 4^2 + 0 \times 4^1 + 1 \times 4^0 = (17)_{10}$$

$$(32)_5 = 3 \times 5^1 + 2 \times 5^0 = (17)_{10}$$

$$(25)_6 = 2 \times 6^1 + 5 \times 6^0 = (17)_{10}$$

$$(23)_7 = 2 \times 7^1 + 3 \times 7^0 = (17)_{10}$$

$$(21)_8 = 2 \times 8^1 + 1 \times 8^0 = (17)_{10}$$

$$(18)_9 = 1 \times 9^1 + 8 \times 9^0 = (17)_{10}$$

Système de numération

- Système de numération binaire
⇒ Base 2, deux chiffres $\{0, 1\}$
- Système de numération octale
⇒ Base 8, huit chiffres $\{0, 1, 2, 3, 4, 5, 6, 7\}$
- Système de numération hexadécimale
⇒ Base 16, seize chiffres $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$
avec $A=10$, $B=11$, $C=12$, $D=13$, $E=14$, $F=15$ en base 10

Méthode des divisions successives

- 1 Faire une suite de divisions euclidiennes par la base b . Le chiffre de poids faible a_0 est égal au reste r_0 de la division entière de N par b .
- 2 Pour trouver les chiffres suivants $a_1, a_2, \dots, a_{n-1}$, on divise le quotient obtenu par la base b et ainsi de suite, jusqu'à ce que le quotient soit égal à zéro.
- 3 Le résultat sera la juxtaposition des restes (mis en gras dans l'exemple ci-dessous).

$$\begin{aligned}
 (N)_{10} &= r_0 + (b \times q_0) \\
 &= r_0 + (b \times (r_1 + (b \times q_1))) \\
 &= r_0 + (b \times (r_1 + (b \times (r_2 + (b \times q_2))))) \\
 &= \mathbf{r_0} + (b \times (\mathbf{r_1} + (b \times (\mathbf{r_2} + b \times \dots (\mathbf{r_{n-1}} + (b \times q_{n-1}))))))
 \end{aligned}$$

Au final, $(N)_{10} = r_{n-1}r_{n-2}\dots r_1r_0$

$$\begin{aligned}
 &= a_{n-1}a_{n-2}\dots a_1a_0 \\
 &= (N')_b
 \end{aligned}$$

Méthode des divisions successives

- 1 Faire une suite de divisions euclidiennes par la base b . Le chiffre de poids faible a_0 est égal au reste r_0 de la division entière de N par b .
- 2 Pour trouver les chiffres suivants $a_1, a_2, \dots, a_{n-1}$, on divise le quotient obtenu par la base b et ainsi de suite, jusqu'à ce que le quotient soit égal à zéro.
- 3 Le résultat sera la juxtaposition des restes (mis en gras dans l'exemple ci-dessous).

$$\begin{aligned}
 (N)_{10} &= r_0 + (b \times q_0) \\
 &= r_0 + (b \times (r_1 + (b \times q_1))) \\
 &= r_0 + (b \times (r_1 + (b \times (r_2 + (b \times q_2))))) \\
 &= \mathbf{r_0} + (b \times (\mathbf{r_1} + (b \times (\mathbf{r_2} + b \times \dots (\mathbf{r_{n-1}} + (b \times q_{n-1}))))))
 \end{aligned}$$

$$\begin{aligned}
 \text{Au final, } (N)_{10} &= r_{n-1}r_{n-2}\dots r_1r_0 \\
 &= a_{n-1}a_{n-2}\dots a_1a_0 \\
 &= (N')_b
 \end{aligned}$$

