

CEG 3555: Systèmes Numériques II  
(Automne 2023)  
Prof. Rami Abielmona  
Devoir #1: *Optimisation de Fonction Logique*

Septembre 11, 2023

## Objectif and Date Due

L'objectif de ce devoir est de démontrer une compréhension d'optimisations de fonctions logiques. La date due de ce devoir est **Lundi 2 Octobre, 2023 à 20:00**, et doit être soumis en ligne.

## Question I

Cette question se concerne avec l'optimisation de fonction logique en utilisant la manipulation algébrique. Les parties a, b, c et d sont questions 2.10, 2.11, 2.54 et 2.56 de votre manuel.

### Partie a

Utilisez la manipulation algébrique pour démontrer que pour trois variables d'entrées  $x_1, x_2$  et  $x_3$

$$\sum m(1, 2, 3, 4, 5, 6, 7) = x_1 + x_2 + x_3 \quad (1)$$

### Partie b

Utilisez la manipulation algébrique pour démontrer que pour trois variables d'entrées  $x_1, x_2$  et  $x_3$

$$\prod m(0, 1, 2, 3, 4, 5, 6) = x_1 \cdot x_2 \cdot x_3 \quad (2)$$

### Partie c

Concevez le circuit le plus simple qui réalise la fonction  $f(x_1, x_2, x_3) = \sum m(3, 4, 6, 7)$  en utilisant des portes **NON-ET** seulement.

### Partie d

Concevez le circuit le plus simple qui réalise la fonction  $f(x_1, x_2, x_3) = \sum m(3, 4, 6, 7)$  en utilisant des portes **NON-OU** seulement.

### Partie e

Déterminez si les fonctions booléennes suivantes sont égales ou non:

$$f(x, y, z) = x \cdot y + y \cdot z + \bar{x} \cdot z + \bar{x} \cdot \bar{y} \quad (3)$$

$$g(x, y, z) = x \cdot y + \bar{x} \cdot \bar{y} + \bar{x} \cdot y \cdot z \quad (4)$$

## Question II

Cette question se concerne avec la synthèse à sorties et niveaux multiples. Revisitez section 8.1 et 8.2 de votre manuel pour vous aider. Les parties a et b sont questions 8.7 et 8.9 de votre manuel.

### Partie a

Trouvez la réalisation la plus simple pour la fonction  $f(x_1, \dots, x_4) = \sum m(0, 3, 4, 7, 9, 10, 13, 14)$ , supposant que les portes logiques ont un fan-in maximum de deux.

### Partie b

Utilisez la décomposition fonctionnelle pour trouver l'implémentation la plus meilleure pour la fonction  $f(x_1, \dots, x_5) = \sum m(1, 2, 7, 9, 10, 18, 19, 25, 31) + d(0, 15, 20, 26)$ . Comment votre réalisation compare-t-elle avec la réalisation SOP la moins chère ? Donnez les coûts.

### Partie c

Utilisez une carte de Karnaugh pour minimiser la fonction booléenne suivante

$$f(a, b, c, d, e) = \sum m(4, 5, 10, 11, 15, 18, 20, 24, 26, 30, 31) + d(9, 12, 14, 16, 19, 21, 25) \quad (5)$$

## Partie d

Considérez les fonctions suivantes:

$$f_1 = \sum m(0, 2, 4, 5, 9, 10, 11, 13, 15) \quad (6)$$

$$f_2 = \sum m(2, 5, 10, 11, 12, 13, 14, 15) \quad (7)$$

$$f_3 = \sum m(0, 2, 3, 4, 9, 11, 13, 14, 15) \quad (8)$$

Minimisez les fonctions et réalisez-les dans un circuit commun le moins cher.

## Question III

Cette question se concerne avec les composants de circuit combinational. Les parties a, b et c sont questions 4.7, 4.8 et 4.23 de votre manuel.

### Partie a

Considérez la fonction  $f = \overline{w_2} + \overline{w_1} \cdot \overline{w_3} + w_1 \cdot w_3$ . Démontrez comment l'application répétée de l'expansion de Shannon peut être employée pour dériver les mintermes de  $f$ .

