C.U. AbdelhafidBoussouf - Mila
Institut des Sciences et Technologie
Département MI
1ère année Mathématiques
Matière : Structure machine1

Année universitaire : 2022/2023

Solution TD n° 02

Exercice 01

1. Les intervalles de codage :
   - Sur n bits :
     - S/VA : $[-(2^{(n-1)}-1), +(2^{(n-1)}-1)]$
     - CA1 : $[-(2^{(n-1)}-1), +(2^{(n-1)}-1)]$
     - CA2 : $[-(2^{(n-1)}), +(2^{(n-1)}-1)]$
   - Sur 8 bits :
     - S/VA : $[-(2^{(8-1)}-1), +(2^{(8-1)}-1)] = [-127, +127]$
     - CA1 : $[-(2^{(8-1)}-1), +(2^{(8-1)}-1)] = [-127, +127]$
     - CA2 : $[-(2^{(8-1)}), +(2^{(8-1)}-1)] = [-128, +127]$
   - Sur 16 bits :
     - S/VA : $[-(2^{(16-1)}-1), +(2^{(16-1)}-1)] = [-32767, +32767]$
     - CA1 : $[-(2^{(16-1)}-1), +(2^{(16-1)}-1)] = [-32767, +32767]$
     - CA2 : $[-(2^{(16-1)}), +(2^{(16-1)}-1)] = [-32768, +32767]$
2. $-(512)_{10} = -(1000000000)_2 \rightarrow$ Il faut au minimum 10 bits pour encoder la valeur absolue.

   Dans la représentation S/VA, il faut ajouter un bit de signe $\rightarrow$Il faut au minimum 11 bits pour encoder la valeur.

   1 octet = 8 bits $\rightarrow$ Il faut au minimum 2 octets pour encoder la valeur.
3.

Décimal	S/VA	CA1	CA2
+27	00011011	00011011	00011011
-45	10101101	11010010	11010011
-128	/	/	10000000
-117	11110101	10001010	10001011
+117	01110101	01110101	01110101
-86	11010110	10101001	10101010
+41	00101001	00101001	00101001
-14	10001110	11110001	11110010
+114	01110010	01110010	01110010

- **Remarque :** On ne peut pas encoder la valeur (-128) en S/VA et CA1 car elle hors des intervalles du codage vues dans la 1ère question de l'exercice (il faut plus que 8 bits pour l'encoder).

Exercice 02

1.

- $(-4)_{10}$:
  - Sur 1 octet : $(-4)_{10} = -(00000100)_2 = (11111011)_{CA1} = (FB)_{CA1\ 16}$
  - Sur 2 octets : $(-4)_{10} = -(0000000000000100)_2 = (1111111111111011)_{CA1} = (FFFB)_{CA1\ 16}$
- $(+4)_{10}$:
  - Sur 1 octet : $(+4)_{10} = +(00000100)_2 = (00000100)_{CA1} = (04)_{CA1\ 16}$
  - Sur 2 octets : $(+4)_{10} = +(0000000000000100)_2 = (0000000000000100)_{CA1} = (0004)_{CA1\ 16}$

2.

a) S'il s'agit d'un nombre non signé :

$(CB)_{16} = (11001011)_2 = (203)_{10}$

b) S'il s'agit d'un nombre signé : il est représenté différemment selon que l'on suppose la représentation en S/VA, en CA1 ou en CA2 :

- S/VA : $(CB)_{S/VA\ 16} = (11001011)_{S/VA} = -(1001011)_2 = -(75)_{10}$
- C.A.1 : $(CB)_{CA1\ 16} = (11001011)_{CA1} = -(00110100)_2 = -(52)_{10}$
- C.A.2 : $(CB)_{CA2\ 16} = (11001011)_{CA2} = -(00110101)_2 = -(53)_{10}$

3.

- $(FFFF)_{CA1} = (1111111111111111)_{CA1} = -(0000000000000000)_2 = -(0)_{10}$
- $(7FFF)_{CA1} = (0111111111111111)_{CA1} = +(0111111111111111)_2 = +(32767)_{10}$
- $(8000)_{CA2} = (1000000000000000)_{CA2} = -(1000000000000000)_2 = -(32768)_{10}$
- $(00FF)_{CA2} = (0000000011111111)_{CA2} = +(0000000011111111)_2 = +(255)_{10}$

Exercice 03

- **$(107-67)_{10}=(?)_{cà2}$**

  $(107-67)_{10} = (107)_{10} + (-67)_{10}$

  $(107)_{10} = (01101011)_2 = (01101011)_{cà2}$

  $(-67)_{10} = -(01000011)_2 = (10111101)_{cà2}$

```
  1 1 0 1 1 1 1 0 1 1 0 1 1 1
+       1 0 1 1 1 1 0 1
-------------------------------
= (1 0 0 1 0 1 0 0 0)cà2
  retenue
```

Les deux nombres sont de signes opposés → Il n'y a pas de débordement.

