

INF3500 : Conception et réalisation de systèmes numériques

Examen final

Décembre 2011

Durée: 2h30.

Pondération: 40%.

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

Calculatrice: Programmable permise.

Directives particulières:

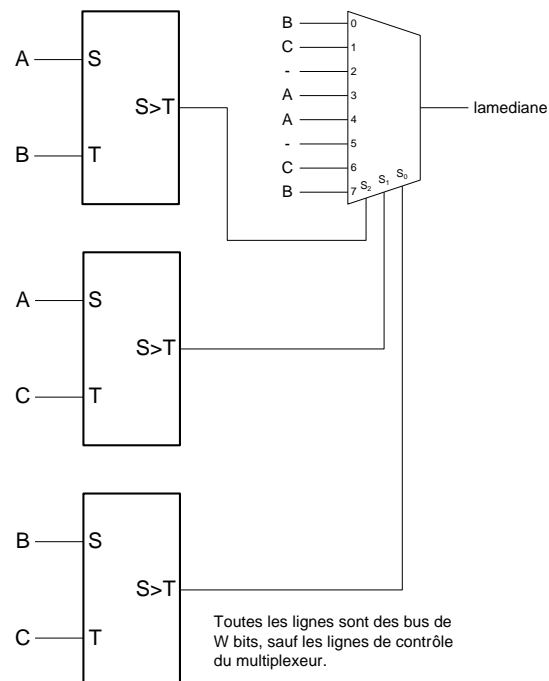
- Ordinateurs interdits.
 - Répondre à toutes les questions, la valeur de chaque question est indiquée.
 - Répondre dans le cahier fourni.
 - Ne pas remettre le questionnaire, sauf la dernière page qui doit être annotée et remise.
 - Ne posez pas de questions. En cas de doute sur le sens d'une question, énoncez clairement toute supposition que vous faites.
-

Question 1. (6 points)

Considérez le diagramme suivant pour un circuit numérique qui accepte en entrée trois nombres signés A, B, et C exprimés avec W bits et qui a pour sortie la médiane des trois nombres. Dans une liste de nombres, la médiane est au milieu de la liste classée par ordre de grandeur. Par exemple, la médiane de {10, -1, 7} est 7. La médiane de {15, 12, 12} est 12.

Dans le diagramme, le bloc S > T donne une sortie de '1' quand l'entrée S est plus grande que l'entrée T, et '0' dans le cas contraire. Le multiplexeur à huit entrées est contrôlé par les sorties des blocs S > T. Les tirets '-' indiquent des entrées sans importance ('*dont care*').

Donnez une architecture en VHDL correspondant à cette entité et à ce diagramme.



```
library IEEE;
use IEEE.std_logic_1164.all;
use IEEE.numeric_std.all;
entity mediane is
  generic (
    W : positive := 16
  );
  port (
    A, B, C : in signed(W - 1 downto 0);
    lamediane : out signed(W - 1 downto 0)
  );
end mediane;
```

Question 2. (6 points)

Donnez une architecture pour l'entité suivante en VHDL synthétisable, en respectant les spécifications données en commentaires.

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

-- Ce compteur a le comportement suivant.
-- Le compte s'effectue sur une transition négative du signal d'horloge.
-- Le compte est activé quand go == '1'.
-- Le signal UD détermine si le compte est vers le haut ('1') ou vers le bas ('0').
-- Le compte est remis à zéro de façon asynchrone quand reset = '1';
entity compteurUD is
generic (W : integer := 8);
port (
    clk, reset, go, UD : in std_logic;
    compte : out signed(W - 1 downto 0)
);
end compteurUD;
```

Question 3. (4 points)

Considérez le code VHDL suivant pour un arbitre de bus rudimentaire.

Donnez un schéma du circuit correspondant à ce code.

