DIRO
IFT 1215

EXAMEN INTRA

Max Mignotte

DIRO, Département d'Informatique et de Recherche Opérationnelle, local 2377
Http : //www.iro.umontreal.ca/∼mignotte/ift1215/
E-mail : mignotte@iro.umontreal.ca

Date : *29/02/2008*

I	Conversion et Arithmétique en Base n (28 pts).
II	Représentation des Données Entières (17 pts).
III	Représentation Flottante (15 pts).
IV	Circuits Logiques (20 pts).
V	Circuits Séquentiels (30 pts).
Total	110 points.

Directives

- Toute documentation est permise
- Toute calculatrice est interdite

◦ Les réponses devront être clairement présentées et justifiées (elles peuvent être concises mais devront néanmoins contenir les résultats intermédiaires nécessaires permettant de montrer sans ambiguïté que vous êtes arrivés au résultat demandé).

◦ Si vous ne comprenez pas une question, faites en une interprétation, et proposez une réponse.

I. Conversion et Arithmétique en Base n (28 pts)

1. Convertir 444_{10} en binaire (base 2). Utiliser le résultat de sa conversion en binaire pour trouver son équivalent octal (base 8) et hexadécimal (base 16).
   <**5 pts**>
2. Convertir 0.11_{10} en base 2, base 8 et base 16 en utilisant comme notation une barre au dessus d'un éventuel motif de plusieurs bits qui se répéterait ou 10 bits après la virgule si ce n'est pas le cas et au moins trois digits après la virgule en base 8 et 16.
   <**5 pts**>
3. En déduire la représentation de 444.11_{10} en base 2, base 8 et base 16.
   <**5 pts**>
4. Convertir le nombre $0.11\,\overline{010}_2$ (i.e., $0.11\,010\,010\,010\ldots$ avec le motif '010' se répétant à l'infinie) en une fraction décimale.
   <**5 pts**>
5. Quel serait l'algorithme qui permettrait, à partir d'une suite de caractères ASCII représentant du texte, et utilisant le code correcteur de parité paire de le transformer en une suite de caractères ASCII avec le code correcteur de parité impaire ?
   <**3 pts**>
6. Représenter $0.12\overline{6}_8$ en une fraction décimale [1].
   <**5 pts**>
   NOTA : Si vous n'y arrivez pas, donner une approximation de cette valeur en base 10 avec trois chiffres significatifs après la virgule (pour la moitié des points de la question).

Réponse

$\boxed{\textbf{1.}}$

$444_{10} = 1\,1011\,1100_2 = 674_8 = 1BC_{16}$.   <**5pts**>

$\boxed{\textbf{2.}}$

$0.11_{10} = 0.0001\ 1100\ 0010_2 = 0.070_8 = 0.1C2_{16}$.   <**5 pts**>

$\boxed{\textbf{3.}}$

$444.11_{10} = 1\ 1011\ 1100\ .0001\ 1100\ 0010_2 = 674.070_8 = 1BC.1C2$ <**5 pts**>

$\boxed{\textbf{4.}}$

▷ 1−**ère méthode** : En utilisant les suites géométriques :

$0.11\,\overline{010}_2$ se traduit en base 10 par la valeur :

$$(2^{-1}+2^{-2})+(2^{-4})\big(1+2^{-3}+2^{-6}\ldots\big) = (2^{-1}+2^{-2})+(2^{-4})\,\frac{1-(2^{-3})^{\infty}}{1-2^{-3}} = \frac{3}{4}+\frac{1}{16}\,\frac{8}{7} = \frac{92}{112} = \frac{23}{28}$$

▷ 2−**ième méthode** : En utilisant la méthode expliquée aussi en cours et utilisant les décalage de bits :

1. Soit $p = 0.11\,\overline{010}_2$. On a

$$\begin{array}{rcl} p = & 0. & 11\,\overline{010}_2 \\ 4\,p = & 11. & \overline{010}_2 \\ 32\,p = & 11010. & \overline{010}_2 \\ \hline \end{array}$$

$$32\,p - 4\,p = \quad 28\,p = \quad 11010_2 - 11_2 = 26_{10} - 3_{10} = 23_{10}$$
$$p = \tfrac{23}{28}$$

<**5 pts**>

5.

