

Examen

Circuits et architecture des ordinateurs

— Master d'informatique —

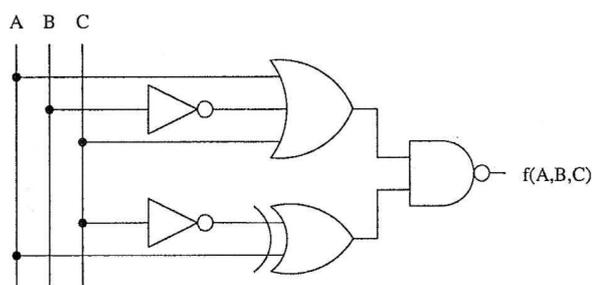
Janvier 2009, durée 2h30.

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

► **Exercice 1** On considère des nombres codés en compléments à 2 sur 4 bits. Remplir le tableau suivant en donnant pour chaque entrée la somme calculée par un additionneur 4 bits (écrite en binaire et en décimal) et si la valeur est exacte (débordement).

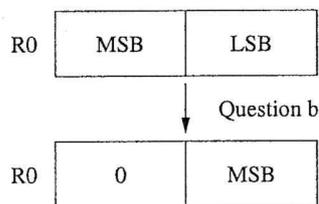
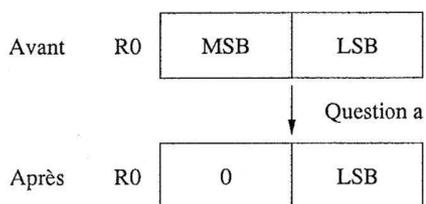
+	1011	1101	0011	0101
1011				
1101				
0011				
0101				

► **Exercice 2**



- Donner la table de vérité de la fonction f réalisée par le circuit ci-dessus.
- Donner un circuit équivalent en utilisant uniquement des portes NOT, AND et OR.
- Donner un circuit équivalent en utilisant uniquement des portes NAND.

► **Exercice 3** L'objectif 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 manipuler des chaînes de caractères codés sur 8 bits en mettant deux caractères dans chaque mot de la mémoire du LC-3. Il est alors nécessaire de savoir extraire chacun des caractères d'un mot mémoire.



- Écrire une routine qui extrait l'octet de poids faible d'un mot mémoire. Cette routine prend un mot mémoire dans R0 et retourne l'octet de poids faible (LSB) dans R0.
- Écrire une routine qui extrait l'octet de poids fort d'un mot mémoire. Cette routine prend un mot mémoire dans R0 et retourne l'octet de poids fort (MSB) dans R0.

► **Exercice 4** On considère le programme suivant écrit en assembleur LC-3.

```

        .ORIG x3000
mystere:LD R0,m
        JSR myst
        TRAP x25                ; HALT
m:      .FILL 18                ; Valeur en décimal
myst:   AND R1,R1,0
        AND R2,R2,0
        LD R3,cst16
        AND R0,R0,R0
        BRzp bit0
bit1:   ADD R1,R1,1
bit0:   ADD R1,R1,-3
        BRn neg
        ADD R2,R2,1
        BR suite
neg:    ADD R1,R1,3
suite:  ADD R3,R3,-1
        BRz fini
        ADD R1,R1,R1
        ADD R2,R2,R2
        ADD R0,R0,R0
        BRn bit1
        BR bit0
fini:   ADD R0,R2,0
        RET
cst16:  .FILL 16
        .END

```

- Quel est le contenu en binaire du registre R0 lorsque le programme atteint l'instruction TRAP ? Il est conseillé de faire les calculs en binaire puis de traduire ensuite en décimal.
- Quel serait le contenu du registre R0 lorsque le programme atteint l'instruction TRAP si la valeur chargée dans le registre R0 à la première instruction était respectivement 0, 3 et 12.
- Que calcule la routine myst quand la valeur contenue au départ dans R0 est positive ?

► **Exercice 5** On désire ajouter deux instructions PUSH et POP permettant au processeur LC-3 d'empiler et de dépiler des registres de la pile. Ces deux instructions utilisent l'opcode 1101.

- On suppose que les instructions PUSH et POP ont la syntaxe PUSH Ri et POP Ri où Ri désigne un des registres généraux. Un seul registre est donc empilé ou dépilé par l'instruction. Donner un codage de ces deux instructions qui soit cohérent avec les codages des autres instructions.
- On suppose que les deux instructions ont la syntaxe PUSH Ri, ..., Rk et POP Ri, ..., Rk où Ri, ..., Rk désigne une liste dans l'ordre et sans répétition de registres généraux comme R1, R4, R5 ou R0, R2, R4, R7. Plusieurs registres sont donc empilés ou dépilés par l'instruction. Donner un codage de ces deux instructions.
- Expliquer laquelle de ces deux instructions est préférable et pourquoi.