

INF3500 : Conception et réalisation de systèmes numériques

Examen final

Automne 2015

Durée: 2h30

Pondération: 50%

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

Calculatrice: Programmable permise.

Directives particulières:

- Ordinateurs, tablettes, téléphones, caméras et autres 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. (8 points)**

On est en juillet 2016. Vous travaillez chez FPGA Solutions Consultants Inc. Tous les employés, sauf vous et la présidente, sont partis en vacances. La présidente vient de recevoir un appel d'un client qui veut implémenter un nouveau produit : un système de vision artificielle pour une voiture autonome. Un prototype est fonctionnel, et le client veut maintenant procéder à un premier déploiement dans 500 voitures en pré-production.

Le système doit traiter des données provenant de plusieurs caméras, pour un débit total de  $750 \times 10^6$  résultats par seconde. On peut l'implémenter avec un processeur unique qui produit un résultat par cycle d'horloge. Ce processeur occupe l'équivalent de 3500 blocs logiques, et sa fréquence maximale dépend de la technologie utilisée. Il est possible d'instancier plusieurs processeurs en parallèle pour multiplier le débit. Cependant, il est essentiel que tout le système puisse être implémenté sur une seule puce.

La présidente vous demande de l'aider à choisir parmi trois technologies dont les données de comparaison sont présentées au tableau suivant. Laquelle des trois technologies privilégiez-vous? Énoncez clairement toutes les suppositions raisonnables que vous faites et montrez tous vos calculs.

Technologie	Frais fixes de développement	Coût par puce	Fréquence d'horloge maximale	Ressources de calcul (blocs logiques)
A. Logique fixe	\$875 000	\$24	2.00 GHz	Max. 25 000
B. FPGA moyenne gamme	\$12 000	\$1 350	0.50 GHz	7 500
C. FPGA bas de gamme	\$12 000	\$49	0.25 GHz	1 750