Méthode des divisions successives

Exemple : $(681)_{10} = (?)_8$

$$\begin{aligned}(681)_{10} &= 1 + (8 \times 85) \\ &= 1 + (8 \times (5 + (8 \times 10))) \\ &= 1 + (8 \times (5 + (8 \times (2 + (8 \times 1))))))\end{aligned}$$

$$= 1 + (8 \times (5 + (8 \times (2 + (8 \times (1 + (8 \times 0))))))))$$

Finalement, on aura $681_{(10)} = 1251_{(8)}$

Méthode des divisions successives

Exemple : $(681)_{10} = (?)_8$

$$\begin{aligned}(681)_{10} &= 1 + (8 \times 85) \\ &= 1 + (8 \times (5 + (8 \times 10))) \\ &= 1 + (8 \times (5 + (8 \times (2 + (8 \times 1)))))\end{aligned}$$

$$= \mathbf{1} + (8 \times (\mathbf{5} + (8 \times (\mathbf{2} + (8 \times (\mathbf{1} + (8 \times 0)))))))$$

Finalement, on aura $681_{(10)} = 1251_{(8)}$

Méthode des divisions successives

Exemple : $(25)_{10} = (?)_2$

Exemple : $(44)_{10} = (?)_{16}$

$$(25)_{10} = (11001)_2$$

$$(44)_{10} = (2C)_{16}$$

Méthode des divisions successives

Exemple : $(25)_{10} = (?)_2$

Exemple : $(44)_{10} = (?)_{16}$

$$(25)_{10} = (11001)_2$$

$$(44)_{10} = (2C)_{16}$$

Méthode des divisions successives

Algorithme 1 : Conversion d'un nombre N de la base 10 en base b

Entrées : Un entier $N > 0$ et un entier $b > 2$.

Sortie : L'écriture de N dans la base b .

$$i \leftarrow 0;$$
$$M \leftarrow N;$$

tant que $M \neq 0$ faire

$(q, r) \leftarrow$ quotient et reste de la division euclidienne de M par b ;

$$a_i \leftarrow r;$$
$$M \leftarrow q;$$
$$i \leftarrow i + 1;$$

retourner $(a_{i-1}, a_{i-2}, \dots, a_0)$

Méthode naïve

- Cette première méthode naïve consiste à appliquer la formule de conversion suivante :

$$(N)_b = \sum_{i=0}^{i=n-1} a_i b^i = (N')_{10}.$$

Exemple : $(4567)_8 = (?)_{10}$

$$(4567)_8 = 4 \times 8^3 + 5 \times 8^2 + 6 \times 8^1 + 7 \times 8^0 = (2423)_{10}$$

Méthode de Horner

- La deuxième méthode consiste à appliquer la formule de *Horner* :

$$N_{(b)} = a_0 + b(a_1 + b(\dots + b(a_{n-2} + ba_{n-1}))) = N'_{(10)}$$

Exemple : $(4567)_8 = (?)_{10}$

$$\begin{aligned}(4567)_8 &= 7 + (8 \times (6 + 8 \times (5 + (8 \times 4)))) \\ &= 7 + (8 \times (6 + 8 \times (5 + 32))) \\ &= 7 + (8 \times (6 + (8 \times 37))) \\ &= 7 + (8 \times (6 + 296)) \\ &= 7 + (8 \times 302) \\ &= 7 + 2416 \\ &= (2423)_{10}\end{aligned}$$

Méthode de Horner

- La deuxième méthode consiste à appliquer la formule de *Horner* :

$$N_{(b)} = a_0 + b(a_1 + b(\dots + b(a_{n-2} + ba_{n-1}))) = N'_{(10)}$$

Exemple : $(4567)_8 = (?)_{10}$

$$\begin{aligned}(4567)_8 &= 7 + (8 \times (6 + 8 \times (5 + (8 \times 4)))) \\ &= 7 + (8 \times (6 + 8 \times (5 + 32))) \\ &= 7 + (8 \times (6 + (8 \times 37))) \\ &= 7 + (8 \times (6 + 296)) \\ &= 7 + (8 \times 302) \\ &= 7 + 2416 \\ &= (2423)_{10}\end{aligned}$$

Méthode de Horner

Algorithme 2 : Algorithme de Horner

Entrées : Un entier $b \geq 2$ et un entier N donné par la liste de ses chiffres dans la base b

Sortie : L'évaluation de N dans la base ambiante

$M \leftarrow a_{n-1};$

pour chaque i allant de $n-2$ à 0 par pas de -1 **faire**

└ $M \leftarrow b \times M + a_i;$

retourner M

Binaire vers octal

Un chiffre en octal est représenté en binaire moyennant 3 digits ($8 = 2^3$)

Symbole octal	Binaire
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Binaire vers octal

- Parcourir le nombre binaire de la droite vers la gauche en regroupant les chiffres binaire par paquet de trois (ajouter éventuellement des 0 à gauche)
- Remplacer chaque paquet par le symbole correspondant dans la base 8.

