

Contrôle Continu UE INF451 : Architectures des ordinateurs

Mars 2021, durée 1 h 30

Document : 1 A4 R/V personnel manuscrit autorisé ; caletttes et téléphones portables interdits.
 La plupart des questions sont indépendantes, si vous avez du mal avec l'une, passez à la suivante.
 Tant que possible, indiquez bien tous les détails et justifiez vos réponses.
 Le barème est donné à titre indicatif.

1 Numération et opération en binaire et en complément à 2 (6 points)

- (a) Déterminer le nombre de bits minimum pour représenter ensemble les valeurs suivantes puis donner la représentation binaire de ces entiers relatifs avec le nombre de bits choisi globalement (codage en complément à 2) : $(+121)_{10}$, $(-323)_{10}$. **(2 points)**

Réponse : Il faut l'intervalle $[-512, +511]$ donc 10 bits

$$(+121)_{10} = (00\ 0111\ 1001)_2 = 79_{16}$$

$$(+323)_{10} = (01\ 0100\ 0011)_2$$

$$\text{Complément à 1 : } (10\ 1011\ 1100)_2$$

$$\text{Complément à 2 : } (-323)_{10} = (10\ 1011\ 1101)_2 = 2BD_{16}$$

- (b) Donner les valeurs binaires et décimales des 2 entiers relatifs suivants codés sur 16 bits en complément à 2 : $(2021)_{16}$ et $(FA57)_{16}$. **(2 points)**

Réponse : $(2021)_{16} = (0010\ 0000\ 0010\ 0001)_2$ donne en décimal 8225

$(FA57)_{16} = (1111\ 1010\ 0101\ 0111)_2$ donne en décimal -1 449

$$C2(FA57)_{16} = 0000\ 0101\ 1010\ 1001_2 : -(5 * 256 + 10 * 16 + 9) = -(1024 + 256 + 169)$$

- (c) Poser chacune des opérations suivantes sur 1 octet, effectuer l'opération en binaire (sur 1 octet, avec les retenues) et donner une interprétation de l'opération pour la représentation binaire usuelle et pour le codage en complément à 2 (sur 1 octet) ainsi que la valeur des indicateurs (Z, N, C et V) : $(0111\ 1011)_2 + (1110\ 1010)_2$ et $(0110\ 1111)_2 - (1101\ 1110)_2$ (soustraction par addition du complémentaire). Pour la soustraction, détailler la méthode employée. **(2 points)**

Réponse :

im2ag-turing:[~]: add 8 123 234

Bit numbers		if natural	if signed
7654 3210			
0111 1011 left	: 0x 7b -->	123 or	+123
+ 1110 1010 right	: 0x ea -->	234 or	-22
C=1 == 1111 0100 < c0=0	(in carries)		
V=0 ^ ---- ----			
Z=0 N=0->0110 0101 =	: 0x 65 -->	101 or	+101

```

im2ag-turing:[~]: subc2 8 111 222
      Bit numbers
      7654 3210
      if natural      if signed
      0110 1111 left   : 0x    6f -->    111 or    +111
      + 0010 0001 right : 0x    21 -->     33 or    +33
      C=0 != 1101 1111 < c0=1 (in carries)
      V=1 ^ ---- ----
      Z=0 N=1->1001 0001 =    : 0x    91 -->    145 or    -111

```

On a choisi ici d'utiliser le complément à 1 comme opérande plus 1 en retenue initiale. Il est possible de prendre le complément à 2 0010 0010 comme opérande et une retenue initiale nulle.

2 Triangle de Pascal (ARM)

voir Caseine

<https://moodle.caseine.org/mod/vpl/view.php?id=40048>

3 Codage Négabinaire (6 points)

“Le système négabinaire (base -2) est un système de numération positionnel non standard utilisé dans l'ordinateur expérimental polonais BINEG, construit en 1957-59. Il possède la propriété inhabituelle d'avoir les nombres négatifs et positifs représentés sans un bit de signe” source Wikipédia.