*Utilisez le verso si nécessaire)*

**Question 2. (8 points)**

Considérez le code VHDL suivant.

```

library ieee;
use ieee.std_logic_1164.all;
entity module9 is
  port (
    clk: in std_logic;
    A, B, C: in std_logic;
    X, Y, Z: out std_logic
  );
end module9;
architecture arch of module9 is
  signal F, G : std_logic;
begin
  process(clk) is
  begin
    if rising_edge(CLK) then
      F <= B and C;
      G <= B xor C;
    end if;
  end process;

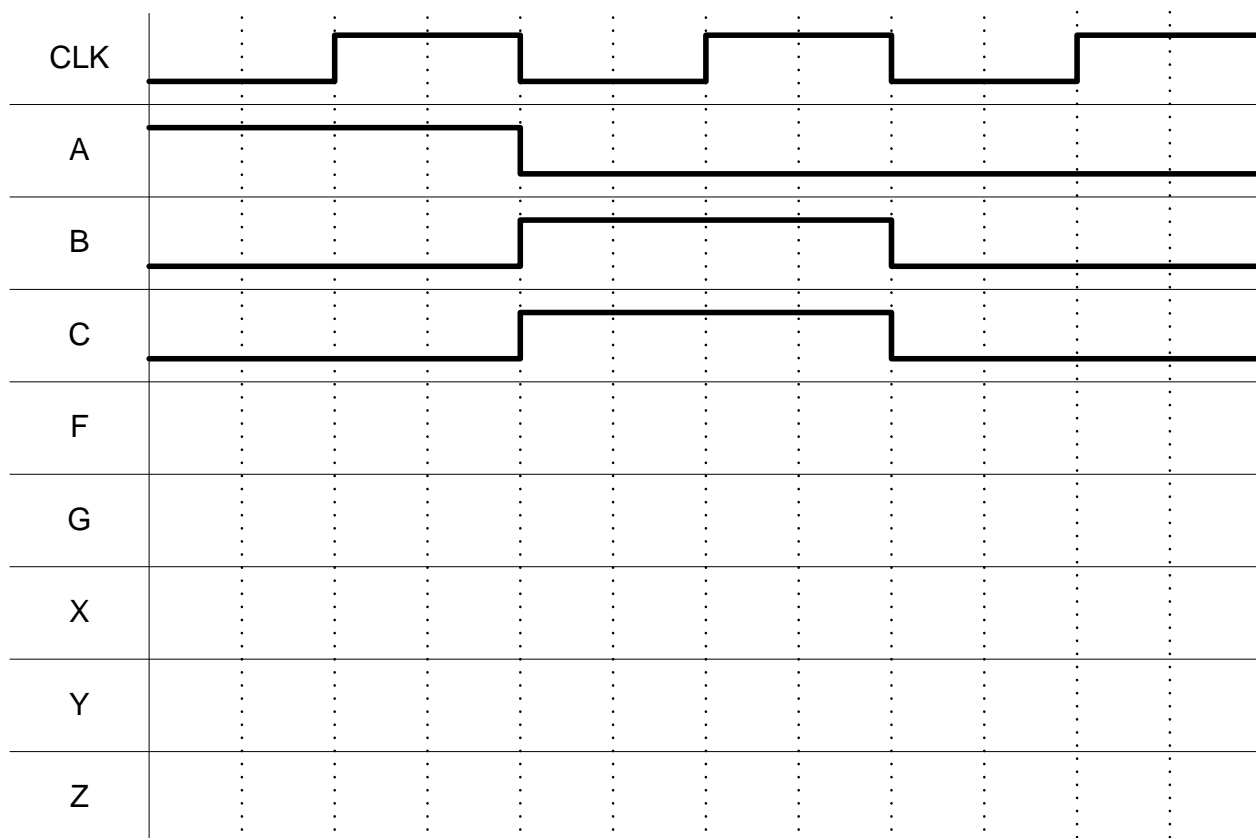
  X <= F or A;

  process(F, G)
  begin
    Y <= F and G;
  end process;

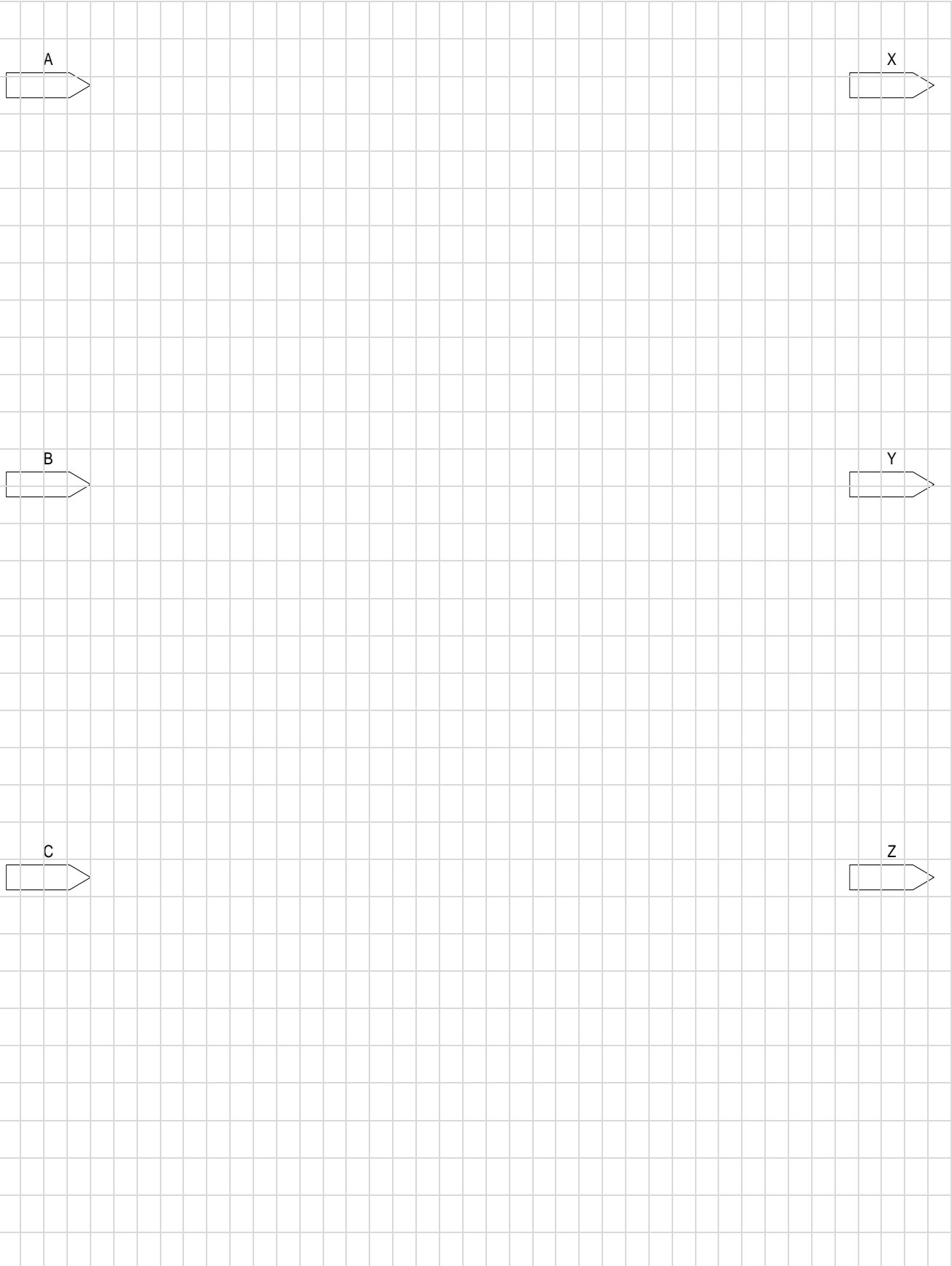
  process(A, B, C)
  begin
    if C = '1' then
      Z <= A or B;
    end if;
  end process;
end arch;