Exemple : $(0101011011011110)_2 = (?)_8$

0	0	0	1	0	1	0	1	1	0	1	1	0	1	1	1	1	0
0			5			3			3			3			6		

Finalement, on aura $(0101011011011110)_2 = (53336)_8$

Binaire vers octal

- Parcourir le nombre binaire de la droite vers la gauche en regroupant les chiffres binaire par paquet de trois (ajouter éventuellement des 0 à gauche)
- Remplacer chaque paquet par le symbole correspondant dans la base 8.

Exemple : $(0101011011011110)_2 = (?)_8$

0	0	0	1	0	1	0	1	1	0	1	1	0	1	1	1	1	0
0			5			3			3			3			6		

Finalement, on aura $(0101011011011110)_2 = (53336)_8$

Octal vers Binaire

- Parcourir le nombre octal de la droite vers la gauche
- Remplacer chaque symbole par le symbole (3 digits) correspondant dans la base binaire.

Example : $(64)_8 = (?)_2$

6	4
110	100

Octal vers Binaire

- Parcourir le nombre octal de la droite vers la gauche
- Remplacer chaque symbole par le symbole (3 digits) correspondant dans la base binaire.

Exemple : $(64)_8 = (?)_2$

6	4
110	100

on aura $(64)_8 = (110100)_2$

Binaire vers hexadécimal

Un chiffre en hexadécimal est représenté en binaire moyennant 4 digits

Symbole hexadécimal	Binaire
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

Binaire vers hexadécimal

- Parcourir le nombre binaire de la droite vers la gauche en regroupant les chiffres binaire par paquet de quatre (ajouter éventuellement des 0 à gauche)
- Remplacer chaque paquet par le symbole correspondant dans la base 16.

Exemple : $(0101011011011110)_2 = (?)_{16}$

0	1	0	1	0	1	1	0	1	1	0	1	1	1	1	0
5				6				D				E			

Finalement, on aura $(0101011011011110)_2 = (56DE)_{16}$

Binaire vers hexadécimal

- Parcourir le nombre binaire de la droite vers la gauche en regroupant les chiffres binaire par paquet de quatre (ajouter éventuellement des 0 à gauche)
- Remplacer chaque paquet par le symbole correspondant dans la base 16.

Exemple : $(0101011011011110)_2 = (?)_{16}$

0	1	0	1	0	1	1	0	1	1	0	1	1	1	1	0
5				6				D				E			

Finalement, on aura $(0101011011011110)_2 = (56DE)_{16}$

Hexadécimal vers binaire

- Parcourir le nombre hexadécimal de la droite vers la gauche
- Remplacer chaque symbole par le symbole (4 digits) correspondant dans la base binaire.

Exemple : $(BA2)_{16} = (?)_2$

B	A	2
1011	1010	0010

Hexadécimal vers binaire

- Parcourir le nombre hexadécimal de la droite vers la gauche
- Remplacer chaque symbole par le symbole (4 digits) correspondant dans la base binaire.

Exemple : $(BA2)_{16} = (?)_2$

B	A	2
1011	1010	0010

on aura $(BA2)_{16} = (101110100010)_2$

Octal vers hexadécimal et inversement

- Passer par le système binaire
- ou bien par le système décimal

Exemple

- $(A)_{16} = (?)_8$
- $(71)_8 = (?)_{16}$

Réponse: On va passer par le système décimal.

- $(A)_{16} = (10)_{10}$. Par la méthode des divisions successives, on obtient $(10)_{10} = (12)_8$, donc $(A)_{16} = (10)_{10} = (12)_8$.

