

INF3500 : Conception et réalisation de systèmes numériques

Examen intra

Mercredi 16 juin 2010

Durée: 2h.

Pondération: 20%.

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

Calculatrice: Programmable permise.

Directives particulières:

- Ordinateurs interdits.
 - Un dictionnaire autorisé.
 - Répondre à toutes les questions, la valeur de chaque question est indiquée.
 - Répondre en français ou en anglais.
 - Répondre dans le cahier fourni et soumettre la page du questionnaire indiquée.
 - Ne pas remettre le questionnaire.
 - Ne posez pas de questions. En cas de doute sur le sens d'une question, énoncez clairement toute supposition que vous faites.
-

Question 1. (4 points)

En justifiant clairement vos réponses, indiquez pour chacun des énoncés VHDL suivants si l'opération est synthétisable ou non. On supposera dans tous les cas que les signaux `sig_1`, `sig_2` et `sig_3` sont de type `unsigned` et qu'ils sont correctement dimensionnés.

- a) `sig_3 <= sig_1 + sig_2;`
- b) `sig_3 <= sig_1 / sig_2;`
- c) `sig_3 <= sig_2 / 6;`
- d) `sig_3 <= sig_2 mod 16;`
- e) `sig_3 <= sig_1 * sig_2;`
- f) `sig_3 <= sig_2 rem 3;`
- g) `sig_3 <= 2*sig_2 + 6*sig_1;`
- h) `sig_3 <= 6*(sig_1 - sig_2/16);`
- i) `sig_3 <= 2.0*sig_2 + 6.0*sig_1;`
- j) `sig_3 <= shift_left(sig_2, sig_1);`
- k) `sig_3 <= shift_right(sig_2, 10.0);`
- l) `sig_3 <= sig_2 ** 4;`
- m) `sig_3 <= sig_2 ** 2.0;`
- n) `sig_3 <= shift_left(sig_2, to_integer(sig_1));`
- o) `sig_3 <= sin(real(to_integer(sig_2)));`
- p) `sig_3 <= to_unsigned(integer(sin(3.14) * 2.0**8), 8);`

Question 2. (4 points)

Considérez le code VHDL suivant, décrivant l'entité `parite_paire`, et le banc d'essai qui lui est associé. L'entité `parite_paire` prend en entrée un vecteur de bits et produit un signal de parité tel que celui utilisé dans les systèmes de communication pour la détection et la correction d'erreurs.

Donnez les valeurs prises par la sortie dans l'exécution du banc d'essai sur un temps de simulation de 0.1 μ s. Le temps de simulation débute au temps 0 s. On supposera que la réponse du circuit est instantanée (pas de délai de propagation des signaux d'entrée à la sortie).

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

entity parite_paire is
  port (
    din      : in std_logic_vector(7 downto 0);
    parite   : out std_logic
  );
end parite_paire;

architecture arch of parite_paire is
begin

  process( din ) is
    variable par : std_logic;

  begin

    par := din( 7 );

    for k in 6 downto 0 loop

      if din(k) = '1' then
        par := not( par );
      end if;

    end loop;

    parite <= par;

  end process;

end arch;
```

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

entity parite_paire_tb is
end parite_paire_tb;

architecture behavior of parite_paire_tb is
  component parite_paire
    port( din : in  std_logic_vector(7 downto 0);
          parite : out  std_logic );
  end component;
  signal din : std_logic_vector(7 downto 0) := (others => '0');
  signal parite : std_logic;

begin

  uut: parite_paire port map (din => din, parite => parite);

  stim_proc: process
  begin

    wait for 10 ns;

    din <= "10101011";

    wait for 10 ns;

    din <= "01011011";

    wait for 10 ns;

    din <= "10100001";

    wait for 10 ns;

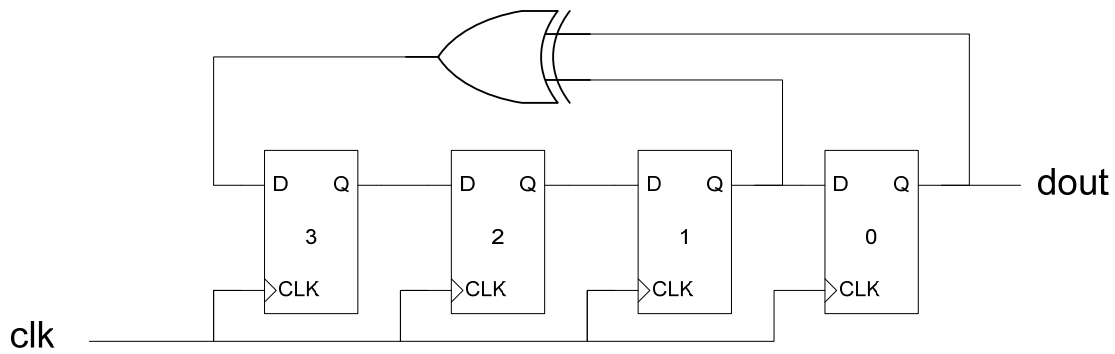
    din <= "00110110";

  end process;

end;
```