$(00101000)_{cà2}$=+$(00101000)_2$ = $(+ 40)_{10}$ = $(107 - 67)_{10}$

Résultat correct → Il n'y a pas de débordement.

- **$(-106- 5)_{10} = (?)_{cà2}$**

$(-106-5)_{10}$ = $(-106)_{10}$ + $(-5)_{10}$

$(-106)_{10}$= - $(01101010)_2$= $(10010110)_{cà2}$

$(-5)_{10}$ = -$(00000101)_2$ = $(11111011)_{cà2}$

$^{1\ 1}1^{1}0^{1}0^{1}1^{1}0^{1}1\ 1\ 0$

\+ 1 1 1 1 1 0 1 1

= (1 | 1 0 0 1 0 0 0 1)$_{cà2}$

retenue

Les deux opérandes sont de même signe → la retenue de l'addition des bits de signes et la retenue de l'addition des bits juste avant les bits de signe sont égales (=1) → Il n'y a pas de débordement.

$(10010001)_{cà2}$ =-$(01101111)_2$ = $(-111)_{10}$= $(-106 - 5)_{10}$

Résultat correct→ Il n'y a pas de débordement.

- **$(111+ 25)_{10} = (?)_{cà2}$**

$(111)_{10}$= $(01101111)_2$ = $(01101111)_{cà2}$

$(25)_{10}$= $(00011001)_2$ = $(00011001)_{cà2}$

$^{1}0^{1}1^{1}1^{1}0^{1}1^{1}1^{1}1\ 1$

\+ 0 0 0 1 1 0 0 1

= (1 0 0 0 1 0 0 0)$_{cà2}$

Les deux opérandes sont de même signe → la retenue de l'addition des bits de signes est (=0) et la retenue de l'addition des bits juste avant les bits de signe (=1) sont différentes → Cas de débordement.

$(10001000)_{cà2}$ = -$(01111000)_2$ = $(-120)_{10}$

$(111 + 25)_{10}$ = $(136)_{10}$

Résultat incorrect→ Cas de débordement.

(La cause du débordement est que la somme des deux nombres (**136**) n'appartient pas à l'intervalle de codage de CA2 sur 8 bits qui est **[-128,+127])**

- **$(-126- 85)_{10} = (?)_{cà2}$**

$(-126 - 85)_{10}$ = (-126) + $(-85)_{10}$

$(-126)_{10}$= - $(01111110)_2$ = $(10000010)_{cà2}$

$(-85)_{10}$= - $(01010101)_2$ = $(10101011)_{cà2}$

$^{1\ 0}1\ 0\ 0\ 0\ 0\ ^{1}0\ 1\ 0$

$+\quad 1\ 0\ 1\ 0\ 1\ 0\ 1\ 1$

$= (1\ 0\ 0\ 1\ 0\ 1\ 1\ 0\ 1)_{cà2}$

retenue

Les deux opérandes sont de même signe → la retenue de l'addition des bits de signes est (=1) et la retenue de l'addition des bits juste avant les bits de signe (=0) sont différentes → Cas de débordement.

$(-126 - 85)_{10} = -(211)_{10}$

$(00101101)_{cà2} = -(00101101)_2 = +(45)_{10}$

Résultat incorrect → Cas de débordement.

(La cause du débordement est que la somme des deux nombres (**-211**) n'appartient pas à l'intervalle de codage de CA2 sur 8 bits qui est **[-128,+127]).**

Exercice 04

1.

- Le plus petit nombre réel (positif) représentable en IEEE 754 simple précision :

$\boldsymbol{Nnp_{min}}$ est représenté en :

$$0\ 00000001\overbrace{000\ldots\ldots0}^{\textbf{23 zéros}}$$

$\boldsymbol{Nnp_{min}} = +a_{min} \times \mathbf{2}^{b_{min}}$

(Eb = 0 et Eb = (2^8 - 1) sont des cas particuliers utilisés pour des nombres dénormalisés).

$b_{min} = Er_{min} = Eb_{min} - 127 = 1 - 127 = \mathbf{-126}$

$a_{min} = \mathbf{1.000\ \ldots\ 0}$

Donc, $\boldsymbol{Nnp_{min}} = 1 \times \mathbf{2}^{-126} = \mathbf{2}^{-126}$

- Le plus grand nombre réel (positif) représentable en IEEE 754 simple précision :

$\boldsymbol{Nnp_{max}}$ est représenté en :

$$0\ 11111110\overbrace{111\ldots\ldots\ldots1}^{\textbf{23 uns}}$$

$\boldsymbol{Nnp_{max}} = +\boldsymbol{a_{max}} \times \mathbf{2}^{bmax}$

$b_{max} = \boldsymbol{Er_{max}} = \boldsymbol{E}b_{max} - \mathbf{127} = 254 - \mathbf{127} = \mathbf{127}$

$\boldsymbol{a_{max}} = \mathbf{1.111\ \ldots\ 1} = \mathbf{2} - \mathbf{2}^{-23}$

Donc, $\boldsymbol{Nnp_{max}} = (\mathbf{2} - \mathbf{2}^{-23}) \times \mathbf{2}^{127} \approx \mathbf{3.4} \times \mathbf{10}^{38}$

2.