Il suffit d'inverser le premier des 8 bits de chaque code ASCII associé a chaque caractère. <**3 pts**>

6.

Pour avoir la valeur exacte sous la forme d'une fraction décimale de $0.12\overline{6}_8$, on peux utiliser les deux méthodes mentionnées à l'exercice 4. La plus simple reste encore celle utilisant les décalage de digit.

$$\begin{array}{rccll}
 & x & = & 0. & 12\,\overline{6}_8 \\
8_{10} & x & = & 1. & 2\overline{6}_8 \\
64_{10} & x & = & 12. & \overline{6}_8 \\
512_{10} & x & = & 126. & \overline{6}_8 \\
\hline
\end{array}$$

$$512\,x - 64\,x = \quad 448\,x = \quad 126_8 - 12_8 = 114_8 = 76_{10}$$

$$x = \tfrac{19}{112} = 0.1696.$$

<**5 pts**>

Sinon, on a besoin de quatre chiffres après la virgule en base 8 pour arriver à la précision demandée en base 10. Dans ce cas, on a :

$$0.12\overline{6}_8 = \frac{1}{8} + \frac{2}{8^2} + \frac{6}{8^3} + \frac{6}{8^4} \approx 0.1694$$

II. Représentation des Entiers Signés (17 pts)

1. Quels sont les avantages de la représentation en complément deux par rapport la représentation en complément un ?
   <**5 pts**>
2. Représenter puis calculer $112 - 123$ sur 8 bits en complément à un (complément logique ou restreint) puis en complément à deux (complément arithmétique ou vrai) en indiquant à chaque fois lorsqu' apparaît un débordement (dépassement de capacité) ou une retenue ainsi que le résultat en base 10 obtenu.
   <**8 pts**>
3. À quoi correspond l'expression (i.e., le calcul) suivante en base 10 ?
   <**2 pts**>

   $$-0111\,0100 - 0111\,0010$$

   exprimées en complément à deux et faites cette opération (calcul sur 8 bits en complément à deux) en indiquant lorsqu'apparaît un débordement ou une retenue.
   <**2 pts**>

Réponse

$\boxed{\textbf{1.}}$

Un seul zéro (donc on peut représenter un nombre supplémentaire) et une retenue (si le nombre de bit du résultat est supérieur au nombre de bits des deux nombres) que l'on se débarrasse (contrairement la représentation en complément a un que l'on doit ajouter au résultat). Deux modifications qui permet de simplifier l'architecture du système informatique qui l'utilise.

<**5 pts**>

$\boxed{\textbf{2.}}$

Sur 8 bits, $112_{10} = 0111\,0000_2$ et $-123_{10} = -0111\,1011_2$

En complément à un (sur 8 bits) on a $112_{10} = 0111\,0000_2$ et $-123_{10} = 1000\,0100_2$ et l'opération suivante,

$$0111\,0000 + 1000\,0100_2 = 1111\,0100 = -0000\,1011_2. \quad < \textbf{3 pts} >$$

On obtient donc un chiffre négatif (premier des 8 bits a "1") et finalement une valeur égale à -11_{10}.
<**1 pt**>

En complément à deux (sur 8 bits) on a $112_{10} = 0111\,0000_2$ et $-123_{10} = -0111\,1011_2 = 1000\,0101$ et l'opération suivante,

$$0111\,0000_2 + 1000\,0101_2 = 1111\,0101 = -(0000\,1010 + 1_2) = -(0000\,1011_2) \quad < \textbf{3 pts} >$$

On obtient donc un chiffre négatif (premier des 8 bits a "1") et finalement une valeur égale à -11_{10}.
<**1 pt**>

$\boxed{\textbf{3.}}$

Le premier chiffre est égale à -116_{10} En complément à deux, sa magnitude est $1000\,1011_2 + 1_2 = 1000\,1100_2$.