- $(71)_8 = (?)_{10}$. Il suffit d'appliquer la formule $\sum_{i=0}^{i=n-1} a_i b^i$, on retrouve

le nombre en base 10: $(71)_8 = 7 \times 8^1 + 1 \times 8^0 = (57)_{10}$. Par la méthode des divisions successives, on obtient

$$(71)_8 = (57)_{10} = (39)_{16}$$

Addition binaire (1)

- **Table d'addition binaire**

a	b	somme	retenue
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Addition binaire (2)

- Exemple 1

$$\begin{array}{r}
 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 0\ 1\ 1 \\
 +\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 1\ 0\ 1 \\
 \hline
 s=
 \end{array}$$

- Exemple 2

$$\begin{array}{r}
 1\ 0\ 1\ 0\ 1\ 1\ 0\ 1 \\
 +\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 0 \\
 \hline
 s=
 \end{array}$$

Addition binaire (3)

- Exemple 1

										1	1	1	1	1	1	1	1
r																	
	1	0	1	0	1	0	1	0	1	0	1	0	0	1	0	0	1
+	0	1	0	0	0	1	0	0	0	0	1	1	0	1	1	0	1
s=	1	1	1	0	1	1	1	1	0	0	0	0	0	0	0	0	0

- Exemple 2 Attention aux retenues!

r					1	1	1
	1	0	1	0	1	1	0
							1
+	0	0	1	0	1	1	0
							0
s=	1	1	0	1	1	0	1

Soustraction binaire (1)

- **Table de soustraction binaire**

a	b	diff	emprunt
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Soustraction binaire (2)

- **Example 1**

$$\begin{array}{r} 1 \quad 1 \quad 1 \quad 1 \\ - \quad \quad 1 \quad 1 \quad 0 \\ \hline \equiv \end{array}$$

- **Example 2**

$$\begin{array}{cccc} & 1 & 0 & 1 & 0 \\ - & & 1 & 0 & 1 \\ \hline = & & & & \end{array}$$

Soustraction binaire (3)

- Exemple 1

$$\begin{array}{r} 1 \ 1 \ 1 \ 1 \\ - \quad 1 \ 1 \ 0 \\ \hline = 1 \ 0 \ 0 \ 1 \end{array}$$

- Exemple 2

$$\begin{array}{r} 1 \ 0 \ 1 \ 0 \\ - \quad 1 \ 0 \ 1 \\ \hline = 0 \ 1 \ 0 \ 1 \end{array}$$

Multiplication binaire (1)

- **Table de multiplication binaire**

a	b	p
0	0	0
0	1	0
1	0	0
1	1	1

Multiplication binaire (2)

- Exemple

$$\begin{array}{r}
 1011 \\
 \times 1001 \\
 \hline
 1011 \\
 0000 \\
 0000 \\
 1011 \\
 \hline
 1100011
 \end{array}$$

ou directement

$$\begin{array}{r}
 1011 \\
 \times 1001 \\
 \hline
 1011 \\
 1011 \\
 \hline
 1100011
 \end{array}$$

Division binaire (1)

- **Example**

$$\begin{array}{r|rrrrrr}
 1 & 0 & 0 & 1 & 0 & 1 & 0 & & 1 & 0 & 1 \\
 \hline
 0 & 0 & 0 & \downarrow & | & | & | & 0 & 1 & 1 & 1 & 0 \\
 \hline
 1 & 0 & 0 & 1 & | & | & | & & & & & \\
 & 1 & 0 & 1 & \downarrow & | & | & & & & & \\
 \hline
 0 & 1 & 0 & 0 & 0 & | & | & & & & & \\
 & 1 & 0 & 1 & \downarrow & | & & & & & & \\
 \hline
 0 & 0 & 0 & 1 & 1 & 1 & | & & & & & \\
 & 1 & 0 & 1 & \downarrow & & & & & & & \\
 \hline
 & 0 & 1 & 0 & 0 & & & & & & &
 \end{array}$$

Récapitulons - Représentation des entiers naturels

Représentation
sur 8 bits

00000000
00000001
11111111

256 nombres
variant de 0 à
255

0
1
255

De façon, générale, avec n
bits, on a 2^n nombres
variant de 0 à 2^n-1

Représentation des entiers naturels

⇒ Comment représenter les entiers relatifs?

