Contrôle Continu - Architecture des Ordinateurs

Vendredi 1er Avril 2011
Documents de cours autorisés. Calculatrices interdites.
Durée 1h25

Tous les résultats doivent être justifiés.

1 Représentations des entiers

1. Convertissez en hexadécimal les entiers suivants : 48_{10}, 62_{10}, 221_{10}.

Solution:
- $48 = 16 + 3 = 30_{16}$
- $62 = 32 + 16 + 14 = 3 * 16 + 14 = 3E_{16}$
- $221 = 128 + 64 + 16 + 13 = (8 + 4 + 1) * 16 + 13 = DD_{16}$

2. Représentez les entiers suivants au format binaire en complément à 2 sur 8 bits : 33_{10}, -20_{10}, 74_{10}, -63_{10}.

Solution:
- $33 = 0010001_{2}$
- $20 = 00010100_{2}$ donc $-20 = 00010100 + 1 = 11101011 + 1 = 11101100_{CA2}$
- $74 = 01001110_{2}$
- $63 = 64 - 1 = 00111110_{2}$ donc $-63 = 11000000 + 1 = 11000001_{CA2}$


Solution:
- $11000001$
- $+11101100$
- \[\begin{array}{c}
110101101 \\
\text{Résultat} = 10101101 = -83
\end{array}\]

2 Codes

On souhaite transmettre sur un canal un mot de 8 bits $A = a_7a_6...a_1a_0$. On utilise le code $C$ suivant pour la transmission : $C(A) = AA = a_7a_6...a_1a_0a_7a_6...a_1a_0$. Par exemple $C(1001101) = 1000110110001101$.

1. Quelle est la distance de Hamming $d(C(240), C(170))$ entre $C(11110000)$ et $C(10101010)$ ?
Solution: 11110000 et 10101010 ont 4 bits de différence donc \( C(11110000) \) et \( C(10101010) \) ont 8 bits de différence. La distance de Hamming entre eux est de 8.

2. Quelle est la distance de Hamming du code \( C \)? Pour rappel

\[
d_h(C) = \min\{d(C(M),C(N))|\forall M \neq N\}
\]

Solution: Soit deux mots différents de 8 bits \( M \) et \( N \). \( d(C(M),C(N)) = 2 \times d(M,N) \). La distance minimale entre \( M \) et \( N \) est 1 donc \( d_h(C) = \min\{d(C(M),C(N))|\forall M \neq N\} = \min\{2.d(M,N)|\forall M \neq N\} = 2 \).

3 Nombres flottants

On représentera les réels sur 16 bits selon la norme suivante (demi précision):

<table>
<thead>
<tr>
<th>S (Signe)</th>
<th>E (Exposant)</th>
<th>P (Pseudomantisse)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 bit</td>
<td>5 bits</td>
<td>10 bits</td>
</tr>
</tbody>
</table>

1. Représentez en demi-précision les réels suivants (on utilisera l’arrondi au plus près si nécessaire):

\( 3 \)

\( 0.8 \)

\(-0.625 \)

Solution: 

\( 3 = 0 \ 10000 \ 1000000000 \)

— Signe : positif.

— \( 3 = 112 = 1,1.2^1 \)

— Pseudo-mantisse : 10000…

— Exposant : \( 15 + 1 = 16 = 10000 \)

\( 0.8 = 0 \ 01110 \ 1001100110 \)

— Signe : positif.

— Par multiplication par 2 successive :

\( .8 \times 2 = 1.6 \ 1 \)

\( .6 \times 2 = 1.2 \ 1 \)

\( .2 \times 2 = 0.4 \ 0 \)

\( .4 \times 2 = 0.8 \ 0 \)

\( \ldots \)

\( 0.8 = 0.11001100 \ldots \times 2^{-1} \times 1.1001100110 \ldots \)

— Pseudo-mantisse : 10011001100

— Exposant : \( 15 - 1 = 14 = 01110 \)

\(-0.625 = 1 \ 01110 \ 010000000 \)

— Signe : négatif.

— \( 0.625 = 0.5 + 0.125 = 0.101 = 1.01 \times 2^{-1} \)

— Pseudo-mantisse : 010000000

— Exposant : \( 15 - 1 = 14 = 01110 \)

2. En partant de leur représentation binaire, effectuez la somme des réels 0.8 et 3 et donnez le résultat en binaire sur 16 bits (en utilisant l’arrondi au plus près, si nécessaire). Détaillez les opérations effectuées.
Solution:
0.8 = 0 01110 1001100110
3 = 0 10000 1000000000

Il faut mettre les deux nombres sous le même exposant (2^-1).
0.8 = 1.1001100110 . 2^-1 = 0.011001100110 . 2^-1

+ 1.100000000000
-----------
1.111001100110

1.1110011010 (en arrondi au plus près)

Le résultat est normalisé, et s’écrit: 0 10000 1110011010

3. En partant de leur représentation binaire, effectuez le produit des réels -3 et 0.625 et donnez le résultat en binaire sur 16 bits (en utilisant l’arrondi au plus près, si nécessaire). Détailliez les opérations effectuées.

Solution:
-0.625 = 1 01110 0100000000
3 = 0 10000 1000000000
Signe du résultat: négatif

Additionne les exposants : 1 + (-1) = 0 / 15 + 0 = 15 = 01111

Multiplie les mantisses:
1.01
* 1.1
-----
101
101
-----
1.111000000....

Le résultat s’écrit donc: 1 01111 1110000000

4 Algèbre de Boole

Simplifier les fonctions suivantes en détaillant les étapes :
1. \( f_1(A, B, C, D) = (A + B + C + D) + (\overline{A}.B.C.D) \)
2. \( f_2(A, B, C, D) = A.D + A.C.D + \overline{A}.C.D + \overline{A}.B.C.D \)
3. \( f_3(A, B, C) = A.B.C + A.B.\overline{C} + A.B.C \)
4. \( f_4(A, B, C) = A.B + A.\overline{C} + B.C \)
5 Tables de Karnaugh

(Ci dessous, on notera \( m \) les minterms et \( d \) les "don’t care")

1. En utilisant une table de Karnaugh donnez une forme disjonctive (somme de produits) minimale de \( f(a, b, c, d) = \Sigma m(0, 1, 2, 3, 8, 10) \) ?

Solution: \( f = a.b + b.d \)

2. En utilisant une table de Karnaugh donnez une forme conjonctive (produit de sommes) minimale de \( g(a, b, c, d) = \Sigma m(1, 5, 9, 13, 15) + \Sigma d(2, 3, 10) \) ?

Solution: On pose \( \overline{g} = \Sigma m(0, 4, 6, 7, 8, 11, 12, 14) + \Sigma d(2, 3, 10) \).

\( \overline{g} = \overline{d} + c.\overline{b} \)
Donc $g = d(\overline{c} + a)(\overline{c} + b)$.

3. Proposer un circuit pour la fonction $f$ qui n’utilise que des portes NOR.

**Solution:**

$$f = \overline{a + b + \overline{b} + d}$$