Le deuxième est égale à -114_{10}. En complément à deux, sa magnitude est $1000\,1101_2 + 1_2 = 1000\,1110_2$.

Donc on a l'opération $-116_{10} - 114_{10} = -230_{10}$. <**2 pts**>

et on a en base 2, et complément à deux (sur 8 bit), on calcul donc l'opération suivante,

$$1000\,1100_2 + 1000\,1110_2 = 1\,0001\,1010_2$$

Il y a une retenue, non prise en compte, mais aussi un débordement de capacité car le premier et deuxième chiffre sont négatifs et le résultats est positif.

<**2 pts**>

III. Représentation Flottante (15 pts)

1. Rappeler (i.e. énumérer) quels sont les différents types de problèmes ou d'erreurs numériques existants lorsqu'on fait des calculs en notation flottantes sur une calculatrice ou sur un ordinateur.
<**5 pts**>

2. Avec une représentation flottante binaire avec un format du type $\pm 0.xxxx\ldots$ (avec 23 x, i.e , 23 bits pour la mantisse), un bit de signe ("0" pour le signe "+") et un exposant sur 8 bits permettant d'exprimer les valeurs de l'exposant de -126 à 127 (l'exposant égale à -127 et 128 permettant d'exprimer des valeurs spéciales) et biais de 127 et une notation flottante utilisant les valeurs arrondies et la disposition suivante [bit de signe | 23 bits mantisse | 8 bits de l'exposant] ;

(a) Donner en binaire la représentation de la plus petite valeur positive représentable par cette représentation flottante. Convertir cette valeur binaire en valeur décimale.
<**5 pts**>

(b) Souvenez vous du problème de la défaillance du missile US patriote présenté en cours (Chapitre Système de nombre, slide 15). Expliquer ce qui ce serait passé si l'informaticien en charge de ce projet avait décidé que le temps de l'horloge interne du missile US devait être mesuré en 1/16 de seconde (i.e., 16 coups d'horloge par seconde plutôt que dix coups d'horloge par seconde).
<**5 pts**>

Réponse

1.

Le premier type de problème vient du fait qu'il y a une infinité de nombres (et pas seulement les nombres a virgules) sans représentation exacte en notation flottante. De ce fait les opérations impliquant ses nombres seront forcément approximés (erreur d'*affectation* i.e., de troncature ou d'arrondi).

Le deuxième problème (qui découle du premier) est lié a l'*underflow* et a l'*overflow*, i.e., l'impossibilité de représenter un nombre plus petit que l'epsilon machine ou un nombre plus grand que celui associé a tous les bits de la mantisse égale a un et avec un exposant le plus grand possible.

Le troisième type d'erreur numérique vient de l'erreur de décalage lorsqu'on additionne (ou soustrait) deux nombres flottants d'ordre de grandeur différente.

<**5 pts**>

2.(a)

La plus petite valeur représentable par cette représentation est de la forme $0.10\ldots 23$ zéros$\ldots \times 2^{-126}$ <**2.5 pts**> qui se traduit en décimal par la valeur 2^{-127} (que l'on appelle *epsilon machine*). <**2.5 pts**>

2.(b)

Dans ce cas, la valeur représenté dans le registre flottant simple précision est $1/16 = 2^{-4}$ (et non plus 1/10) qui a cette fois une représentation exacte dans ce format. Du coup, il n'y aurait eu aucune erreur d'accumuler lorsqu'on aurait mesuré le temps de l'horloge interne du missile US patriote (si ce n'est le facteur multiplicatif de ce nombre qui n'aurait pas toujours eu une représentation exacte).
<**5 pts**>

IV. Circuits Logiques (20 pts)

On dispose sur une automobile de quatre commandes indépendantes : C_V pour la commande des veilleuses, C_C pour la commande des deux phares de croisement, C_R pour les deux phares de route et finalement C_A pour les deux phares antibrouillard (valeur 1 au travail (i.e., lorsque la commande est active) et valeur 0 au repos).