Signe et grandeur (1)

Principe:

- Réserver le bit le plus fort pour indiquer le signe de l'entier
- Utiliser les autres bits pour représenter sa valeur absolue
 - Le bit le plus fort=0 $\Rightarrow$ le nombre est positif
 - Le bit le plus fort=1 $\Rightarrow$ le nombre est négatif

Considérons un nombre sur 8 bits:

$$\left| a_7 \right| \left| a_6 \right| \left| a_5 \right| \left| a_4 \right| \left| a_3 \right| \left| a_2 \right| \left| a_1 \right| \left| a_0 \right|$$

- a_7 sera utilisé pour représenter le signe
- Les autres bits ($a_6 \dots a_0$) seront utilisés pour représenter la valeur absolue du nombre.

Signe et grandeur (2)

Exemple

- $(+8)_{10} = (?)_2$
- $(-8)_{10} = (?)_2$

	0		0		0		0		1		0		0		0	
	1		0		0		0		1		0		0		0	

- $(+8)_{10} = (00001000)_2$
- $(-8)_{10} = (10001000)_2$

Signe et grandeur (3)

Représentation sur 8 bits 256 nombres variant de -127 à +127

00000000	+0
00000001	+1
01111111	+127
10000000	-0
10000001	-1
11111111	-127

De façon, générale, avec n bits, on a 2^n nombres variant de $-(2^{n-1}-1)$ à $2^{n-1}-1$

Attention : l'existence de deux zéros, l'un positif et l'autre négatif

Représentation des entiers relatifs avec signe et grandeur

Représentation en Complément à 2

Représentation en Complément 2: deux méthodes

- Méthode 1: par ajout de 2^n
- Méthode 2: par inversion de bits et ajout de 1

Représentation en Complément à 2

Méthode 1: par ajout de 2^n

- Un entier relatif x positif ou nul est représenté comme l'entier naturel x
- Un entier relatif x strictement négatif est représenté comme l'entier naturel p tel que $p = x + 2^n$.

Représentation en Complément à 2

Méthode 1: par ajout de 2^n

- Un entier relatif x positif ou nul est représenté comme l'entier naturel x
- Un entier relatif x strictement négatif est représenté comme l'entier naturel p tel que $p = x + 2^n$.

Exemple:

- Donner la représentation en complément 2 sur 8 bits, des entiers relatifs suivants: 56 et -12
 - 56 est strictement positif. $(56)_{10} = (00111000)_2$, Donc 00111000 est la représentation en complément 2 de 56.
 - -12 est strictement négatif donc est représenté par $-12+256=244$. $(244)_{10} = (11110100)_2$. Donc 11110100 est la représentation en complément 2 de -12.

Représentation en Complément à 2

Méthode 1: par ajout de 2^n

- Un entier relatif x positif ou nul est représenté comme l'entier naturel x
- Un entier relatif x strictement négatif est représenté comme l'entier naturel p tel que $p = x + 2^n$.

Exemple:

- Donner la représentation en complément 2 sur 8 bits, des entiers relatifs suivants: 56 et -12
 - 56 est strictement positif. $(56)_{10} = (00111000)_2$, Donc 00111000 est la représentation en complément 2 de 56.
 - -12 est strictement négatif donc est représenté par $-12+256=244$. $(244)_{10} = (11110100)_2$. Donc 11110100 est la représentation en complément 2 de -12.

Représentation en Complément à 2

Méthode 2: par inversion de bits et ajout de 1

- Un entier relatif x positif ou nul est représenté comme l'entier naturel x
- Un entier relatif x strictement négatif est représenté de la manière suivante:
 - ① représenter la valeur absolue de x
 - ② remplacer chaque 0 par 1 et chaque 1 par 0
NB: à ce niveau, vous avez une représentation en complément à 1, noté $(N)_{C1}$.
 - ③ ajouter 1

Exemple: Donner la représentation en complément 2 sur 8 bits, de -12 par la méthode 2.

