

Nom : \_\_\_\_\_ Matricule : \_\_\_\_\_

INF3500 : Conception et réalisation de systèmes numériques

Examen intra #1 – jeudi 2 octobre 2014

Durée: 1 heure.

Documentation: Une feuille recto verso 8.5"×11" ou A4 permise.

Pondération: 10%.

Calculatrice: Programmable permise.

Directives particulières:

- Ordinateurs interdits. Appareils mobiles interdits.
- Répondre à toutes les questions, la valeur de chaque question est indiquée.
- Répondre sur le questionnaire et le remettre.
- Ne posez pas de questions. En cas de doute sur le sens d'une question, énoncez clairement toute supposition que vous faites.

---

**Question 1. (1 point) Réponses brèves**

a. (0.25 point) Nommez deux facteurs qui affectent le taux de traitement (en nombre de résultats produits par seconde) d'un système numérique.

---

---

---

---

---

---

b. (0.25 point) Nommez deux contraintes données à l'outil de synthèse pour le guider dans son travail.

---

---

---

---

---

---

c. (0.25 point) Vrai ou faux : une mémoire PROM est moins efficace, en termes du pourcentage du circuit utilisé, qu'un circuit PLA pour implémenter la plupart des fonctions logiques. Justifiez votre réponse.

---

---

---

---

---

---

d. (0.25 point) Nommez deux des quatre composantes principales d'une tranche d'un FPGA Virtex 5.

---

---

---

---

---

---



**Question 3. (3 points)**

Un système de contrôle de la température ambiante d'une pièce reçoit une entrée d'un thermomètre numérique et a deux sorties, une pour activer la climatisation et une pour activer le chauffage. Le thermomètre numérique exprime les températures selon un code Gray sur 4 bits, selon le tableau suivant. On vise à maintenir une température entre 20°C et 24°C. Le système de contrôle est très simple : le climatiseur doit être activé quand il fait trop chaud, et la fournaise doit être activée quand il fait trop froid.

Donnez une architecture en VHDL synthétisable d'un module combinatoire pour implémenter le contrôleur de température qui correspond à la définition d'entité suivante.

T3	T2	T1	T0	Température	
0	0	0	0	$T \leq 0^\circ \text{C}$	<pre> library ieee; use ieee.std_logic_1164.all; use ieee.numeric_std.all;  entity temperature is port (     temperature : in unsigned(3 downto 0);     climatisation, chauffage: out std_logic ); end temperature; </pre>
0	0	0	1	$0^\circ \text{C} < T \leq 2^\circ \text{C}$	
0	0	1	1	$2^\circ \text{C} < T \leq 4^\circ \text{C}$	
0	0	1	0	$4^\circ \text{C} < T \leq 6^\circ \text{C}$	
0	1	1	0	$6^\circ \text{C} < T \leq 8^\circ \text{C}$	
0	1	1	1	$8^\circ \text{C} < T \leq 10^\circ \text{C}$	
0	1	0	1	$10^\circ \text{C} < T \leq 12^\circ \text{C}$	
0	1	0	0	$12^\circ \text{C} < T \leq 14^\circ \text{C}$	
1	1	0	0	$14^\circ \text{C} < T \leq 16^\circ \text{C}$	
1	1	0	1	$16^\circ \text{C} < T \leq 18^\circ \text{C}$	
1	1	1	1	$18^\circ \text{C} < T \leq 20^\circ \text{C}$	
1	1	1	0	$20^\circ \text{C} < T \leq 22^\circ \text{C}$	
1	0	1	0	$22^\circ \text{C} < T \leq 24^\circ \text{C}$	
1	0	1	1	$24^\circ \text{C} < T \leq 26^\circ \text{C}$	
1	0	0	1	$26^\circ \text{C} < T \leq 28^\circ \text{C}$	
1	0	0	0	$28^\circ \text{C} < T$	

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

(Utilisez le verso si nécessaire)