Question 3. (4 points)

- a) Donnez en VHDL une description possible du circuit suivant. L'entité prend en entrée le signal d'horloge `clk`, un signal de réinitialisation `rst` (non représenté sur le schéma) et produit en sortie le bit `dout`. L'initialisation doit être synchrone et mettre les bascules à l'état "0011".



- b) À partir de l'état initial, donnez les 16 premières valeurs de la sortie `dout`.

Question 4. (4 points)

- a) Le code suivant comporte un certain nombre d'erreurs. La personne qui l'a écrit avait pour intention de décrire une machine de Moore classique. Rapportez quatre erreurs que vous constatez en justifiant clairement vos réponses. Identifiez les erreurs que vous rapportez en indiquant le ou les numéros de ligne de code qui leurs sont associés.

Attention, seules vos quatre premières réponses sont prises en compte !

```
1  library ieee;
2  use ieee.std_logic_1164.all;
3
4  entity mauvaiscodeur is
5      port (
6          reset : in std_logic;
7          clk : in std_logic;
8          a : in std_logic;
9          b : in std_logic;
10         c : out STD_LOGIC;
11     );
12 end mauvaiscodeur;
13
14 architecture arch of mauvaiscodeur is
15
16     type etats is (etat_A, etat_B, etat_C, etat_D);
17     signal etat_present : etat;
18     signal etat_prochain : etat;
19
20     begin
21
22         process(clk, reset) is
23         begin
24             if reset = '0' then
25                 etat_present <= etat_A;
26             end if;
27
28             if rising_edge(clk) then
29                 etat_present <= etat_prochain;
30             end if;
31
32         end process;
33
34         process(etat_present, a)
35         begin
36
37             if( etat_present = etat_A ) then
38                 if( a = '1' ) then
39                     etat_prochain <= etat_B;
40                 elsif( b = '1' ) then
41                     etat_prochain <= etat_C;
42                 end if;
43             elsif( etat_present = etat_B ) then
44                 etat_prochain <= etat_D;
45             else
46                 etat_prochain <= etat_A;
47             end if;
48
49         end process;
```

```
50
51     process(etat_present)
52     begin
53         if( etat_present = etat_A ) then
54             c <= '0';
55         end if;
56     end process;
57
58     end arch;
```

Question 5. (4 points)

Considérez le code VHDL ci-après. Montrez, sur le diagramme qui suit un résultat possible de la synthèse et de l'implémentation de ce code sur un FPGA Virtex II Pro. Indiquez directement sur le dessin où chaque signal se situe ainsi que les interconnexions entre les blocs. Indiquez dans les tables de vérité fournies le contenu de chacune des tables de conversion que vous utilisez.