$(12)_{10} = (00001100)_2$. On inverse, on a donc 11110011. On ajoute 1. On obtient 11110100.

⇒ **Vérifier votre résultat avec la méthode précédente!**

Représentation en Complément à 2

Représentation
sur 8 bits

256 nombres
variant de -128
à +127

00000000	+0
00000001	+1
01111111	+127
10000000	-128
10000001	-127
11111111	-1

De façon, générale, avec n
bits, on a 2^n nombres
variant de -2^{n-1} à $2^{n-1}-1$

Représentation des entiers relatifs en complément 2

Représentation en Complément à 2

Question: En utilisant la représentation en complément 2, quel est l'entier relatif représenté par 00010111? de même pour 10001100?

⇒ **Comment retrouver l'entier relatif donné en binaire sur n bits?**

⇒ 2 ou 3 techniques.

Représentation en Complément à 2

Technique 1:

- Calculer p l'entier naturel représenté par le nombre binaire donné
- Si p est inférieur ou égal à $2^{n-1} - 1$, alors p est l'entier relatif représenté.
- S'il est strictement supérieur à $2^{n-1} - 1$, l'entier relatif représenté est $p - 2^n$
- Exemple: Donner les représentations décimales des entiers relatifs dont les représentations en complément 2 sur 8 bits sont: 00010111 et 10001100.
- 00010111 représente l'entier naturel 23 et donc l'entier relatif 23.
- 10001100 représente l'entier naturel 140 et donc l'entier relatif $140 - 256 = -116$.

Représentation en Complément à 2

Technique 1:

- Calculer p l'entier naturel représenté par le nombre binaire donné
- Si p est inférieur ou égal à $2^{n-1} - 1$, alors p est l'entier relatif représenté.
- S'il est strictement supérieur à $2^{n-1} - 1$, l'entier relatif représenté est $p - 2^n$
- Exemple: Donner les représentations décimales des entiers relatifs dont les représentations en complément 2 sur 8 bits sont: 00010111 et 10001100.
- 00010111 représente l'entier naturel 23 et donc l'entier relatif 23.
- 10001100 représente l'entier naturel 140 et donc l'entier relatif $140 - 256 = -116$.

Représentation en Complément à 2

Technique 2 :

- Calculer p l'entier naturel représenté par le nombre binaire donné
- Si le bit du poids fort du nombre binaire est gale à 0, alors p est l'entier relatif représenté.
- Sinon, l'entier relatif représenté est $p - 2^n$

Représentation en Complément à 2

Technique 3 :

- La valeur de x est donnée directement par la formule suivante:

$$x = -a_{n-1}2^{n-1} + \sum_{i=0}^{i=n-2} a_i 2^i$$

Intérêt du Complément à 2

- Le signe d'un entier se reconnaît par le bit de poids fort (0 pour un nombre positif, 1 pour un nombre négatif);
- L'addition des entiers relatifs en complément à 2 utilise le même principe que pour les entiers naturels.
- La soustraction des entiers relatifs en complément à 2 revient à effectuer une addition : $A - B \Rightarrow A + (B)_{c2}$

Addition/Soustraction en Complément à 2

- addition de 72 et 59 sur 8 bits.

$$(72)_{10} = (1001000)_{c2}.$$

$$(59)_{10} = (111011)_{c2}.$$

$$\begin{array}{cccccccc} & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ + & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ \hline \equiv & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{array}$$

⇒ Qu'est-ce que vous remarquez?

Addition/Soustraction en Complément à 2

- addition de (-4) et (-5) sur 4 bits.

$$(-4)_{10} = (1100)_{c2}.$$

$$(-5)_{10} = (1011)_{c2}.$$

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

⇒ Qu'est-ce que vous remarquez?

Addition/Soustraction en Complément à 2

- addition de (3) et (-1) sur 4 bits.

$$(3)_{10} = (0011)_{c2}.$$

$$(-1)_{10} = (1111)_{c2}.$$

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

⇒ Qu'est-ce que vous remarquez?

