

---

# Module : **Programmation Embarqué**

TD2



A.U. 2019-2020

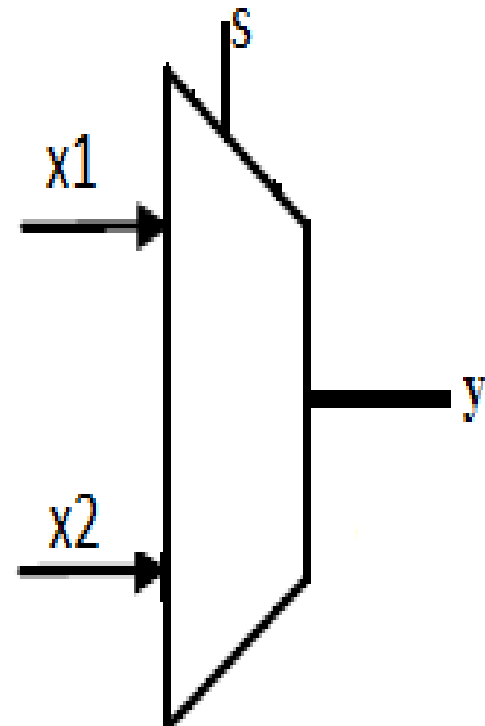
## 1. Code VHDL pour un multiplexeur 2 vers 1 en utilisant l'affectation « with-Select ».

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

entity mux2_1 is
    port(
        .....: in std_logic;
        .....: out std_logic);
end ..... ;

architecture ..... of ..... is
    begin
        with ..... select
            y <= ..... ;
        ..... when others;

end behavior;
```



## 2 Code VHDL pour un multiplexeur 2 vers 1 en utilisant l'affectation conditionnelle « when-else »

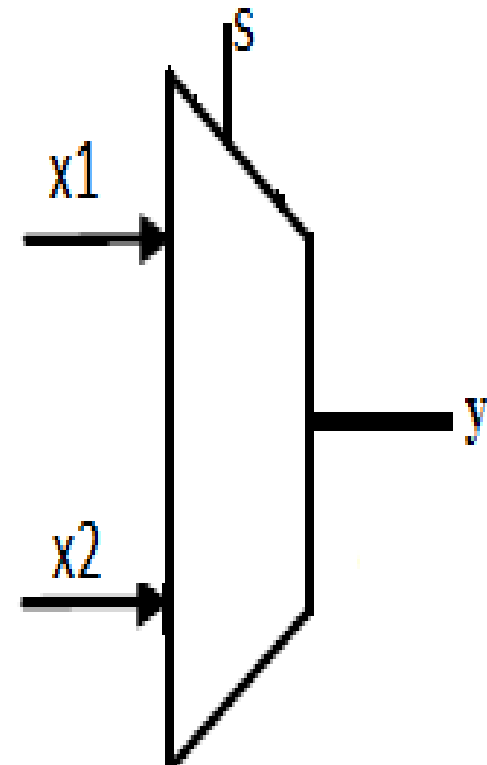
```
library ieee ;
use ieee.std_logic_1164.all;

entity mux2_1 is
    port(
        .....: in std_logic;
        .....: out std_logic);
end ..... ;

architecture ..... of ..... is
    begin

        y <= ..... when ..... else ..... ;

    end behavior;
```

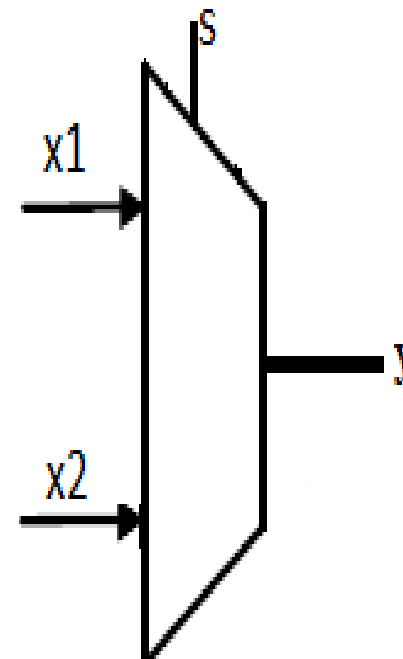


### 3. Code VHDL pour un multiplexeur 2 vers 1 en utilisant l'instruction « If-then-else ».

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

entity mux2_1 is
    port(
        .....: in std_logic;
        .....: out std_logic);
end ..... ;

architecture behavior of mux2_1 is
begin
    process ( ..... )
    begin
        if ..... then;
            y <= .....;
        else
            y <= .....;
        end if;
    end process;
end behavior;
```