### Partie b

Répétez la partie a, mais pour  $f = w_2 + \overline{w_1} \cdot \overline{w_3}$ .

### Partie c

Dérivez le circuit d'un 8-à-3 encodeur de priorité.

### Partie d

Réalisez les fonctions suivantes en utilisant un 3-à-8 décodeur:

$$f_1(a, b, c) = a \cdot b + \overline{a} \cdot \overline{b} \cdot \overline{c} \quad (9)$$

$$f_2(a, b, c) = \overline{a} \cdot b + a \cdot \overline{b} \quad (10)$$

### Partie e

Réalisez un décodeur 6-à-64 en utilisant quatre décodeurs 4-à-16 et un décodeur 2-à-4. Utilisez  $A_5 A_4 A_3 A_2 A_1 A_0$  comme données et  $D_0 - D_{63}$  comme sorties. N'oubliez pas le signal *Enable* pour chaque décodeur.

## Question IV

Cette question se concerne avec le VHDL au niveau structurel. Les parties a, b et c sont question 4.26, 4.27 et 4.29 de votre manuel.

### Partie a

Créez une entité en VHDL appelé `if2à4` qui représente un décodeur 2-à-4 binaire, en utilisant une expression `if-then-else`. Donnez le code de VHDL au niveau comportemental. Créez une seconde entité en VHDL appelé `h3à8` qui représente le décodeur 3-à-8 binaire démontré dans la figure 4.15, en utilisant deux exemples de l'entité `if2à4`. Donnez le code de VHDL au niveau structurel.

### Partie b

Créez une entité en VHDL appelé `h6à64` qui représente un décodeur binaire 6-à-64. Employez la structure *comme-arbre (treelike)* sur le schéma 4.16, dans lequel le décodeur 6-à-64 est construit en utilisant cinq exemples du décodeur `h3à8` créé dans le problème dans la partie a. Donnez le code de VHDL au niveau structurel.

### Partie c

Concevez un circuit de décalage, semblable à celui sur le schéma 4.51, qui peut décaler un vecteur de quatre bits d'entrée,  $W = w_3w_2w_1w_0$ , une position d'un bit vers la droite quand le signe de commande *Droite* est égal à 1, et une position d'un bit vers la gauche quand le signal de commande *Gauche* est égal à 1. Quand tous les deux sont 0 ( $Droite = Gauche = 0$ ), le rendement du circuit devrait être identique que le vecteur d'entrée. Supposez que la condition  $Droite = Gauche = 1$  ne se produira jamais.

## Question V

Cette question se concerne avec la méthode d'ASM. Revisez les leçons et le laboratoire #1 pour avoir une idée plus profonde sur cette technique de concevoir des circuit numériques systématiquement.

Un compteur à quatre bits doit être conçu pour compter dans l'ordre suivant:  $0000 \rightarrow 0001 \rightarrow 0011 \rightarrow 0010 \rightarrow 0110 \rightarrow 0111 \rightarrow 0101 \rightarrow 0100 \rightarrow 1100 \rightarrow 1101 \rightarrow 1111 \rightarrow 1110 \rightarrow 0000 \rightarrow \dots$ . Le compteur commence à l'état 0000 et compte quand le signal de commande *COMMENCE* est allumé.

### Partie a

Fournissez le **pseudo-code** de ce compteur.

## Partie b

En utilisant le pseudo-code ci-dessus, votre tâche est de concevoir et démontrer le diagramme d'ASM (en format graphique), le datapath, le diagramme détaillé d'ASM (en format graphique) et le controlpath réalisé avec la méthode d'une bascule-par-état. Notez que vous pouvez prendre la sortie du compteur comme la sortie d'un ensemble de registres à 1-bit.

## Question de Boni

Montrez que si deux nombres du complément de 2 sont ajoutés, le bit de débordement est le EX-OR du report-donnée (carry-in) et report-sortie (carry-out) du bit de poids fort (MSB).