Addition/Soustraction en Complément à 2

- Quand le résultat d'une opération n'appartient pas à l'intervalle $[-2^{n-1}, 2^{n-1}-1]$. On parle de dépassement de capacité mémoire.
- Quand on travaille sur une représentation à n bits et que le résultat d'une opération s'écrit sur $n + 1$ bits, on parle de débordement de bit, qu'il faudra ignorer.

Addition en Complément à 2	Le résultat s'écrit sur	Dépassement mémoire
deux positifs	n bits	Possible
deux négatifs	$n + 1$ bits	Possible
signe opposé	n ou $n + 1$ bits	Impossible

Représentation des nombres réels

Deux représentations:

- Représentation en virgule fixe
- Représentation en virgule flottante

Représentation en virgule fixe

Comment représenter $(125, 50)_{10}$ en binaire?

$$(125.50)_{10} = 64 + 32 + 16 + 8 + 4 + 1 + 0.5 = 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^0 + 2^{-1}$$

$$\Rightarrow \text{Donc } (125.50)_{10} = (1111101.1)_2$$

Représentation en virgule fixe

Du décimal au binaire: deux étapes en considérant $(N)_{10} = PE, PF$

- 1 Représenter la partie entière PE en binaire $\rightarrow$ divisions successives par 2.
- 2 Représenter la partie fractionnaire PF en binaire $\rightarrow$ multiplication de PF par 2 jusqu'à ce que PF soit nulle ou qu'on atteigne un nombre de bits défini.

Exemple:

- $(5,625)_{10} = (?)_2$
- $PE = (5)_{10} = (101)_2$
- $PF = 0,625 \times 2 = 1,250$
- $0,250 \times 2 = 0,5$
- $0,50 \times 2 = 1,0$
- donc $0,625 = 0,101$
- Au final, $(5,625)_{10} = (101,101)_2$

Représentation en virgule fixe

Du binaire au décimal:

En considérant N écrit en base 2, tel que

$$(N)_2 = a_{n-1}a_{n-2}\dots a_1a_0, a_{-1}, a_{-2}\dots a_{-p}$$

avec

- n le nombre de chiffres avant la virgule
- p le nombre de chiffres après la virgule

La valeur de N en base 10 est donnée par:

$$(N)_{10} = \sum_{i=-p}^{i=n-1} a_i 2^i$$

Exemple:

$$(1111101.1)_2 = 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^0 + 2^{-1} = (125.5)_{10}$$

Représentation en virgule flottante

$$N = \text{signe} \times m \times B^e$$

avec

- *signe* : + (bit 0) ou - (bit 1)
- *m*: mantisse (comportant une partie entière et une partie fractionnaire)
- *B*: base
- *e*: exposant

Mantisse normalisée: En base 2, on a $1 \leq |m| < 2$ de la forme $1, M$.
On ne représente pas le 1 $\Rightarrow$ économiser un bit.

L'exposant *e* est codé selon la technique du décalage: représenté en machine par l'entier positif $e' = e + 2^{n-1} - 1$

Représentation en virgule flottante

Standard IEEE 754

- Simple précision sur 32 bits
 - 1 bit pour le signe
 - 8 bits pour l'exposant
 - 23 bits pour la mantisse
- Double précision sur 64 bits
 - 1 bit pour le signe
 - 11 bits pour l'exposant
 - 52 bits pour la mantisse

Représentation en virgule flottante



- Bit du signe = 1 donc le nombre est négatif
- $e' = 00011110 = 30$ donc $e = 30 - 127 = -97$
- $M = 2^{-1} + 2^{-2} + 2^{-3} = 0,875$
- Ajouter 1 à la mantisse (Bit caché) donc $m = 1,875$
- Donc $X = -1,875 \cdot 2^{-97}$

Représentation en virgule flottante

Trouver la représentation sur 32 bits du nombre 278.

- Le signe est positif donc le bit du signe=0
- L'exposant e tel que $x = m \times 2^e$ avec $1 \leq |m| < 2$ d'où $e = 8$ car $2^8 \leq x < 2^9$
- $e' = 127 + 8 = 135 = 10000111$
- la mantisse $m = 278/256 = 1,085975$, d'où $m = 1,0001011$ en base 2 donc $M = 0001011$