$(11.011)_2 = 1 * 2^0 + 1 * 2^1 + 0 * 2^{-1} + 1 * 2^{-2} + 1 * 2^{-3} = (3.375)_{10}$

$(11.625)_{10} = (1011.101)_2$ (on utilise les divisions et multiplications successives).

3.

- $(+ 45)_{10} = +(101101)_2 = +(1.01101)_2 * 2^{+5}$

  La mantisse M = 01101

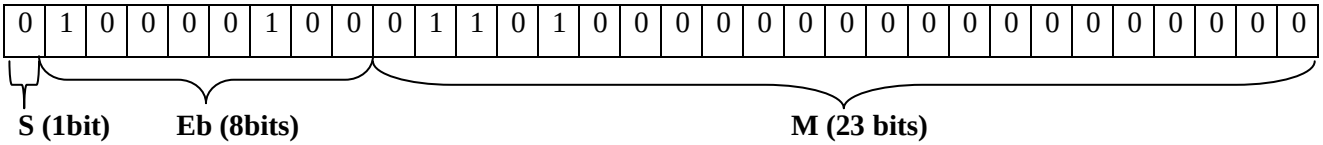
  Le signe S= 0 (positif)

  L'exposant réel Er = 5

  L'exposant biaisé Eb = Er+127 = 5 + 127 = 132

  $Eb = (10000100)_2$

  Et alors la représentation en virgule flottante (IEEE 754 simple précision) est :



  Le nombre s'écrit :

  N = 01000010001101000000000000000000

- $(+ 13.5)_{10} = +(1101.1)_2 = +(1.1011)_2 * 2^{+3}$

  La mantisse M= 1101

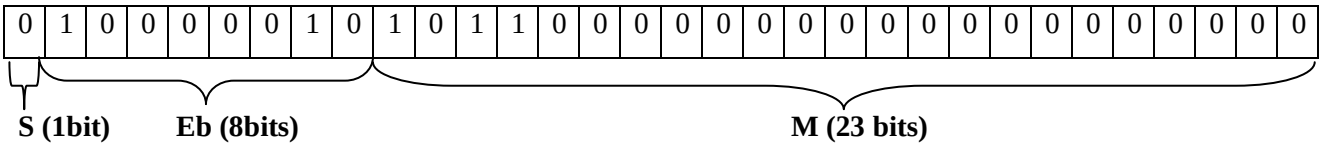
  Le signe S = 0 (positif)

  L'exposant réel Er = 3

  L'exposant biaisé Eb = Er+127 = 3 + 127 = 130

  $Eb = (10000010)_2$

  Et alors la représentation en virgule flottante (IEEE 754 simple précision) est :



  Le nombre s'écrit :

  N = 01000001010110000000000000000000

- $(-160.75)_{10} = -(10100000.11)_2 = -(1.010000011)_2 * 2^{+7}$

  La mantisse M = 010000011

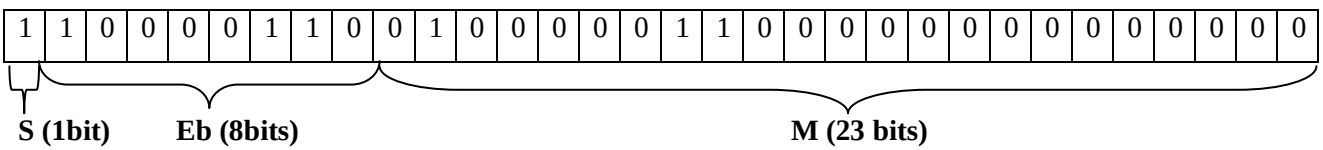
  Le signe S = 1 (négatif)

  L'exposant réel Er = 7

  L'exposant biaisé Eb = Er+127 = 7 + 127 = 134

  $Eb=(10000110)_2$

  Et alors la représentation en virgule flottante (IEEE 754 simple précision) est :



  Le nombre s'écrit :