```

a. (4 points) Complétez le chronogramme suivant pour ce module.



b. (4 points) Donnez un schéma d'éléments à mémoire et de portes logiques correspondant à ce code.



Question 3. (10 points)

Les mémoires adressables par contenu (*Content Adressable Memory* – CAM) ont une multitude d’applications. On les retrouve entre autres dans les mémoires caches des microprocesseurs ainsi que dans les routeurs des réseaux informatiques. En mode écriture, une CAM fonctionne comme une RAM; on fournit une adresse et une donnée, et la CAM entrepose la donnée à l’adresse indiquée. Alors qu’une RAM a un mode lecture, une CAM a plutôt un mode recherche. Dans une RAM en mode lecture, on fournit une adresse et la RAM retourne une donnée. Dans une CAM en mode recherche, on fournit une donnée, et la CAM indique si la donnée est présente ou non dans sa mémoire. Si la donnée est présente, la CAM indique l’adresse de la cellule où la donnée réside, et elle active un signal spécial de succès. Si la donnée est présente dans plus d’une cellule, la CAM retourne l’adresse de la cellule la plus près de 0. Si la donnée n’est pas présente, la CAM n’active pas le signal de succès.

Considérez la déclaration d’entité en VHDL d’une CAM donnée ici. En mode écriture, la donnée doit être écrite lors d’une transition positive d’horloge. En mode recherche, le signal de succès et l’adresse de la cellule où la donnée réside doivent être entreposés dans des registres activés sur une transition positive d’horloge.

```
library ieee; use ieee.std_logic_1164.all;
entity CAM is
  generic (
    N : positive := 4; -- nombre de mots dans la CAM
    W : positive := 8 -- largeur des mots dans la CAM
  );
  port(
    clk, reset : in std_logic;
    mode : in std_logic; -- 0 recherche, 1 écriture
    donnee : in natural range 0 to 2 ** W - 1;
    adresseIn : in natural range 0 to N - 1;
    adresseOut : out natural range 0 to N - 1;
    succes : out std_logic
  );
end cam;
```

a. (5 points) Donnez une architecture en VHDL synthétisable correspondant à ces spécifications.

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

*Utilisez le verso si nécessaire)*

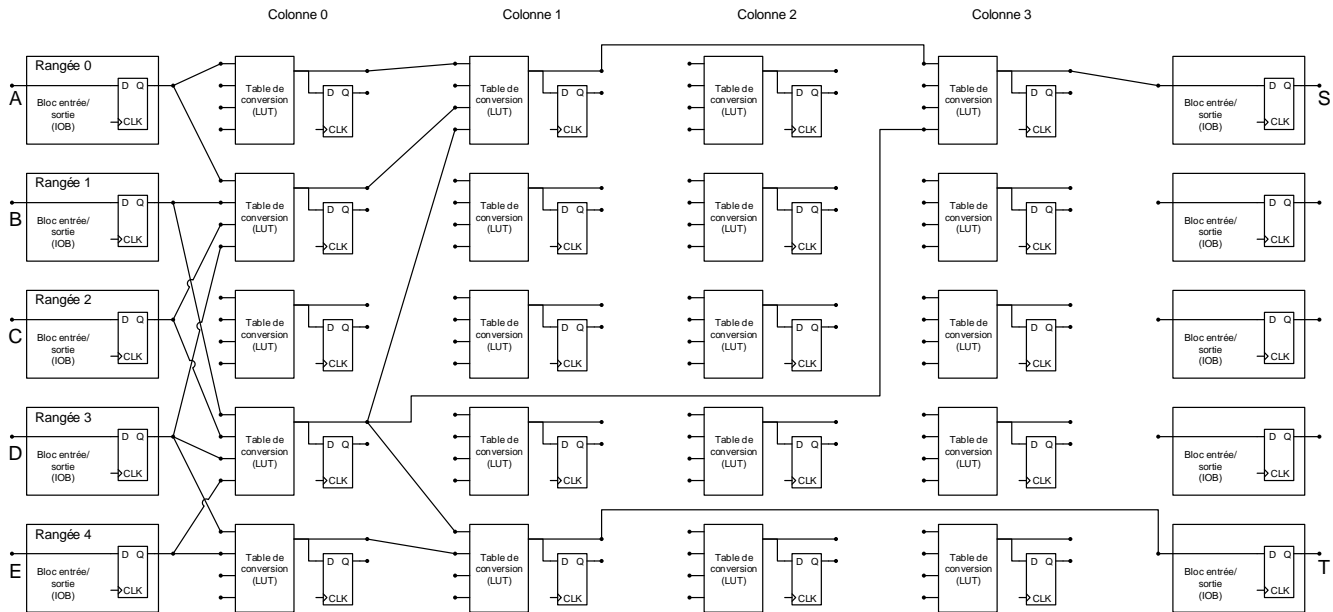
b. (5 points) Donnez un diagramme du chemin des données d'une CAM correspondant à ces spécifications.