```
library ieee;
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
entity arbitreBus is
    generic (n : positive := 2);
    port(
        requetes : in std_logic_vector(2 ** n - 1 downto 0);
        donneesIn : in std_logic_vector(2 ** n - 1 downto 0);
        donneeOut, donneeValide : out std_logic
    );
end arbitreBus;

architecture arch of arbitreBus is
    signal choix : unsigned(n - 1 downto 0);
begin

    process(requetes, donneesIn)
    begin
        donneeValide <= '0';
        for k in 2 ** n - 1 downto 0 loop
            if requetes(k) = '1' then
                choix <= to_unsigned(k, n);
                donneeValide <= '1';
                exit;
            end if;
        end loop;
    end process;

    process(choix, donneesIn)
    begin
        donneeOut <= donneesIn(to_integer(choix));
    end process;

end arch;
```

Question 4. (6 points)

Faites la conception d'un circuit numérique combinatoire qui accepte en entrée un vecteur de 8 bits représentant un nombre positif de secondes. Votre circuit doit avoir deux sorties binaires donnant le nombre de minutes et secondes correspondantes. Donnez un diagramme de votre circuit, ne donnez pas de code VHDL. Indiquez la largeur des ports de sortie en bits. Vous pouvez utiliser des opérations logiques et arithmétiques (mais pas la division ni le modulo), la comparaison, des multiplexeurs, décodeurs et encodeurs, et toute autre porte logique de base. Votre circuit doit être purement combinatoire, il ne doit pas inclure d'éléments à mémoire.

Question 5. (8 points)

Considérez le code VHDL suivant pour un module combinatoire et son banc de test associé.

```

library IEEE;
use IEEE.std_logic_1164.all;

entity module3 is
  port (
    A, B, C : in std_logic;
    F, G : out std_logic
  );
end module3;

architecture arch of module3 is
  signal S, T, U : std_logic;
begin

  S <= A or B;
  T <= A xor B;
  U <= A and C;

  process (S, T)
  begin
    F <= S and T and C;
  end process;

  process (B, U)
  begin
    G <= B or U;
  end process;

end arch;

library ieee;
use ieee.std_logic_1164.all;

entity module3_TB is
end module3_TB;

architecture arch of module3_TB is
  signal A, B, C, F, G : std_logic;
begin

  UUT : entity module3(arch) port map
    (A, B, C, F, G);

  A <= '1' after 0 ns;
  B <= '0' after 0 ns;
  C <= '0' after 0 ns, '1' after 10 ns;