On note de la façon suivante, les états des lumières (valeur 1 a l'allumage et 0 pour l'extinction).
— V pour les veilleuses
— C pour les feux de croisement
— R pour les feux de route
— A pour les feux antibrouillard

Les veilleuses n'étant pas considérées comme des phares, il est précisé que

- 4 phares ne peuvent être allumés simultanément
- Les feux de **croisement** ont priorité sur les feux de route et sur les antibrouillards (ce qui veux dire que si on allume les feux de croisements et les feux de route, seule les feux de croisement s'allumeront. De même, si on commande l'allumage des trois feux, seule les feux de croisement s'allumeront).
- Les feux **antibrouillard** ont priorité sur les feux de route (ce qui veux dire que si on commande l'allumage des feux antibrouillards et les feux de route, seule les feux antibrouillard s'allumeront).
- Les **veilleuses** peuvent être allumées seules mais l'allumage des autres feux (croisement, route et antibrouillard) entraîne obligatoirement l'allumage des veilleuses.

1. Construire la table de vérité de V, C, R et A en fonction des quatre variables Booléennes C_V, C_C, C_R, C_A.
   <10 pts>
2. En utilisant les tables de Karnaugh (ou les propriétés de l'algèbre Booléenne selon votre goût) simplifier le plus possible les fonctions V, C, R et A sous la forme d'une **SOP**.
   <10 pts>

Réponse

$\boxed{\textbf{1.}}$

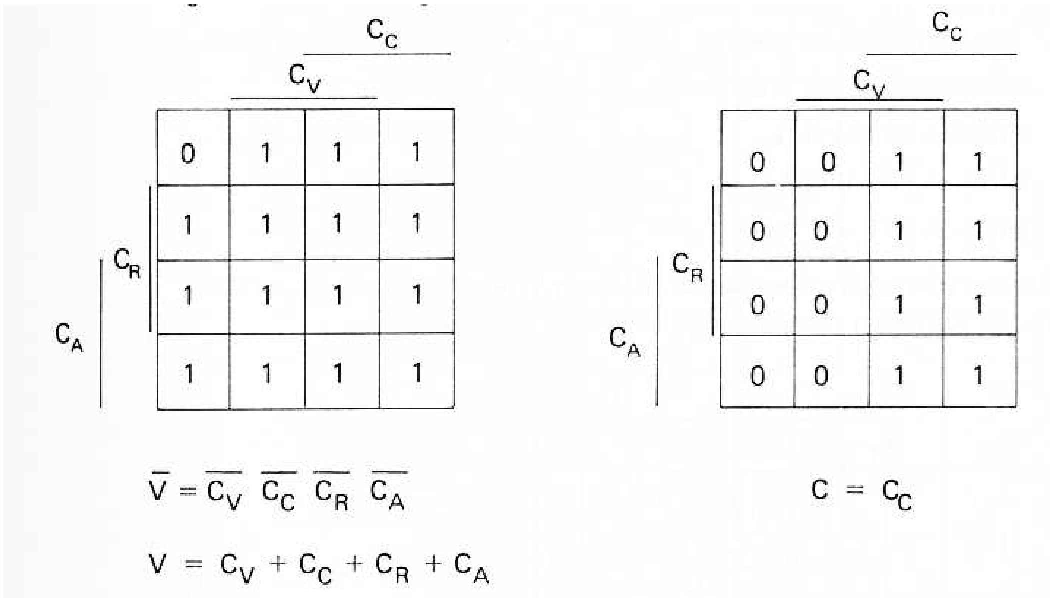
Table de Vérité :