A large grid of graph paper, consisting of 20 columns and 30 rows of small squares, intended for drawing a data path diagram.

**Question 4. (8 points)**

Considérez le diagramme suivant montrant un circuit implémenté sur un FPGA simplifié. Les entrées du circuit sont les signaux A, B, C, D et E. Il y a deux sorties S et T. Le FPGA est composé de tranches ayant chacune une table de correspondance (LUT) et une bascule pouvant être utilisée ou non, et de blocs d'entrée/sortie (IOB) dans lesquels les bascules sont toujours utilisées.

Pour ce FPGA, les LUT ont un délai de 0.3 ns. Les bascules ont un délai de 0.4 ns, un temps de préparation  $t_{su}$  de 0.2 ns, et un temps de maintien  $t_h$  de 0.1 ns. Chaque fil d'interconnexion ajoute un délai de 0.25 ns pour chaque rangée et chaque colonne de distance. Bien que des lignes diagonales soient tracées sur le diagramme, tous les signaux sont routés en lignes horizontales et/ou verticales.



a. (2.5 points) Identifiez le chemin critique du circuit et donnez la fréquence maximale d'horloge.

---

---

---

---

---

---

---

---

---

---

b. (0.5 point) Donnez la latence du circuit, en nombre de cycles d'horloge.

---

---

---

---

---

---

---

---

---

---

c. (3 points) On veut maximiser le débit de ce circuit par la technique du pipeline. Montrez clairement sur le diagramme quelles bascules utiliser. Quel est le débit maximal qui peut être atteint par ce circuit? Énoncez clairement toutes vos suppositions et justifiez complètement votre réponse.

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

d. (2 points) Par rapport à votre réponse en c., serait-il possible d'augmenter encore plus le débit du circuit en modifiant son placement et/ou son routage? Si oui, expliquez comment et donnez la fréquence maximale d'horloge attendue. Si non, expliquez pourquoi.