end arch;

```

- (4 points) Donnez la liste des événements, telle qu'elle pourrait être dressée par un simulateur qui exécuterait le banc d'essai.
- (4 points) En vous basant sur votre liste des événements, donnez la valeur de tous les ports et signaux internes du module combinatoire en fonction du temps, en tenant compte des délais deltas.

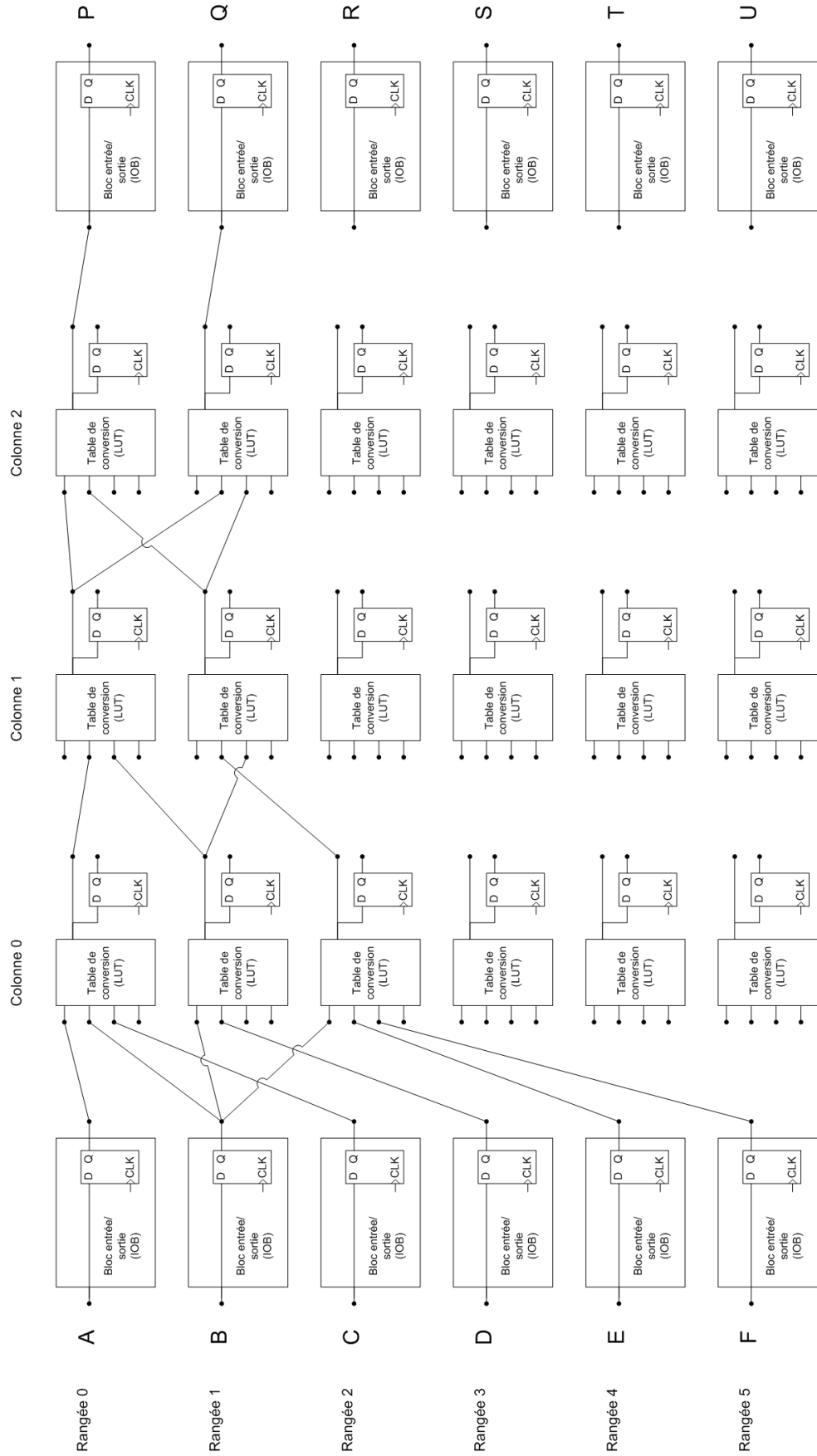
Question 6. (10 points)

Considérez le diagramme de la page suivante montrant un circuit implémenté sur un FPGA simplifié. Les entrées du circuit sont les signaux A, B, C, D, E et F. Il y a deux sorties P et Q. 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 LUTs ont un délai de 2.25 ns. Les bascules ont un délai de 0.75 ns, un temps de préparation t_{su} de 0.25 ns, et un temps de maintien t_h de 0.1 ns. Les interconnexions entre les rangées de tranches et entre les IOB et les tranches ont un délai de 0.2 ns.

- a. (2 points) Identifiez le chemin critique du circuit et donnez la fréquence maximale d'horloge.
- b. (1 point) Donnez la latence du circuit en secondes.
- c. (3 points) On désire atteindre un débit minimal de 165×10^6 résultats par seconde, où un résultat est une paire (P, Q). Expliquez comment faire pour atteindre ce niveau de performance. Justifiez complètement votre approche.
