

Troisième EMD, STRM3

Alger le 4 juin 2008

Durée = 02 heures

Nom :

Prénom :

Groupe :

Nb : aucun document n'est autorisé sauf la double feuille des instructions VHDL.

Exercice 01: (6 points) :

Partie I : (3 points)

Réaliser (en complétant les deux versions suivantes) un programme vhdl (comportemental) pour décrire une porte AND a 2 entrées :

a) en utilisant des instructions
Concurrentes : (1,5 points) :

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
entity and2 is
Port ( A : in bit_vector (2 Downto 1);
      s : out bit);
end and2;
architecture bhv of and2 is
begin
with A select
      S<=
end bhv;
```

b) en utilisant un processus (utiliser
seulement if et else: (1,5 points) :

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;
entity and2process is
Port ( a : in bit;
      b : in bit;
      s : out bit);
end and2process
architecture bhv of and2process is
begin
process (
Begin
if a =      or      ;
else      ;
end if
end process;
end bhv;
```

Partie II : (3 points)

a) Donner les 03 caractéristiques d'un processus en VHDL : (1.5 points).

1 :

2 :

3 :

b) Donner la fonction du circuit modélisé par le programme suivant : (1.5 points).

<pre>library IEEE; use IEEE.STD_LOGIC_1164.ALL; use IEEE.STD_LOGIC_ARITH.ALL; use IEEE.STD_LOGIC_UNSIGNED.ALL; entity circuit is Port (sel : in bit_vector (1 downto 0); int: in bit_vector (3 downto 0); s : out bit); end circuit ;</pre>	<pre>architecture progproces of circuit is begin process (sel, int) Begin if sel = "00"then s<=int(3); elsif sel= "01" then s<=int(2); elsif sel = "10" then s <=int(1); else s<= int(0); end if; end process; end progproces;</pre>
--	---

Réponse : **Le circuit modélisé est un :**

Exercice 2: (3 points)

Compléter la description en vhdl (programme ci-dessous) du comparateur 02 bits A et B suivant la table de vérité suivante:

A	B	S0	S1	état
0	0	0	0	A = b
0	1	0	1	A >b
1	0	1	0	b>a
1	1	0	0	A=b

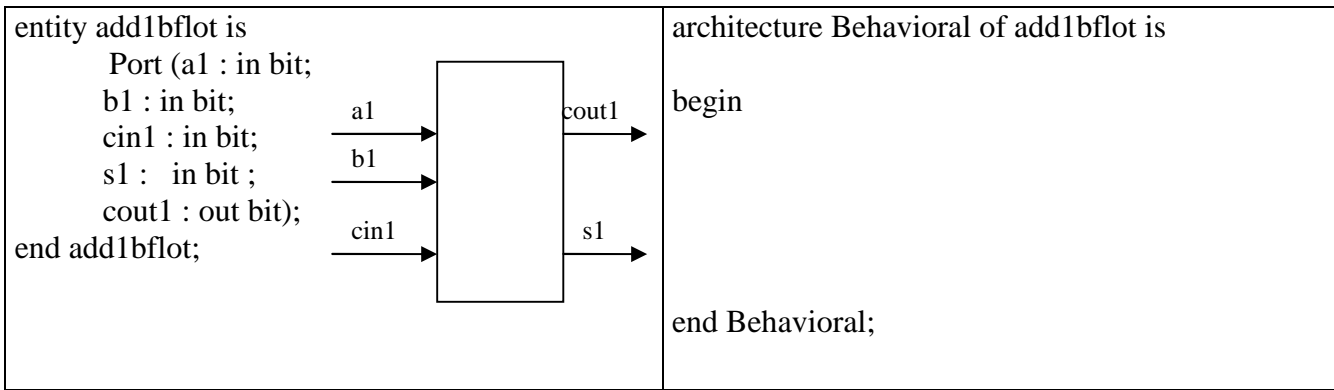
<pre>library IEEE; use IEEE.STD_LOGIC_1164.ALL; use IEEE.STD_LOGIC_ARITH.ALL; use IEEE.STD_LOGIC_UNSIGNED.ALL; entity comp2 is Port (a : in bit; b : in bit; s0 : out bit; s1 : out bit); end comp2;</pre>	<pre>architecture Behavioral of comp2 is subtype v (downto); signal q: begin with v() select q<= when , when , when , when ; s0<= ; s1 <= ; end Behavioral;</pre>
--	---

Exercice 03: (2 points)

- a) compléter la description (style : Equation ou flot de données) en vhdl d'un additionneur 01 bit (cad : 1 bit + 1 bit) en utilisant les équations suivantes d'un additionneur: (2 points)

