Architecture

Systèmes de numération

I Introduction de système décimal

$$14235 = 5 \times 10^0 + 3 \times 10^1 + 2 \times 10^2 + 4 \times 10^3 + 1 \times 10^4$$

Base : 10

Un nbr s'écrit avec des chiffre < 10

Rang d'un chiffre : Position du chiffre dans le nombre (On part de la droite et on commence à 0)

Poids d'un chiffre : 10^{rang}

Pds fort : 1
Pds faible : 5

$$\rightarrow N = a_n a_{n-1} \dots a_1 a_0 = \sum_{i=0}^{n} a_i 10^i$$

où $a_i \in [\![0,9]\!] \; \forall i$

II Généralisation : Base b

Toutes les définitions précédente sont valables en remplaçant 10 par b.

$$\Rightarrow N_b = (a_n a_{n-1} \dots a_0)_b \qquad a_i \in [\![0, b-1]\!]$$

$$= \sum_{i=0}^{n} a_i b^i =$$ Écriture polynomiale

Ex: Base 2 = Binaire $a_i \in \{0,1\}$

a_i BIT (Binary digiT)

Notation : $(101010)_2$
%101010

Base 5 $a_i \in \{0,1,2,3,4\}$

Base 10 Décimal $a_i \in \{0,1,2,3,4,5,6,7,8,9\}$

Notation $(42)_{10}$
42

Base 8 Octal $a_i \in [\![0,7]\!]$

Base 16 Hexadécimal
$a_i \in \{0,\dots,9,A,B,C,D,E,F\}$

Notation : $(CAFE)_{16}$
\$CAFE

$A_{16} \to 10_{10}$
$B_{16} \to 11_{10}$
$C_{16} \to 12_{10}$
$D_{16} \to 13_{10}$
$E_{16} \to 14_{10}$
$F_{16} \to 15_{10}$

III Changement de base : cas des entiers

1) Base b → Base 10

On utilise l'écriture polynomiale, et on fait le calcul.

Ex: $(531)_8$

$\%10011101$

$(ACB)_{12}$

$531 = 1\times 8^0 + 3\times 8^1 + 5\times 8^2$
$= 1 + 24 + 320$
$= 345$

$\%10011101 = 1 + 0\times 2^1 + 2^2 + 2^3 + 2^4 + 2^7$
$= 1 + 4 + 8 + 16 + 128$
$= 157$

$(ACB)_{12}$ ⇒ n'existe pas dans la base.

2) Base 10 → Base b

Rappel : division euclidienne de a par b

$a = qb + r$
avec a, q, b et $r \in \mathbb{N}$

$N_3 (a_n a_{n-1} \dots a_0)_b = \sum_{i=0}^{n} a_i b^i$

$= \sum_{i=1}^{n} a_i b^i + a_0 b^0 = \underbrace{\left(\sum_{i=0}^{n} a_i b^{i-1}\right)}_{\in \mathbb{N}} b$

$= \sum_{i=1}^{n} a_i b^{i-1} = b.\sum_{i=2}^{n} a_i b^{i-2} + a_i$

Méthode des divisions successives

• On divise N par b
(division euclidienne)

• On réitère l'opération avec le quotient
• On s'arrête :
_ au 1er quotient < b (A)
- au 1er quotient = 0 (B)
On écrit le nombre en prenant

(A) Pds fort : Dernier quotient
Pds faible : Premier reste
(B) Pds fort : Dernier reste
Pds faible : Premier reste

Ex : 345 → Base 8
1654 → Base 16
2751 → Base 16

$345 = 8 \times 43 + 1 \qquad = (531)_8$

$1654 = 16 \times 103 + 6 \quad / \; 103 = 16 \times 6 \qquad = (676)_{16}$

$2751 = 16 \times 171 + 31$
$171 = 16 \times 10 + 11 \qquad = (ABF)_{16}$

3) Base b -> Base b'

Base b -> Base 10 -> Base b'

Sauf si b et b' sont des puissance de 2
=> On passe par le binaire.

4) Méthodes de conversion rapide
(Base 2, 8, 16)

a- Base |8 |16 -> Binaire

En octal : $a_i \in \{0, \dots, 7\}$.
0 -> %000 ; 7 -> %111 => Ts les symboles peuvent s'écrire sur 3 bits

En hexa : $a_i \in \{0, \dots, 9, A, \dots, F\}$
0 -> %0000 ; F -> %1111
-> Tous les symboles s'écrivent sur 4 bits

Pour convertir 1 nbr |octal |hexadecimal| en binaire, on code chaque symbole sur |3 |4| bits et on juxtapose les résultats.