- d. (1 point) Donnez la latence du circuit modifié selon votre approche décrite en c. en secondes.
- e. (2 points) On désire atteindre un débit minimal de 550×10^6 résultats par seconde, où un résultat est une paire (P, Q). Est-ce possible? Si oui, expliquez comment faire. Sinon, expliquez pourquoi.
- f. (1 point) Pour ce FPGA, donnez la fréquence maximale d'horloge qu'un circuit pourrait atteindre. Expliquez votre réponse.

Question 6



INF3500 : Conception et réalisation de systèmes numériques
 Solutionnaire d'examen final
 Décembre 2011

Solutions

Q1 Solution:

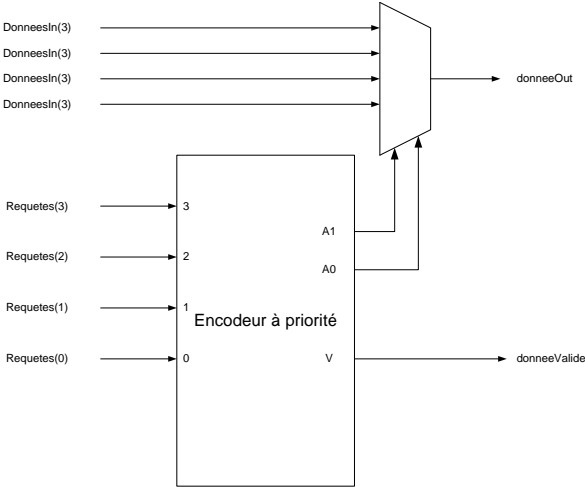
```
architecture arch of mediane is
  signal ApgB : std_logic := '0';
  signal ApgC : std_logic := '0';
  signal BpgC : std_logic := '0';
begin
  ApgB <= '1' when A > B else '0';
  ApgC <= '1' when A > C else '0';
  BpgC <= '1' when B > C else '0';
  with ApgB & ApgC & BpgC select
    lamediane <=
      B when "000",
      C when "001",
      A when "011",
      A when "100",
      C when "110",
      B when "111",
      (others => 'X') when others;
end arch;
```

Il y a une multitude de solutions acceptables. On peut aussi imaginer d'utiliser un processus avec des conditions if-else.

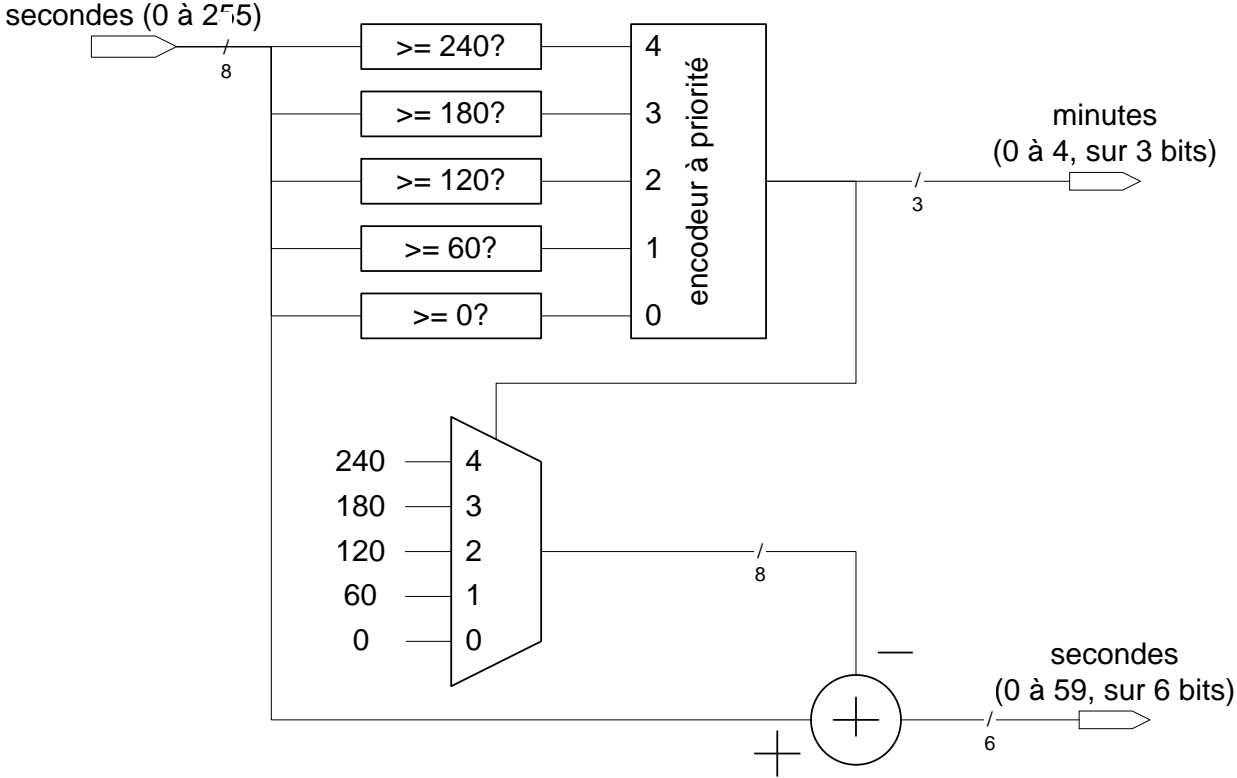
Q2 Solution:

```
architecture arch of compteurUD is
begin
  process(CLK, reset)
    variable compte_int : signed(W - 1 downto 0);
  begin
    if reset = '1' then
      compte_int := (others => '0');
    else
      if falling_edge(CLK) then
        if go = '1' then
          if UD = '1' then
            compte_int := compte_int + 1;
          else
            compte_int := compte_int - 1;
          end if;
        end if;
      end if;
    end if;
    compte <= compte_int;
  end process;
end arch;
```

Q3 Solution:



Q4 Solution:



Q5 Solution :

a. Liste des événements

À $t = 0 + 0 \Delta$, initialisation de la simulation (tout à 'U')À $t = 0 + 1 \Delta$, assignation de 100 à (A, B, C)À $t = 0 + 2 \Delta$, évaluation des signaux S, T, U et G (provoquées par les changements sur A, B, C)À $t = 0 + 3 \Delta$, évaluation de F et G (provoquées par les changements sur S, T, U)

Aucun nouvel événement n'est ajouté à la liste d'événements.

À $t = 10 \text{ ns} + 0 \Delta$, assignation de 1 à CÀ $t = 10 \text{ ns} + 1 \Delta$, évaluation de U (provoquée par le changement sur C), ** mais pas d'évaluation de F parce que C n'est pas dans la liste de sensibilité du processus **À $t = 10 \text{ ns} + 2 \Delta$, évaluation de G (provoquée par le changement sur U)

Aucun nouvel événement n'est ajouté à la liste d'événements.

b.

Time	Delta	UUT/A	UUT/B	UUT/C	UUT/S	UUT/T	UUT/U	UUT/F	UUT/G
0 ps	0	U	U	U	U	U	U	U	U
0 ps	1	1	0	0	U	U	U	U	U
0 ps	2	1	0	0	1	1	0	U	U
0 ps	3	1	0	0	1	1	0	0	0
10000 ps	0	1	0	1	1	1	0	0	0
10000 ps	1	1	0	1	1	1	1	0	0
10000 ps	2	1	0	1	1	1	1	0	1

Q6 Solution :

a. Le chemin critique va des bascules des blocs d'entrée/sortie A-F aux bascules des blocs d'entrée/sortie P-Q. Le délai total est donné par :

délai bascule + 3 × délai LUT + 4 × délai interconnexions + $t_{su} = 0.75 + 3 \times 2.25 + 4 \times 0.2 + 0.25 = 8.55$ ns. La fréquence maximale est donc 117 MHz.

b. La latence est de 8.55 ns.

c. Dans les tranches de la colonne 1, on insère les bascules dans le chemin du circuit. Le chemin critique est donc coupé en deux et le délai est donné par :

délai bascule + 2 × délai LUT + 2 × délai interconnexions + $t_{su} = 0.75 + 2 \times 2.25 + 2 \times 0.2 + 0.25 = 5.9$ ns. La fréquence maximale est donc 169 MHz – OK.

d. La latence est de deux cycles d'horloge, soit 11.8 ns.

e. Tentative #1 : On insère les bascules dans les tranches des colonnes 0, 1 et 2. Le chemin critique du circuit va maintenant d'une colonne à une autre, et son délai est de :

délai bascule + 1 × délai LUT + 1 × délai interconnexions + $t_{su} = 0.75 + 1 \times 2.25 + 1 \times 0.2 + 0.25 = 3.45$ ns. La fréquence maximale est donc 289 MHz, ce qui est insuffisant.

Puisqu'on ne peut pas pipeliner d'avantage, il faut donc dupliquer le circuit afin de doubler le nombre de résultats calculés par seconde. Il y a suffisamment de tranches disponibles pour faire cela. En principe, on pourrait ainsi atteindre un débit de 2×289 Mrésultats/s, soit 578 MRésultats/s. Cependant, il faudrait aussi doubler le débit des entrées A à F, et aiguiller une entrée sur deux vers le circuit actuel, et l'autre entrée sur deux vers le nouveau circuit.

f. La fréquence maximale d'horloge de ce FPGA est atteinte quand le chemin critique se résume à : bascule, interconnexion, LUT, bascule, avec un délai de $0.75 + 2.25 + 0.2 + 0.25$, correspondant à 289 MHz.