Remettez les deux feuilles dans votre cahier d'examen.

```

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

entity compte_2bits is
  port (
    clk : in std_logic;
    en   : in std_logic;
    up   : in std_logic;
    so   : out std_logic
  );
end compte_2bits;

architecture arch of compte_2bits is

  signal c : unsigned(1 downto 0) := "00";

begin

  process( clk ) is
  begin

    if ( rising_edge( clk ) ) then
      if ( en = '1' ) then
        if ( up = '1' ) then
          c <= c + 1;
        else
          c <= c - 1;
        end if;
      end if;
    end if;
  end process;

  process( c ) is
  begin

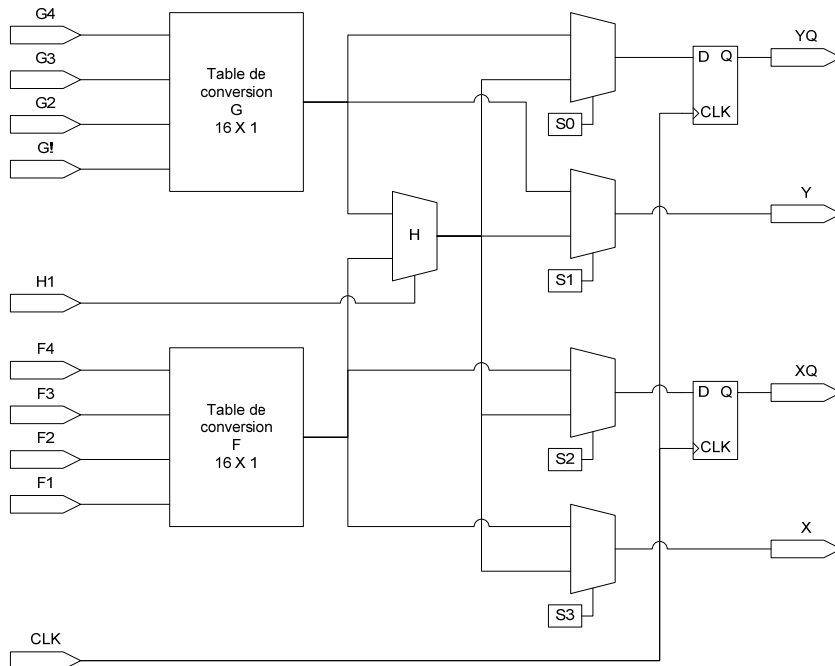
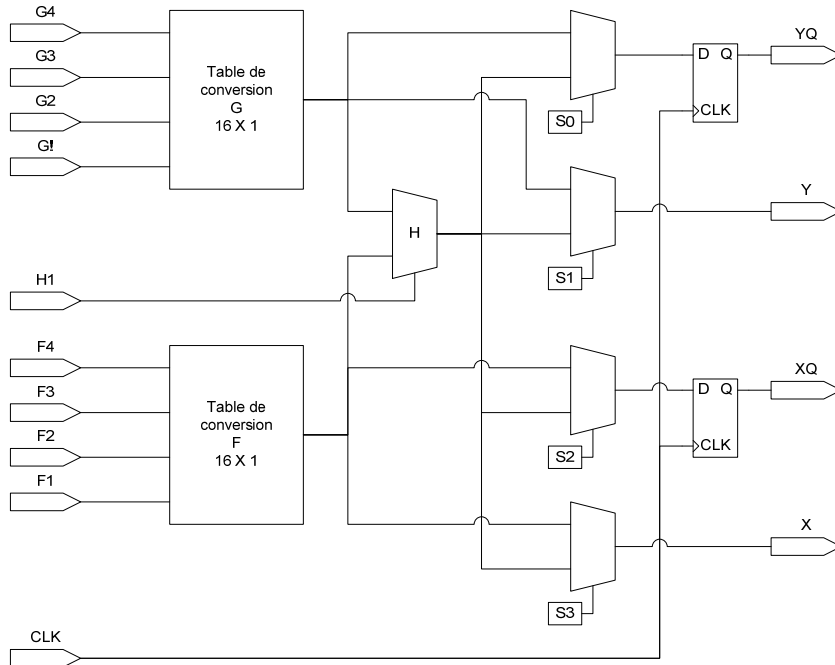
    if ( c = "11" ) then
      so <= '1';
    else
      so <= '0';
    end if;
  end process;

end arch;

```


Annotez cette page et remettez-la avec votre cahier d'examen

Nom : _____ Matricule : _____



Annotez cette page et remettez-la avec votre cahier d'examen

Nom : _____ Matricule : _____

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	