**Question 4. (3 points)**

Considérez le code VHDL suivant.

```
library ieee;
use ieee.std_logic_1164.all;

entity module8 is
  port (
    A, B, C, D, E: in std_logic;
    X, Y, Z : out std_logic
  );
end module8;

architecture arch of module8 is
begin

  X <= (not(A) and B and C) or (D and not(E));

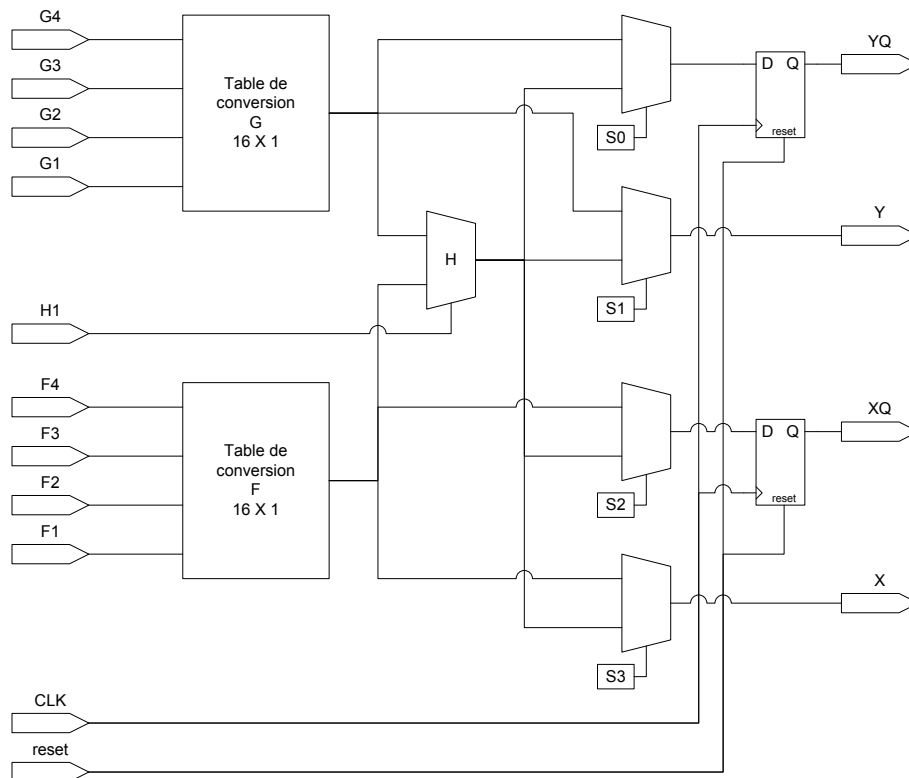
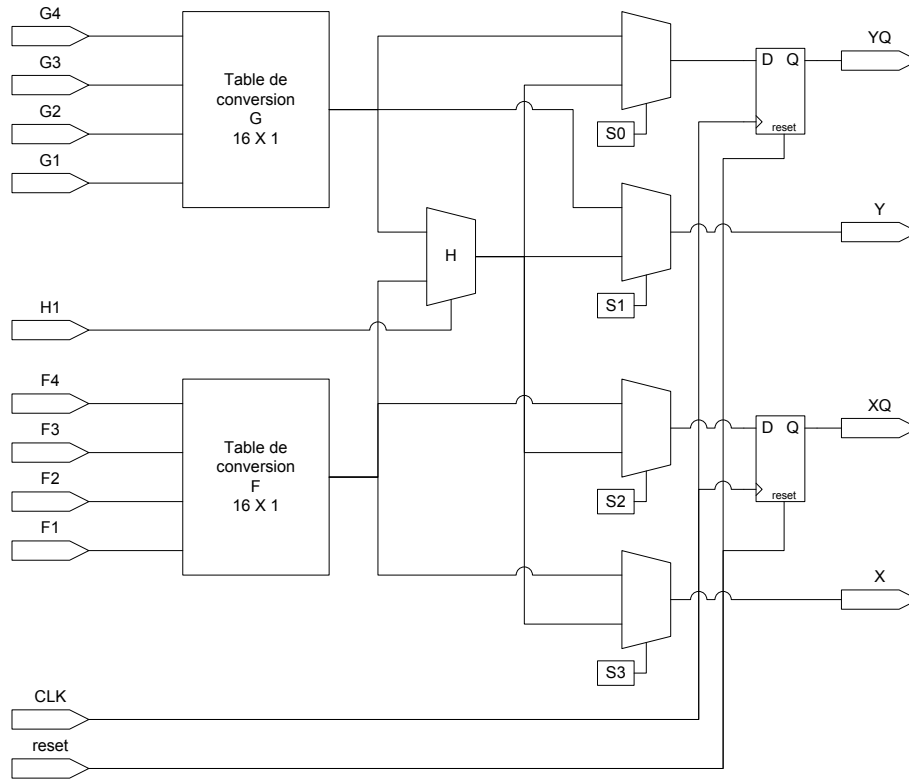
  Y <= '1' when (A = '1' and B = '1' and D = '0') else '0';

  process(B, C, D, E) is
  variable t1, t2 : std_logic;
  begin
    t1 := B and C;
    t2 := D or E;
    Z <= t1 and t2;
  end process;

end arch;
```

Montrez, sur le diagramme qui suit, un résultat possible de la synthèse et de l'implémentation de ce code sur un FPGA. Indiquez directement sur le dessin où chaque signal se situe ainsi que les interconnexions entre les blocs.

Indiquez dans les tables de vérité le contenu de chacune des tables de conversion que vous utilisez.