Le négabinaire (base -2), c'est comme le binaire (base 2), les chiffres possibles sont aussi les bits 0 et 1. La différence ? en binaire, un nombre $(N_3N_2N_1N_0)_2$ a pour valeur $\sum_i N_i \times 2^i$; en négabinaire, un nombre $(N_3N_2N_1N_0)_{-2}$ a pour valeur $\sum_i N_i \times (-2)^i$.

Questions :

- (a) Donnez les valeurs décimales des 16 premiers nombres binaires et négabinaires sur 4 bits. Faites un tableau à trois colonnes (1 : combinaisons binaires sur 4 bits [dans l'ordre], 2 : valeurs associées en binaire habituel, 3 : valeurs associées en négabinaire). Encadrez les lignes où la valeur binaire est égale à la valeur négabinaire. (2 points)

Réponse :

code	bin	negabin	code	bin	negabin	code	bin	negabin	code	bin	negabin
0000	0	0	0001	1	1	0010	2	-2	0011	3	-1
0100	4	4	0101	5	5	0110	6	2	0111	7	3
1000	8	-8	1001	9	-7	1010	10	-10	1011	11	-9
1100	12	-4	1101	13	-3	1110	14	-6	1111	15	-5

Les valeurs sont identiques si les chiffres de rangs pairs sont nuls.

- (b) Donnez le plus grand nombre sur 8 bits, en binaire, en hexadécimal et en décimal dont la valeur en binaire est égale à la valeur en négabinaire (**1 points**).
- (c) Dénombrez le nombre de demi-mots de 16 bits où la valeur binaire est égale à la valeur négabinaire. Expliquez votre calcul. (**1 points**).

Réponses :

Le plus grand nombre sur 8 bits de valeurs identiques est celui dont tous les bits de rangs impairs sont nuls et tous ceux de rang pairs non nuls :

$$01010101_2 = 55_{16} = 5 * 16 + 5 = 85$$

Prendre toutes les combinaisons possibles des 8 bits de rang pairs, en mettant 0 pour tous les bits de rangs impairs, soit $2^8 = 256$ nombres de valeur identique dans les deux conventions.

Le négabinaire (base -2), c'est comme le binaire (base 2), pour trouver le codage d'un nombre il "suffit" d'utiliser le même algorithme de divisions successives : en binaire, des divisions successives par 2 ; en négabinaire, des divisions successives par -2. Attention, cependant, ne pas se tromper dans les divisions par -2 des nombres impairs (rappel, le reste ne peut prendre que les 2 valeurs possibles 0 et 1) : 7 divisé par -2 vaut -3 reste 1 ; -9 divisé par -2 vaut 5 reste 1.

Questions :

- (d) Appliquer l'algorithme des divisions successives par -2 pour 22 et donner la décomposition négabinaire de 22 sur 8 bits. (**1 points**)
- (e) Appliquer l'algorithme des divisions successives par -2 pour -34 et donner la décomposition négabinaire de -34 sur 8 bits. (**1 points**)

Réponse :

$$\begin{aligned} 22 &= -2 * -11 \text{ reste } 0 \\ -11 &= -2 * 6 \text{ reste } 1 \\ 6 &= -2 * -3 \text{ reste } 0 \\ -3 &= -2 * 2 \text{ reste } 1 \\ 2 &= -2 * -1 \text{ reste } 0 \\ -1 &= -2 * 1 \text{ reste } 1 \\ 1 &= -2 * 0 \text{ reste } 1 \end{aligned}$$

$$0 \ 110 \ 1010 \quad 64 \ -32 \ -8 \ -2 = 32 - 10 = 22$$

$$\begin{aligned} -34 &= -2 * 17 \text{ reste } 0 \\ 17 &= -2 * -8 \text{ reste } 1 \\ -8 &= -2 * 4 \text{ reste } 0 \\ 4 &= -2 * -2 \text{ reste } 0 \\ -2 &= -2 * 1 \text{ reste } 0 \\ 1 &= -2 * 0 \text{ reste } 1 \end{aligned}$$

$$00 \ 10 \ 0010 = -32 \ -2 = -34$$