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

---

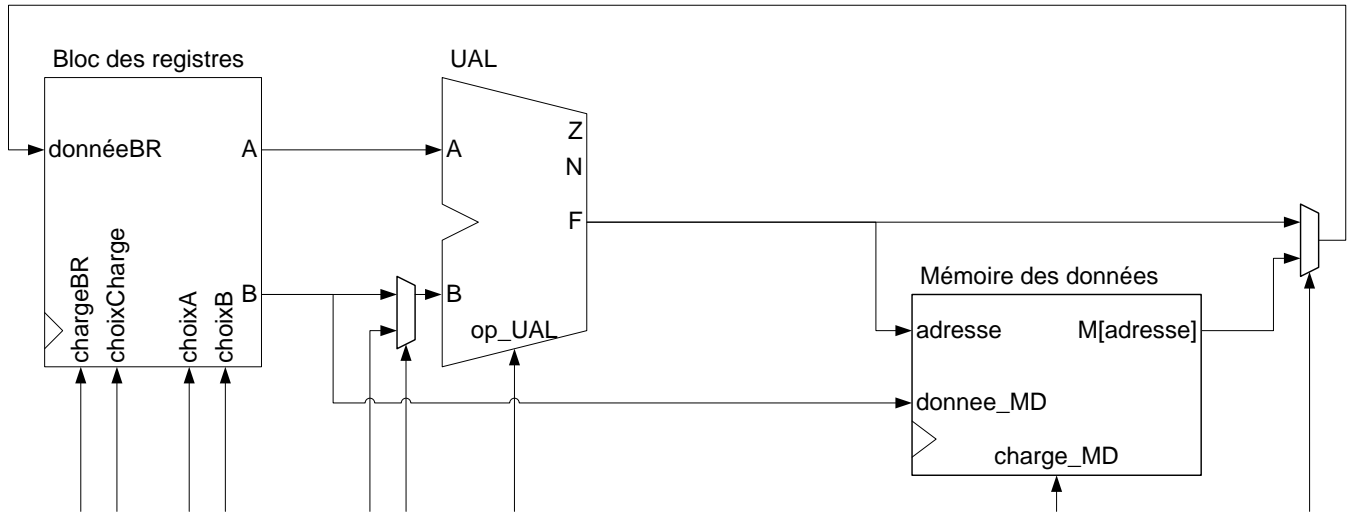
---



**Question 5. (8 points)**

Considérez le chemin des données du processeur PolyRISC montré ici. Supposez que le bloc des registres comporte 4 registres de 16 bits, et que l'UAL peut effectuer les 8 opérations suivantes : A, B, A + B, A – B, A ET B, A OU B, NON A, A OUX B. Supposez que la mémoire des données comporte 256 mots de 16 bits.

Estimez combien de ressources seraient nécessaires pour implémenter le chemin des données du processeur PolyRISC sur un FPGA de la famille Virtex-5 que vous avez utilisé au laboratoire. Énoncez clairement toutes vos suppositions et justifiez complètement votre réponse. Donnez votre réponse en termes de LUT, de bascules, de tranches DSP48 et de bits de mémoire Block RAM.



*Utilisez le verso si nécessaire)*

**Question 6. (8 points)**

a. (2 points) Considérez le code VHDL suivant qui modélise une machine à états. Combien de vecteurs de tests seraient nécessaires, au minimum, pour faire un test exhaustif? Justifiez complètement votre réponse.

<pre>library ieee; use ieee.std_logic_1164.all; use ieee.numeric_std.all; entity module13 is   port (     clk, reset : std_logic;     A, B, C : in std_logic;     unnombre : in signed(2 downto 0);     unentier : in integer range 0 to 15;     unbool : in boolean   ); end module13;</pre>	<pre>architecture arch of module13 is   type type_etat is (s1, s2, s3, s4, s5, s6);   signal etat : type_etat; begin   --   -- des énoncés cachés!   -- end arch;</pre>
---	---

b. Vous êtes responsable de la vérification d'un système de contrôle de la vitesse de l'autopilote d'une voiture autonome. Le système a quatre entrées : la limite de vitesse (50, 70, 90 ou 100 km/h), la période de la journée (jour ou nuit), la visibilité (au mètre près), et l'état de la chaussée (sèche, mouillée ou enneigée). La sortie du système est la vitesse désirée. La vitesse désirée doit être la plus grande possible sans dépasser la limite de vitesse.

Quand la visibilité est inférieure à 10 m, la voiture doit s'arrêter. Quand la visibilité est entre 10 m et 100 m, la vitesse désirée doit être réduite de moitié. Pour une visibilité supérieure à 100 m, il n'y a pas de contrainte. La nuit, la vitesse désirée maximale doit être réduite de 10%. Quand la chaussée est mouillée ou enneigée, la vitesse désirée doit être réduite de 15%. Par exemple, dans une zone de 90 km/h, la nuit, chaussée mouillée, visibilité de 37 m, la vitesse doit être :  $90 \times 0.5 \times 0.9 \times 0.85 = 34.4$  km/h.

i. (3 points) Proposez un partitionnement en classes pour chacune des quatre entrées.

ii. (1 points) Donnez un ensemble de vecteurs de tests pour un test faible selon votre partitionnement proposé.

iii. (1 point) Combien de vecteurs de tests sont nécessaires pour effectuer un test fort ? Montrez vos calculs.

*Utilisez le verso si nécessaire)*

## Solutions

### Q1.

#### A. Logique Fixe :

Débit : un seul processeur 750 MHz < 2 GHz ok

Espace : 3500 blocs < 25000 ok

Coût : \$875 K + 500 unités × \$24 = \$887 K.