G4 ()	G3 ()	G2 ()	G1 ()	G ()
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

F4 ()	F3 ()	F2 ()	F1 ()	F ()
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

G4 ()	G3 ()	G2 ()	G1 ()	G ()
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

F4 ()	F3 ()	F2 ()	F1 ()	F ()
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

## Solutions

1. a. La fréquence d'horloge, le nombre de cycles d'horloge requis par résultat, le nombre de calculs effectués en parallèle
- b. Famille et type de puce utilisée, fréquence d'horloge cible, emphase sur la latence de calcul ou l'espace utilisé
- c. La PROM est moins efficace que le PLA en général, parce que la PROM décode tous les mintermes de ses entrées. En général on n'utilise pas tous les mintermes. Le PLA permet de choisir quels mintermes sont générés.
- d. Tables de correspondance, portes logiques pour l'addition rapide, multiplexeurs pour router les signaux et éléments à mémoire.

2.

## A. Analyse des coûts

Quoi	FPGA	Logique fixe
Main d'oeuvre	$3 \times 4/12 \times \$100K / \text{an} = \$100K$	$3 \times 6 / 12 \times \$100K / \text{an} = \$150K$
Licenses	\$5K	\$20K
Autres frais	\$0	(fonderie) \$250K
Total des frais fixes	\$105K	\$420K
Par puce	\$250	\$10
Bombardier : 203 systèmes	$\$105K + 203 \times \$0.25K = \$156K$	$\$420K + 203 \times \$0.01K = \$422K$
Autres contrats : $10^3$ systèmes	$\$105K + 10000 \times \$0.25K = \$2.61M$	$\$420K + 10000 \times \$0.01K = \$520K$

Pour le contrat de Bombardier, le FPGA est plus avantageux par un facteur de  $422/156 = 2.7\times$ .

Si les autres contrats se matérialisent, alors il vaudrait mieux investir dans la logique fixe (facteur de  $2610/520 = 5\times$ ).

B. Le facteur temps peut être important aussi. La logique fixe a un délai de 4 mois. Les risques sont grands. Si on découvre un bogue, il faut attendre à nouveau 4 mois et repayer les frais fixes de fonderie pour faire une nouvelle puce. Avec le FPGA, les puces seraient disponibles dans un mois. Mais si un bogue était découvert, il ne serait pas nécessaire de faire une nouvelle commande, seulement reprogrammer la puce.

C. La performance n'est pas discutée dans la question, alors on suppose que le FPGA et la logique fixe rencontrent aussi bien les performances requises l'une que l'autre.

D. Conclusion. Le FPGA est nettement préférable à tous points de vue pour le contrat de 203 systèmes. Par contre, si des contrats pour 10000 systèmes se présentaient, il y aurait alors un net avantage à aller en logique fixe (\$2M de moins en dépenses). Il semblerait donc prudent de débiter avec la solution FPGA et à attendre ensuite pour voir si les autres contrats se matérialisent. Les \$156K de la solution FPGA sont assez faibles: c'est le salaire de 1.5 personnes pendant un an.