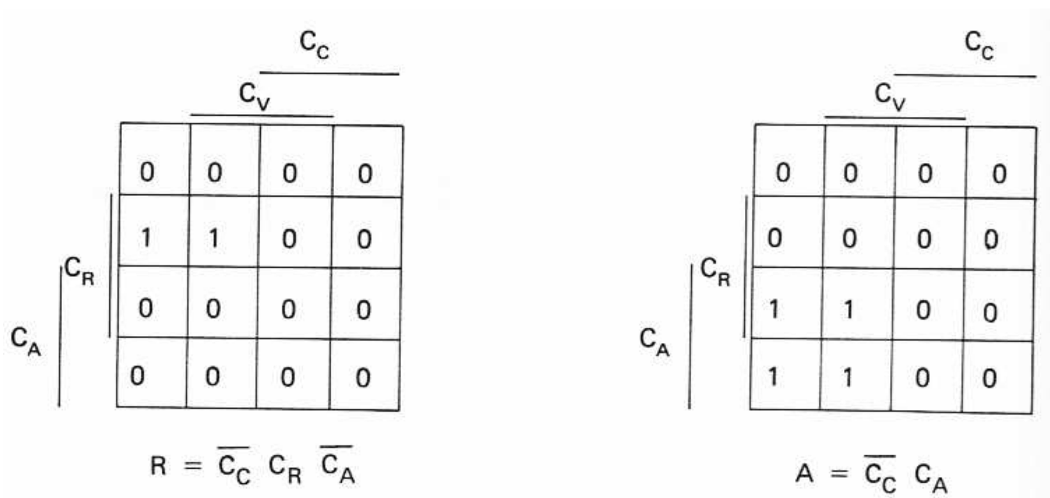
C_V	C_C	C_R	C_A	V	C	R	A	n°
0	0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	1	1
0	0	1	0	1	0	1	0	2
0	0	1	1	1	0	0	1	3
0	1	0	0	1	1	0	0	4
0	1	0	1	1	1	0	0	5
0	1	1	0	1	1	0	0	6
0	1	1	1	1	1	0	0	7
1	0	0	0	1	0	0	0	8
1	0	0	1	1	0	0	1	9
1	0	1	0	1	0	1	0	10
1	0	1	1	1	0	0	1	11
1	1	0	0	1	1	0	0	12
1	1	0	1	1	1	0	0	13
1	1	1	0	1	1	0	0	14
1	1	1	1	1	1	0	0	15

<10 pts>

$\boxed{2.}$

Tableaux de Karnaugh :





<10 pts>

V. Circuits Séquentiels (30 pts)

On veut réaliser le dispositif logique séquentiel qui permette de détecter le motif "1101".. De telle façon que pour une entrée du type (donnée ci dessous), on génère la sortie (donnée ci dessous) suivante

Entré	1	0	0	1	0	0	**1**	**1**	**0**	**1**	0	**1**	**1**	**0**	**1**	**1**	**0**	**1**	0	0	**1**	**1**	**0**	**1**	0	0	1
Sortie	0	0	0	0	0	0	0	0	0	**1**	0	0	0	0	**1**	0	0	**1**	0	0	0	0	0	**1**	0	0	0

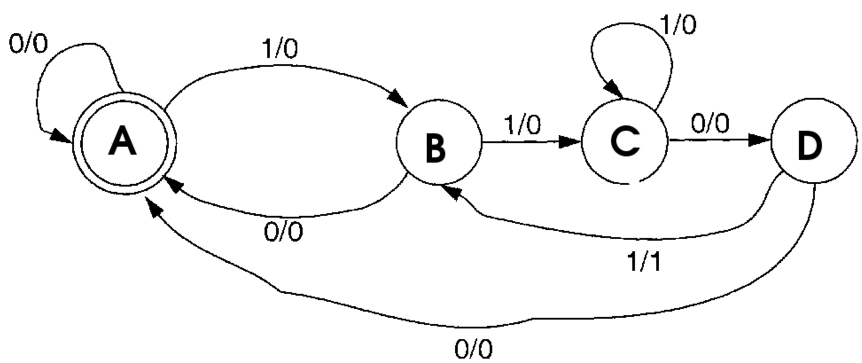
1. Dessiner le diagramme de transition d'états de ce système[1].
   <**10 pts**>
2. Donner ensuite la table d'état de ce système.
   <**3 pts**>
3. Établir la table de vérité de ce système.
   <**5 pts**>
4. Donner les équations logiques (**simplifiées au maximum**) des signaux de sortie et des états futurs de ce système.
   <**6 pts**>.
5. Donner le logigramme de ce système en utilisant des bascules D.
   <**6 pts**>

Réponse

1.