#### B. FPGA haut de gamme :

Débit : deux processeurs nécessaires @ 500 MHz chacun, total 1000 MHz > 750 MHz requis ok

Espace : deux processeurs, total 7000 blocs logiques, donc une seule puce de 7500 blocs logiques ok

Coût : \$12 K + 500 unités × \$1350 = \$687 K

#### C. FPGA bas de gamme :

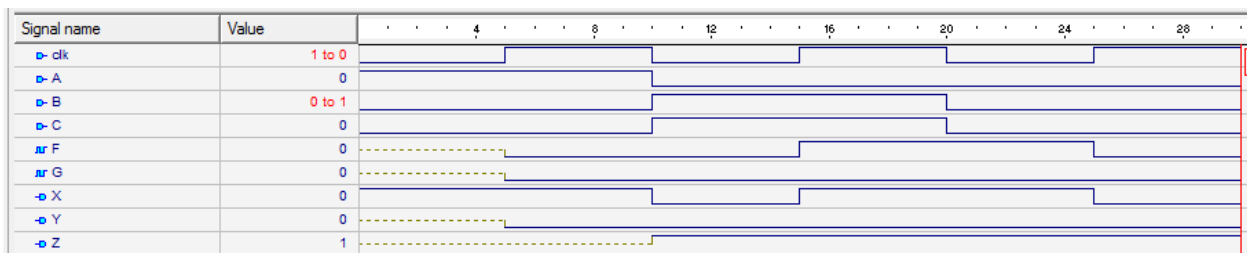
Débit : trois processeurs nécessaires @ 250 MHz chacun, total 750 MHz = 750 MHz requis ok

Espace : trois processeurs, total 10,5 K blocs logiques, donc 6 puces. Cette solution n'est pas acceptable.

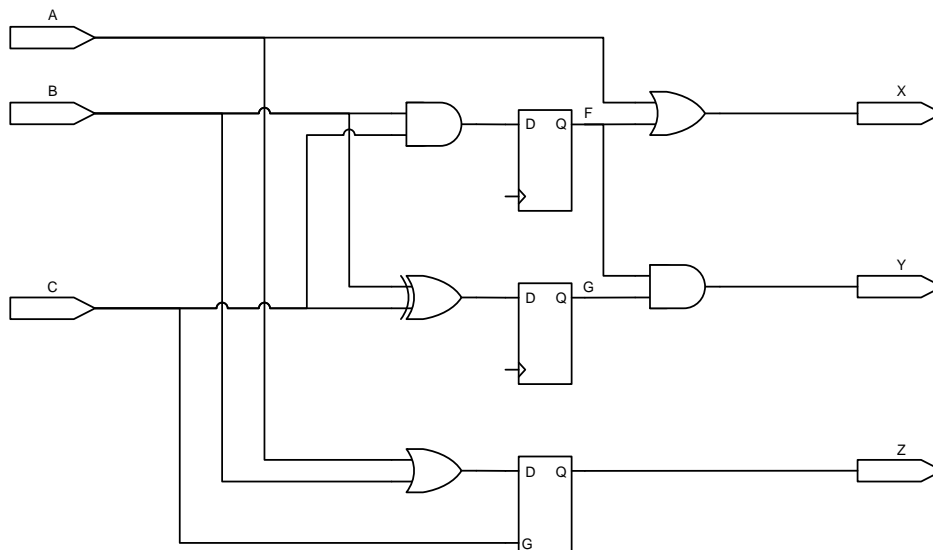
Pour ces conditions, on privilégierait donc la solution B, le FPGA de moyenne gamme.

### Q2.

a.



b.



## Q3.

## a. code

architecture arch1 of cam is

```
type tableautype is array (integer range 0 to N - 1) of natural range 0 to 2 ** W - 1;
signal letableau : tableautype;
```

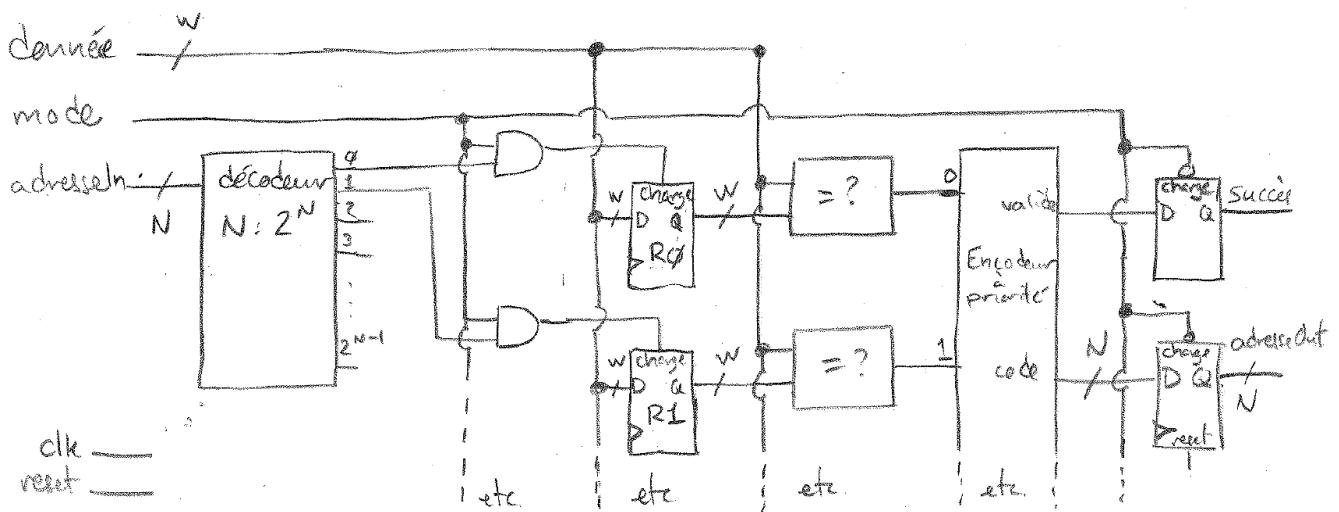
```
begin
```

```
process(clk)
begin
  if rising_edge(clk) then
    if mode = '1' then
      letableau(adresseIn) <= donnee;
    end if;
  end if;
end process;
```

```
process(clk, reset)
begin
  if reset = '1' then
    succes <= '0';
  elsif rising_edge(clk) then
    succes <= '0';
    if mode = '0' then
      for k in 0 to N - 1 loop
        if donnee = letableau(k) then
          adresseOut <= k;
          succes <= '1';
          exit;
        end if;
      end loop;
    end if;
  end if;
end process;
```

```
end arch1;
```