Ex : $(752)_8 = 7\times8^2 + 5\times8^1 + 2\times8^0$
$= 7\times(2^3)^2 + 5\times(2^3)^1 + 2\times(2^3)^0$
$= (2^2+2^1+2^0)\times2^6 + (2^2+2^0)\times 2^3 + 2^1\times2^0$

$= 2^8 + 2^7 + 2^6 + 2^5 + 2^3 + 2^1$

$= \% 111101010$

• $(131)_8 = 1 \times 8^2 + 3 \times 8^1 + 1 \times 8^0$

001 011 001

$\% 1011001 = 1 \times 2^{3^2} + 3 \times 2^{3^1} + 1 \times 2^{3^0}$

=

$(CAFE)_{16} =$

$\% 1100\ 1010\ 1111\ 1110$

<u>6- Binaire → Base $\big|^8_{16}$</u>

Processus inverse On regroupe les bits par 3 (Octal) ou 4 (hexa) et on convertit en décimal chaque groupement, Puis on juxtapose les résultats obtenus.

<u>Ex</u> : $\%$ 1001110111110001 → Base $\big|^8_{16}$

<u>Base 8</u> : 1 1 6 7 6 1

$(116761)_8$

<u>Base 16</u> : 9 D F 1

$ 9DF1

IV Changements de Base : Cas des nbres fractionnaires

1- Base b → Base 10

Rq $42,31 = 4\times10^1 + 2\times10^0 + 3\times10^{-1} + 1\times10^{-2}$

⇒ On généralise la décomposition polynômiale aux puissances <0 de la base et on utilise la même méthode que pour les entiers naturels.

Ex : $537,2C

Base 10 %0101 0011 0111,0010 1100

$$537,2C = 5\times16^2 + 3\times16 + 7 + 2\times16^{-1} + 12\times16^{-2}$$

$$= 1335 + \frac{2}{2^4} + \frac{12}{(2^4)^2}$$

$$= 1335,171875$$

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

$2^{-1} = 0,5$
$2^{-2} = 0,25$
$2^{-3} = 0,125$
$2^{-4} = 0,0625$

2- Base 10 → Base b

NE , NF

$\sum_{i=0}^{n} a_i 16^i$

Méthode

- On convertit la partie entière par div. successives.
- On convertit la partie fractionnaire par multiplications successives.

⚠ Pas de critère d'arrêt.

$$N_F = \sum_{i=1}^{m} a_i b^{-i}$$

$$N_F \times b = \sum_{i=1}^{n} a_{-i}\, b^{-i+1} = \underbrace{a_{-1} b^0}_{\text{entier}} + \underbrace{\sum_{i=2}^{m} a_{-i}\, b^{1-i}}_{\text{fractionnaire.}}$$

$$\left(\sum_{i=2}^{m} a_{-i}\, b^{1-i} \right) \times b = a_{-2} + \sum_{i=3}^{m} a_{-i}\, b^{2-i}$$

Ex : $351{,}943 \rightarrow$ Base 8

$$351 = 3 \times 8^2 + 5 \times 8^1 + 7 \times 8^0$$
$$= (537)_8$$

$0{,}943 \times 8 = 7{,}544 \rightarrow 0{,}544 \times 8 = 4{,}352$

$\downarrow$

$0{,}352 \times 8 = 2{,}816$

$\downarrow$

$0{,}816 \times 8 = 6{,}528$

$$351{,}943 = (537{,}7426)_8$$

Rq Precision en decimal $= 10^{-3} = 0{,}001$

Precision en octal $= 8^{-4} \simeq 0{,}0002$

Rq $(537{,}742)_8 \rightarrow$ Base 10

$(537{,}7426)_8 \rightarrow$ Base 10

V Opérations arithmétiques

1. Addition

Ex : $\%0 + \%0 = \%0$

$\%0 + \%1 = \%1 + \%0 = \%1$

$\%1 + \%1 = \%10$

```
   %10011101        532₈        $ABCD
 + %01010011      + 478       + $ 823
 -----------      -----       -------
   %11110000        601       $B3F0
```

2. Soustraction

Rappel : En base 10

```
   931
 - 142
 -----
   789
```

$\%0 - \%0 = \%0$

$\%1 - \%0 = \%1$

$\%0 - \%1 = \%1$

Ex :

```
   %1000 1100        $42B61
 - %0101 0001      - $123AC
 ------------      --------
    0011 1011        $307B5
```

3. Multiplication (Base 2)

```
      1001                 1001 1100
    x  101              x       1101
    ------              ------------
      1001                  10011100
    (0000)                 00000000.
    1001..                1001 1100..
    ------               10011100...
    101101              ------------
                         11111101100
```

4- Division

101001 | 101

Si

1000,0011

-101

00

001000

-101

0110

-101

010

11011011 | 1010

1010 1 ,111

-1010

00111

1110

-1010

01001

10011

-1010

010010

-1010

010000

-1010

01110

VI. Conversion des nbr signés (Base 2)

1- Complément à 1

On obtient le complément à 1 d'un nbr binaire en remplaçant chaque 0 par 1 et chaque 1 par 0.