Graphe de Transition d'États :

Dans ce diagramme d'états, A est l'état initial, B déigne l'état pour lequel le premier symbole est détecté (1), C quand le motif 11 est détecté et enfin D quand le motif 110 est détecté.



<**10 pts**>

2.

Table d'États :

Input / Etat present	0	1
A : 00	A: 00 / 0	B : 01/ 0
B : 01	A : 00/ 0	C : 10/ 0
C : 10	D : 11/ 0	C : 10/ 0
D : 11	A : 00/ 0	B : 01/ 1

<**3 pts**>

Table de Vérité :

1. On utilisera sur les flèches de transition d'états, la convention suivante Entrée/Sortie, i.e., *Input/Output* du style 1/0 indiquant que l'on a un 1 en entré et un 0 en sortie.

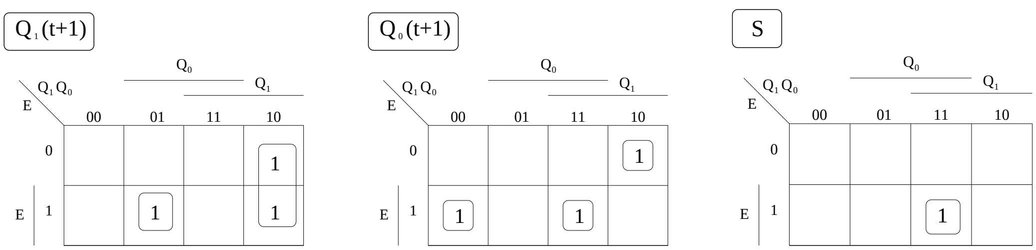
$q_1(t)$	$q_0(t)$	E	$q_1(t+1)$	$q_0(t+1)$	S
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	0
0	1	1	1	0	0
1	0	0	1	1	0
1	0	1	1	0	0
1	1	0	0	0	0
1	1	1	0	1	1

<5 pts>

$\boxed{\textbf{4.}}$

Équations Logiques :

On trouve, à partir de la table de vérité, et se servant des tableaux de Karnaugh, les équations logiques suivantes, simplifiées au maximum,

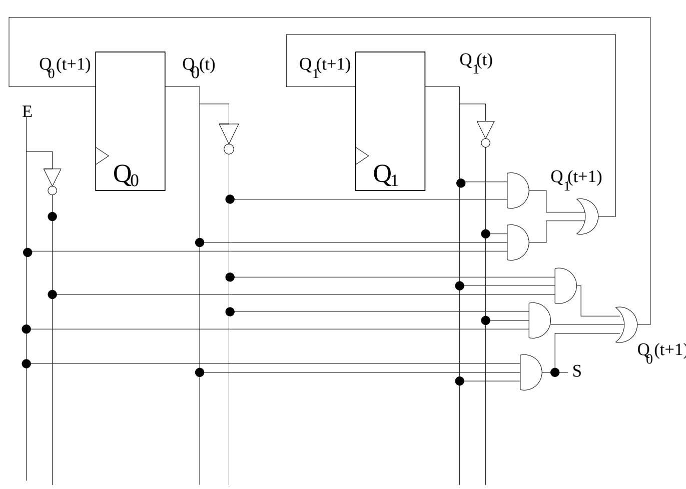


$$
\begin{aligned}
S &= E\,Q_0(t)\,Q_1(t) \\
Q_0(t+1) &= \overline{Q}_1(t)\,\overline{Q}_0(t)\,E + Q_1(t)\,Q_0(t)\,E + Q_1(t)\,\overline{Q}_0(t)\,\overline{E} \\
Q_1(t+1) &= Q_1(t)\,\overline{Q}_0(t) + \overline{Q}_1(t)\,Q_0(t)\,E
\end{aligned}
$$

<6 pts>

$\boxed{\textbf{5.}}$

On en déduit le schéma de notre circuit séquentiel :



<6 pts>