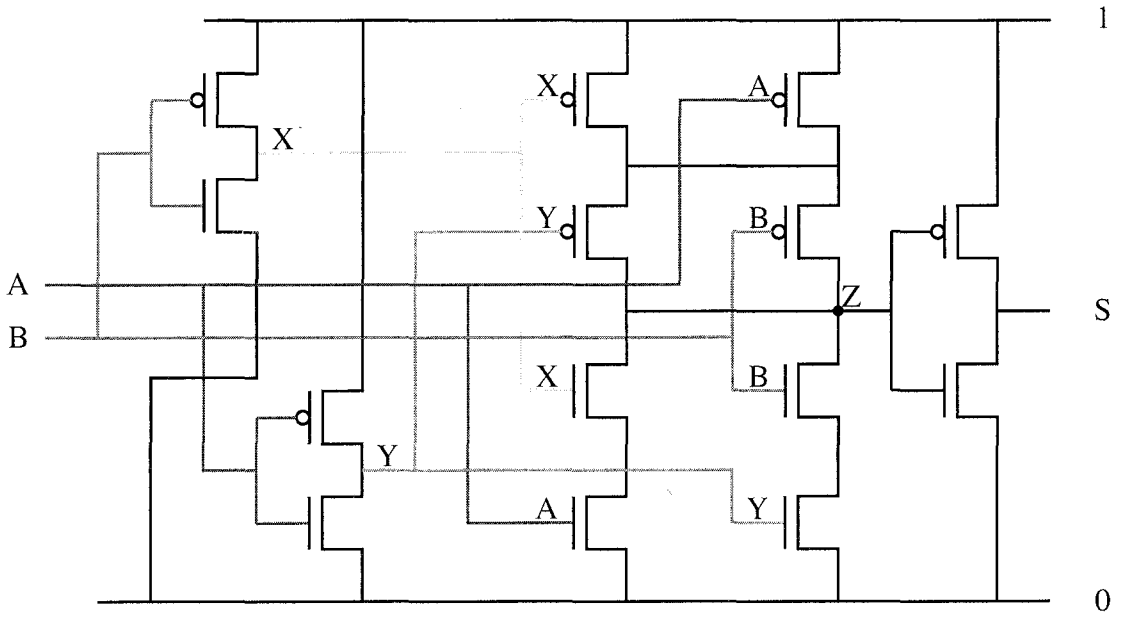
Examen
Circuits et architecture des ordinateurs

— Master d'informatique —

Janvier 2008, durée 2h30.

L'examen se compose de quatre exercices indépendants. Les documents sont interdits à l'exception d'une feuille de mémento.

► **Exercice 1** On considère le circuit suivant



réalisé avec des transistors CMOS.

a) Remplir le tableau suivant (à recopier) qui donne la valeur de X, Y, Z et la sortie S en fonction des deux entrées A et B.

A	B	X	Y	Z	S
0	0	?	?	?	?
⋮	⋮	⋮	⋮	⋮	⋮

b) Donner la fonction réalisée par ce circuit.

► **Exercice 2** L'objetif de cet exercice est d'écrire des petits programmes en assembleur LC-3. Avant d'écrire ces programmes, il est impératif de décrire l'algorithme utilisé. Tout programme donné sans explication ne sera pas corrigé. On désire réaliser des fonctions de comparaison pour des entiers signés et non signés.

a) Écrire une sous-routine `cmp` qui prend deux entiers m et n dans les registres R0 et R1 et qui retourne dans R0 la valeur z donnée par

$$z = \begin{cases} -1 & \text{si } m < n, \\ 0 & \text{si } m = n, \\ 1 & \text{si } m > n. \end{cases}$$

Pour cette première fonction on pourra supposer que les deux entiers m et n sont positifs et vérifient $0 \leq m, n \leq 2^{15} - 1$. Les contenus des registres peuvent être modifiés.

b) Écrite une sous-routine `cmps` semblable à la routine `cmp` mais qui fonctionne quand les deux entiers m et n sont signés et codés en compléments à 2 sur 16 bits ($-2^{15} \leq m, n \leq 2^{15} - 1$).

c) Écrite une sous-routine `cmpp` semblable à la routine `cmps` qui fonctionne quand les deux entiers m et n sont non signés et vérifient $0 \leq m, n \leq 2^{16} - 1$.

d) Comment faire pour préserver les registres autres que R0.

► **Exercice 3** On considère le programme suivant écrit en assembleur LC-3.

```
        .ORIG x3000
mystere:LD R0,valeur
        JSR myst
        NOP
valeur: .FILL 11    ;   1011 en binaire

myst:   AND R1,R1,0
        ADD R1,R1,-1
        ADD R2,R1,-1
        AND R0,R0,R0
        BRzp m0
m1:     AND R1,R1,R2
m0:     AND R2,R2,R2
        BRp fini
        ADD R2,R2,R2
        ADD R2,R2,1
        ADD R0,R0,R0
        BRn m1
        BR m0
fini:   NOT R0,R1
        RET
        .END
```

a) Quelle est le contenu en binaire du registre R0 lorsque le programme atteint l'instruction `NOP` ? Il est conseillé de faire les calculs en binaire.

b) Quelle serait le contenu en binaire du registre R0 lorsque le programme atteint l'instruction `NOP` si la valeur chargée dans le registre R0 à la première instruction était respectivement 2^{15}, 1 et $2^{15} + 1$.

c) Que calcule la routine `myst` ?

► **Exercice 4** On considère le cache d'un processeur utilisant des adresses sur 16 bits. On suppose que le cache est constitué de 16 lignes de 16 octets. On suppose qu'au départ le cache est vide, c'est-à-dire que toutes les lignes cachent les octets à l'adresse 0000. Le processeur accède successivement aux adresses suivantes : 3000, 2040, 3001, 2404, 3002, 20C4, 3003 et 24C4. Dans les deux cas suivants, donner les éléments correspondant à chaque accès : défaut ou succès, ligne du cache concernée, chargement ou non. On suppose pour simplifier que tous les accès sont des lectures. On note $b_{15}b_{14}\ldots b_2b_1b_0$ les 16 bits d'une adresse.

a) On suppose que le cache est direct. La ligne est déterminée par les 4 bits $b_7b_6b_5b_4$ de chaque adresse.

b) On suppose que le cache est 2-associatif. Chaque ligne i du cache est associée avec la ligne $i + 8$ pour $0 \leq i \leq 7$. La paire de lignes est déterminée par les 3 bits $b_6b_5b_4$ des adresses.