Ex : $C_1(\%1011\,0011) = \%0100\,1100$

2-

$$C_2(\%N) = C_1(\%N) + \%1$$

C_2 : $\%1011\,0011 \xrightarrow{C_1} \%0100\,1100$

$\%0100\,1101$

$\%1010\,1000$

C_2 : $\%0101\,0111$

C_2 : $\%0101\,1000$

$\%1000\,0000$

C_2 : $\%1000\,0000$

$$\begin{array}{rl} & \%1011\,0011 \\ + & \%0100\,1101 \\ \hline (1) & \%0000\,0000 \end{array}$$

3- Soustraction par addition du complément à 2

$$\%A - \%B = \%A + C_2(\%B) - 2^n$$

puissance de 2 immédiatement supérieure à %A

$$\begin{array}{rl} & \%1000\,1100 \\ - & 0101\,0001 \end{array} \rightarrow \begin{array}{rl} & \%1000\,1100 \\ + & \%1010\,1111 \\ \hline (1) & \%0011\,1011 \end{array}$$

$\%A - \%B = \%A + C_2(\%B)$

$\%A + |-\%B|$

4) Normalisation et nombres signés

Rq : $C_2(\%101010) = \%010110$

$C_2(\%00101010)$ $\%11010110$

→ Le nbr de bits avec lequel on va coder notre nbre est important.

C_2 = complément à 2

- On normalise l'écriture du nbr binaire (on fixe la taille)
- Le bit de poids le + fort indique le signe :

→ 0 → nbr ≥ 0

→ 1 → nbr < 0

(= bit de signe)

a) Conversion décimal → binaire signé

- Si $N_{10} \geq 0$: divisions successives
- Si $N_{10} < 0$: On code, en binaire sa valeur absolue, puis, on prend son complément à 2.

Ex : −42 avec 1 minimum de bits :

$|-42| = \%0101010 \xrightarrow{C_2} \%1010110$

Rq : En binaire signé, on ne peut pas écrire 42 et -42 sur 6 bits

$\Rightarrow$ 7 bits

b) Conversion Binaire Signé $\rightarrow$ Décimal

On regarde le bit de poids fort.

. S'il vaut 0 $\Rightarrow$ N est ≥ 0

On le convertit directement en décimal en utilisant la forme polynomiale.

. S'il vaut 1 $\Rightarrow$ N est < 0

On convertit le $C_2(N)$ avec l'écriture polynomiale et on ajoute 1 signe $\ominus$ devant le résultat.

Ex : Sur 1 octet :

a) % 1001 1100 = -100
b) % 0110 1001 = 105
c) % 101 1100 = 92

a) $N < 0 \Rightarrow C_2(\%10011100) = \%01100100$

$= 4 + 32 + 64 = 100$

b) $64 + 32 + 8 + 1 = 105$

c) $64 + 16 + 8 + 4 = 92$

c) Valeurs limites

Sur n bits, on peut écrire 2^n combinaisons / valeurs.

→ En binaire non signé, on peut coder des valeurs de 0 à $2^n - 1$.

En binaire signé on peut coder des valeurs comprises entre -2^{n-1} et $2^{n-1} - 1$ (car on code 2^{n-1} valeurs ≥ 0) et 2^{n-1} valeurs.

42.

→ Binaire non signé : 6 bits car $42 \in [0, 2^6 - 1]$
$[0, 63]$

→ Binaire signé : sur 6 bits car $42 \in [-32, 31]$

→ $42 \notin [-32, 31]$

$42 \in [-64, 63]$ (7 bits)

Rq %1111 1111 = -1

%1000 0000 = -128

Hexadécimal signé :

$\$ S_F \ldots\ldots S_f$ → %0 ...... ≥ 0 ; → %1 ...... < 0

F = Symbole de poids fort

Sf = Symbole de poids faible

⇒ Si $S_F \geq 8$ ⇒ nbr < 0

Si $S_F \leq 7$ ⇒ nbr ≥ 0

Extension de signe :

Soit N_b, nombre binaire signé écrit sur 1 octet.
Comment l'écrire sur 2 octets

Ex : %1000 0000 = %1111 1111 1000 0000
%0111 1111 = %0000 0000 0111 1111

Méthode : On duplique le bit de signe autant de fois qu'il faut pour atteindre la taille demandée

\$80 = \$FF80

5- Opérations en nbr signé

Idem binaire non signé

+ %a - %b avec %a < %b

~~%0111 1111~~
~~+ %1~~
~~= %1000 0000~~

VI Différents types de Codes

1) Code BCD

Binaire Codé Décimal

Chaque chiffre (décimal) est écrit sur 4 bits
et on juxtapose les groupements.

Ex : 1623 → BCD

↳ $(1000\ 0101\ 1100\ 1001)_{BCD}$

a) $(0110\ 0010\ 0011)_{BCD}$

b) 85 → faux pas du BCD

2) Code Gray (Binaire réfléchi)

Binaire naturel

0001
0010
0011
0100
0101
0110
0111
1000
|