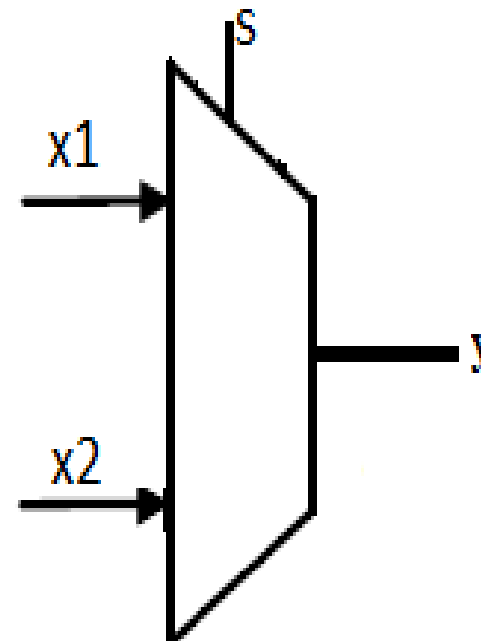
s	y
0	x1
1	x2

#### 4. Code VHDL pour un multiplexeur 2 vers 1 en utilisant l'instruction « case-when ».

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

entity mux2_1 is
    port(
        .....: in std_logic;
        .....: out std_logic);
end ..... ;

architecture behavior of mux2_1 is
begin
    process ( ..... )
    begin
        case ..... is;
            when ..... => ..... <= ..... ;
            when others => ..... <= ..... ;
        end case;
    end process;
end behavior;
```



## Exercice 2

On donne la description VHDL suivante :

```
Library ieee ;
Use ieee.std_logic_1164.all;
Use work.std_arith.all;

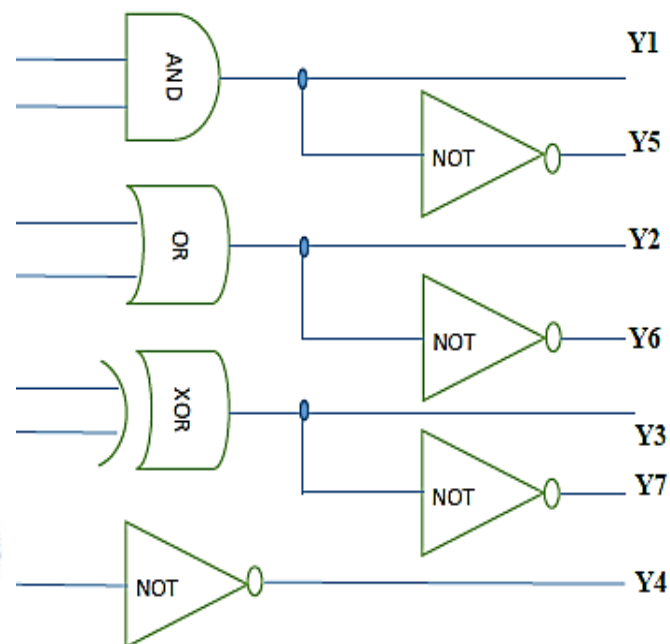
Entity Portes is
Port (A, B: in std_logic;
      Y1, Y2, Y3, Y4, Y5, Y6, Y7: out std_logic);
end Portes ;

Architecture Arch_Portes of Portes is
begin
Y1<= A and B;
Y2<= A or B;
Y3<= A xor B;
Y4<= A not B;
Y5<= A nand B;
Y6<= A nor B;
Y7<= not (A xor B);
end Arch_Portes;
```

- 1) A partir du fichier donner les noms des bibliothèques utilisées.
- 2) Quel est le nom de l'entité ? Quel est le nom de l'architecture ?
- 3) Représenter le schéma fonctionnel de la fonction. On placera naturellement les entrées à gauche et les sorties à droites.

## Correction

- 1) les noms des bibliothèques utilisées.
  - ✓ La bibliothèque IEEE du standard VHDL,
  - ✓ Le paquetage standard pour ieee.Std\_logic\_1164 pour typé les entrées/ sorties et les signaux internes,
  - ✓ Le paquetage personnalisé par l'utilisateur pour work.std\_arith pour spécifié les fonctions utilisées dans le programme VHDL.
- 2) Quel est le nom de l'entité ? **Portes**  
Quel est le nom de l'architecture ? **Arch\_Portes**
- 3) Représenter le schéma fonctionnel de la fonction.



### Correction exercice 3

#### Exercice 3

Entrées				Sorties	
A3	A2	A1	A0	S1	S0
0	1	0	1	1	1
0	1	1	0	0	1
1	1	0	1	1	0

1) Faire la description VHDL de ce circuit ?

On peut passer de `std_logic` vers `std_logic_vector` en présentant les entrées et les sorties comme des vecteurs respectivement.

2) Ecrire une description VHDL qui utilise les entrées et les sorties en vecteurs dans une entité et plusieurs architectures qui contiennent différentes descriptions :

- ✓ Affectation sélective (`with... select... when`) ?
- ✓ Affectation conditionnelle (`when... else`) ?
- ✓ Structure de test (`if... then... else`) ?
- ✓ Structure de choix (`case... is... when`) ?

## Exercice 4

### Exercice 4

Soit la description VHDL suivante :

```
Library ieee ;  
Use ieee.std_logic_1164.all;  
  
Entity Fonction_mystere is  
Port (e0, e1: in std_logic;  
      sel: in std_logic;  
      Sort: out std_logic);  
end Fonction_mystere ;  
  
Architecture V0 of Fonction_mystere is  
begin  
Sort <= e0 when sel= '0' else e1;  
end V0;
```

- 1) Quelle est la fonction réalisée par cette architecture ?
- 2) Réécrire la même fonction avec la structure de test ?
- 3) Pour cette architecture, recopier le chronogramme suivant et complétez le avec la trace du signal « sort » :