## b. diagramme



**Q4.**

a. Le chemin critique va de l'entrée B, à la LUTR3C0, à la LUTR0C1, à la LUT R0C3, à la sortie S.

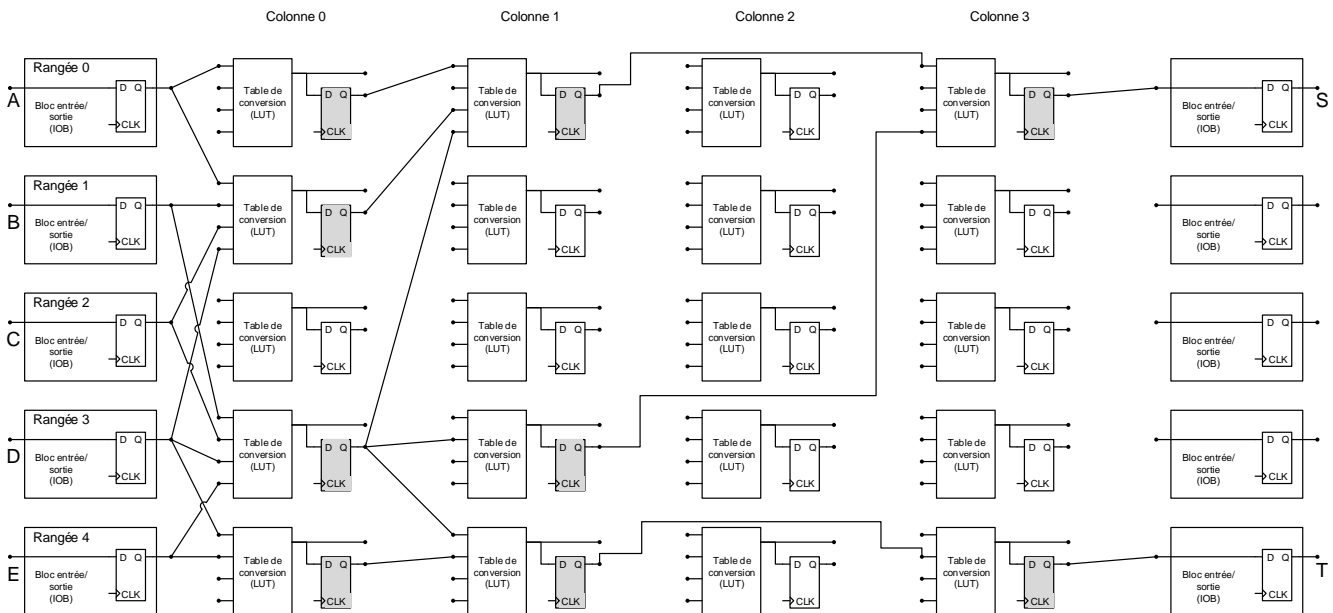
Le délai total est  $t_d + 3 \times t_{LUT} + 10 \times t_{fils} = 0.4 + 3 \times 0.3 + 10 \times 0.25 = 3.8$  ns.