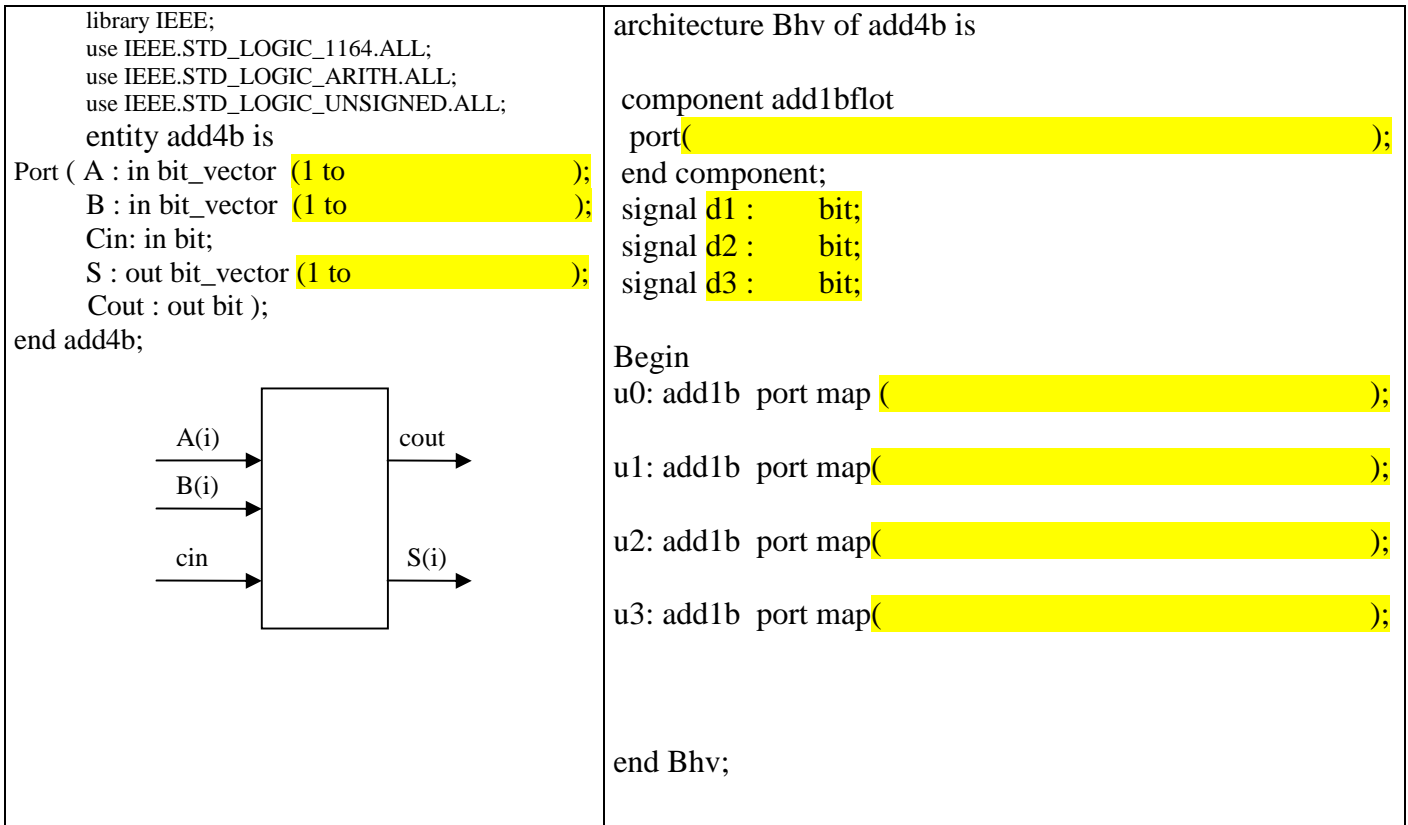
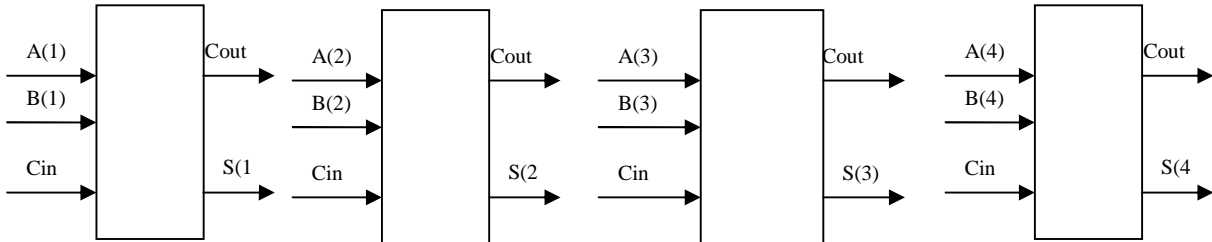
La somme est : **s1=a1 xor b1 xor cin1.**

La retenue est: **cout1= (a1 and b1) or (a1 and cin1)or (b1 and cin1).**



Exercice 04 : (5 points)

Compléter **le schéma** (les connexions et les différents signaux utilisés suivant le programme ci-dessous) et la **description vhdl** en structurel de l'additionneur 04 bits suivant en utilisant l'additionneur 1 bit précédent:



Exercice 05: (4 points)

Compléter le **schéma** et la **description structurelle vhd** d'un additionneur n bits , Suivant les notations du schéma et les données du programme ci-dessous.

<pre> library IEEE; use IEEE.STD_LOGIC_1164.ALL; use IEEE.STD_LOGIC_ARITH.ALL; use IEEE.STD_LOGIC_UNSIGNED.ALL; entity addnb is generic (n:); Port (A : in ; B : ; cin : ; S : ; cout :); end addnb; </pre>	<pre> architecture Behavioral of addnb is component add1bflot port(a1,b1,cin1:in bit;cout1,s1:out bit); end component; signal d: bit_vector (); begin L1: for i in 1 to ,generate add_nbit: add1b port map (); end generate; cout<= ; d(1)<= ; end Behavioral; </pre>
---	---