### 3. Solution

Deux solutions sont proposées ici.

```
architecture codegray1 of temperature is
begin

    with temperature select
    chauffage <=
        '0' when "1110" | "1010" | "1011" | "1001" | "1000",
        '1' when others;

    with temperature select
    climatisation <=
        '1' when "1011" | "1001" | "1000",
        '0' when others;

end codegray1;

architecture codegray2 of temperature is
begin

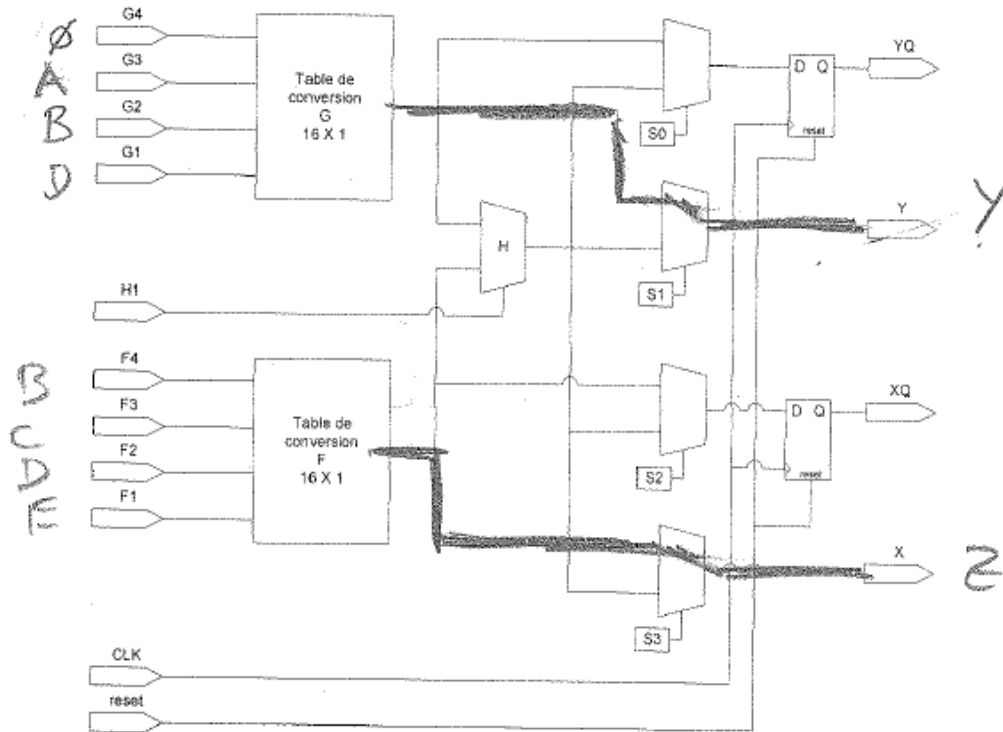
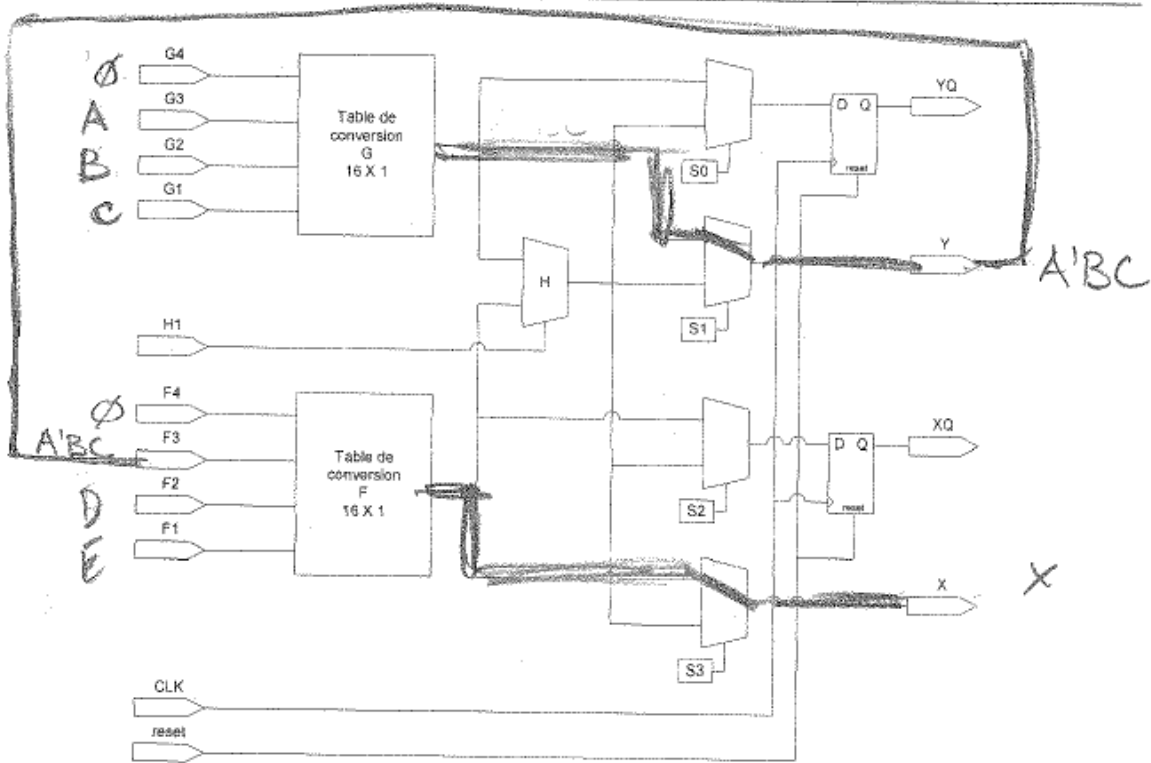
    with to_integer(unsigned(temperature)) select
    chauffage <=
        '0' when 14 | 10 | 11 | 9 | 8,
        '1' when others;

    with to_integer(unsigned(temperature)) select
    climatisation <=
        '1' when 11 | 9 | 8,
        '0' when others;

end codegray2;
```



4. Solution



$\emptyset$	A	B	C	$A'BC$
G4()	G3()	G2()	G1()	G()
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

$\emptyset$	A'BC	D	E	X
F4()	F3()	F2()	F1()	F()
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

$A'BC$

$X = A'BC + DE'$

$\emptyset$	A	B	D	Y
G4()	G3()	G2()	G1()	G()
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

B	C	D	E	Z
F4()	F3()	F2()	F1()	F()
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1

$Y = ABD'$

$Z = BC(D+E)$