  N = 11000011001000001100000000000000

- $(-32.625)_{10} = -(100000.101)_2 = -(1.00000101)_2 * 2^{+5}$

  La mantisse M= 00000101

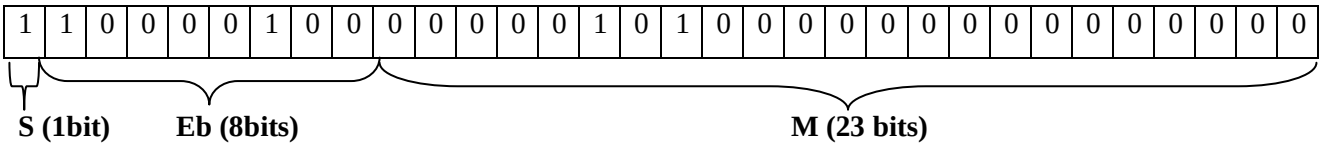
  Le signe S = 1 (négatif)

  L'exposant réel Er = 2

  L'exposant biaisé Eb = Er+127 = 5 + 127 = 132

  $Eb=(10000100)_2$

  Et alors la représentation en virgule flottante (IEEE 754 simple précision) est :



  Le nombre s'écrit :

  N = 11000010000010100000000000000000

4.

- $(17BE0000)_{16}$ = (00010111101111100000000000000000)

  S (1bit) **Eb (8bits)** **M (23 bits)**

  Le signe S = 0 (positif)

  $Eb = (00101111)_2 = 47$

  Eb = Er + 127 => Er = Eb - 127 = 47-127= -80

  Le nombre écrit en forme normalisée : **$(+/-)\ (1,M)_2 * 2^{ER}$**

  $(17BE0000)_{16} = + (1.011111)_2 * 2^{-80} = + (1011111)_2 * 2^{-6} * 2^{-80} = (+ 95) * 2^{-86}$

- $(C3F00000)_{16}$ = (11000011111100000000000000000000)

  S (1bit) **Eb (8bits)** **M (23 bits)**

  $Eb = (10000111)_2 = 135$

  Eb = Er + 127 => Er = Eb - 127 = 135-127 = + 8

  $(C3F00000)_{16} = - (1.111)_2 * 2^{+8} = - (1111)_2 * 2^{-3} * 2^{+8} = -(15)_{10} * 2^{+5}$

- $(42E48000)_{16}$ = (01000010111001001000000000000000)

  S (1bit) **Eb (8bits)** **M (23 bits)**

  $Eb = (10000101)_2 = 133$

  Eb = Er + 127 => Er = Eb - 127 = 133-127 = + 6

  $(42E48000)_{16} = + (1.11001001)_2 * 2^{+6} = + (111001001)_2 * 2^{-8} * 2^{+6} = (+ 457)_{10} * 2^{-2}$

- $(80000008)_{16}$ = (10000000000000000000000000001000)

  S (1bit) **Eb (8bits)** **M (23 bits)**

  $Eb = (00000000)_2 = 0_{10}$ => le nombre est **dénormalisé**.

Exercice 05

1.

$(87)_{10} = (1010111)_2 = (10000111)_{BCD} = (10111010)_{XS3} = (1111100)_{GR}$

$(637)_8 = (415)_{10} = (110011111)_2 = (010000010101)_{BCD} = (011101001000)_{XS3} = (101010000)_{GR}$

$(1101001)_{BCD} = (01101001)_{BCD} = (69)_{10} = (1000101)_2 = (10011100)_{XS3} = (1100111)_{GR}$

2. Faire en 3 méthodes :

- Une première méthode : commencer l'écriture de nombres en gray depuis le 0 :

Nombres en décimal	Nombres en gray
0	00000
1	00001
2	00011
3	00010
4	00110
5	00111
6	00101
7	00100
8	01100
9	01101
10	01111
11	01110
12	01010
13	01011
14	01001
15	01000
16	11000
17	11001
18	11011
19	11010
20	11110
21	11111
22	11101
23	11100
24	10100
25	10101
26	10111
27	10110
28	10010
29	10011
30	10001
31	10000

- Une deuxième méthode : conversion directe de chaque nombre du binaire en gray.

  Par exemple :

  $(13)_{10} = (1101)_2 = (1011)_{GR}$

- Une troisième méthode : commencer par l'écriture du nombre 13 en gray, puis calculer le nombre suivant et le suivant…en utilisant la règle suivante pour passer d'un nombre au suivant :
    - Si le nombre de 1 est pair, il faut inverser le dernier chiffre (le chiffre du poids faible).
    - Si le nombre de 1 est impair, il faut inverser le chiffre situé à gauche du 1 le plus à droite.

  Par exemple :

  $(13)_{10} = (1101)_2 = (1011)_{GR}$, le nombre de 1 est impair alors :

  $(14)_{10} = (1001)_{GR,}$ le nombre de 1 est pair alors :

  $(15)_{10} = (1000)_{GR}$

Nombres en décimal	**Nombres en gray**
13	01011
14	01001
15	01000
16	11000
17	11001
18	11011
19	11010
20	11110
21	11111
22	11101
23	11100
24	10100
25	10101