La fréquence maximale d'horloge est  $1/(3.8 + tsu) = 1 / 4$  ns = 250 MHz.

b. Après les bascules aux entrées, il y a seulement une autre bascule dans chaque chemin, soit celles aux sorties. La latence est donc de 1 cycle.

c. Il faudrait tout d'abord insérer des registres de pipeline dans le chemin critique, bascules R3C0, R0C1 et R0C3. Ensuite il faudrait balancer les autres chemins. En y allant colonne par colonne, il faut utiliser toutes les bascules sur des chemins dans la colonne 0. Pour la colonne 1, on doit utiliser la bascule R4C1. Il faut aussi insérer une bascule dans le chemin du milieu, par exemple avec la LUT et la bascule R3C1. La LUT R3C1 ferait la fonction identité. On ajouterait aussi une LUT identité et une bascule à R4C3 sur le chemin du bas.

Le chemin critique serait alors de R3C1 à R0C3 :  $t_d + 5 \times t_{fil} + t_{lut} + t_{su} = 0.4 + 5 \times 0.25 + 0.3 + 0.2 = 2.15$  ns, et la fréquence maximale d'horloge correspondante serait 465 MHz.



d. Oui ça semble possible. Selon la solution en c., le chemin critique va de R3C1 à R0C3, avec une distance de 5 pour les fils. En déplaçant la LUT/basculer R3C1 à la position R2C2, on aurait maintenant une distance de 3 fils seulement. Le chemin critique deviendrait R3C0 à R0C1, avec une distance de 4 fils. On pourrait déplacer R0C1 à R1C1, pour une distance de seulement 3 fils. La distance entre R1C1 et R0C3 serait aussi 3 fils.

Le chemin critique serait alors  $t_d + t_{lut} + 3 \times t_{fils} + t_{su} = .4 + .3 + 3 \times .25 + .2 = 1.65$  ns pour une fréquence d'horloge de 606 MHz.

Pour augmenter davantage la fréquence d'horloge, il faudrait insérer d'autres registres de pipeline.

**Q5.**

1. Bloc des registres (il faut avoir le diagramme du bloc des registres ou en estimer le contenu).

Il faut  $4 \times 16 = 64$  bascules.

Les deux multiplexeurs de sortie ont chacun 6 entrées : un bit par registre et deux bits pour le choix de la sortie. Donc il faut 1 LUT par bit pour chacun des multiplexeurs, donc 16 LUT chacun, donc 32 LUT.

Il faut 4 LUT pour les 4 portes ET.

Le décodeur a deux entrées et 4 sorties, il faut 4 LUT.

Total BR : 64 bascules, 40 LUT environ.

2. UAL

Les fonctions + et – peuvent être combinées, il faut 1 LUT par bit ajouté/soustrait, donc 16 LUT.

Les 6 fonctions logiques A, B, ET, OU, NON et OUX ont 5 entrées : les bits  $A_i$  et  $B_i$ , et 3 bits pour choisir quelle opération est faite. Donc il faut 16 LUT pour ces fonctions.

Finalement il faut choisir entre une somme ou une fonction logique pour chaque bit, donc 16 autres LUT.

Total UAL : environ 48 LUT.

3. Mémoire des données

Il y a 256 mots de 16 bits, donc il faut 4096 bits de mémoire en Block RAM.

4. Deux multiplexeurs

Les deux multiplexeurs nécessitent chacun 1 LUT par bit, soit 2 fois 16 LUT donc 32 LUT.

5. Grand total : environ 64 bascules, 120 LUT et 4096 bits de Block RAM.

(Le FPGA XC5VLX50T contient 28800 LUT, 28800 bascules, et un total de 2160 Kb de Block RAM).

**Q6.**

a. Il y a effectivement 11 bits d'entrée : 3 pour A, B, C; 3 pour « unnombre »; 4 pour « unentier »; et 1 pour « unbool ». On ne compte pas l'horloge ni le reset.

Il y a 6 états.

Dans chaque état, il faut essayer chacune des combinaisons possibles d'entrée. Donc au total on a  $6 \times 2^{11} = 12288$  vecteurs de test.

b.

limite de vitesse, 4 classes : {50}, {70}, {90}, {100}

période de la journée, 2 classes {jour}, {nuit}

visibilité, 3 classes : {[0, 10[}, {[10, 100]}, {]100, ∞[}

chaussée, 2 classes : {sèche}, {mouillée, enneigée}

c.

Il y a plusieurs réponses possibles. Il faut quatre vecteurs de test au minimum. Chaque classe doit être représentée par un élément, au moins une fois dans l'ensemble.

{50, jour, 5, sèche}, {70, nuit, 15, mouillée}, {90, jour, 125, sèche}, {100, nuit, 7, enneigée}

d.

On fait le produit du nombre de classes, donc  $4 \times 2 \times 3 \times 2 = 48$  vecteurs de test.