



TP3 : DEVELOPPEMENT D'UN EXEMPLE DE CIRCUIT EN VHDL
DETECTEUR DE PARITE ET MULTIPLEXEUR 4 VERS 1

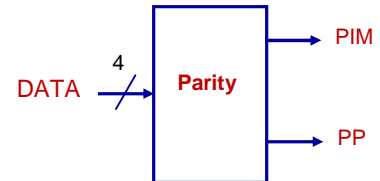


Objectif : L'objectif essentiel de ce 3^{ème} TP est toujours l'écriture de programmes de description VHDL, l'exécution de ces programmes, la simulation fonctionnelle et l'interprétation des résultats. Nous nous intéressons dans ce TP, en particulier, aux instructions concurrentes en VHDL (ex. détecteur de parité et multiplexeur).

Exercice 1 : instruction concurrente WITH ... SELECT ... When ... (affectation sélective)

On souhaite modéliser un **détecteur de parité**, pour des **mots de 4 bits**, par une description VHDL qui est composée de 2 parties :

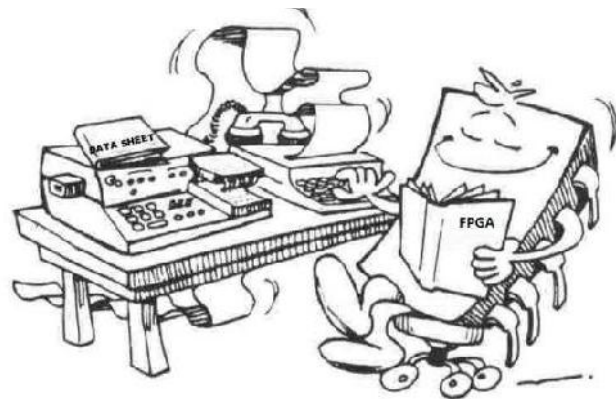
- Signal d'entrée : DATA (la donnée d'entrée est de type vecteur 4 bits),
- Signaux de sortie : PIM (pour parité impaire) et PP (pour parité paire).
 - ✓ **PIM = 1** si le nombre de 1 dans DATA est impaire, sinon PIM = 0.
 - ✓ **PP = 1** si le nombre de 1 dans DATA est pair, sinon PP = 0.



1) Donner la table de vérité de ce circuit ; selon les 16 différentes valeurs possibles de DATA ?

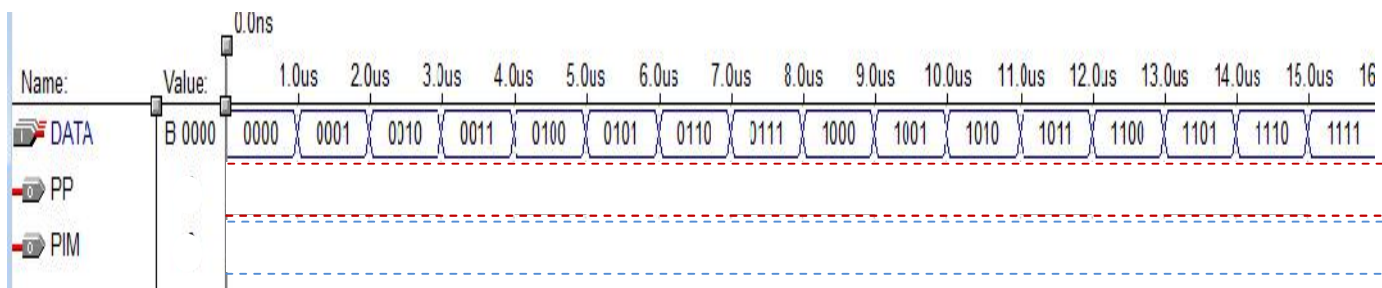
2) Coder le comportement avec l'instruction d'assignation sélective With ... Select ... When ... ?

D3	D2	D1	D0		PIM	PP
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			



3) Sauvegarder, Compiler, Corriger les erreurs si nécessaire, Simuler et Vérifier les résultats via Waveform editor ?

4) Visualiser les chronogrammes de l'entrée "DATA" et les sorties PIM et PP en binaire (temps de 0 à 16 µs) ?



Donner le code de description VHDL de ce circuit transcodeur ?

.....

.....

.....

.....

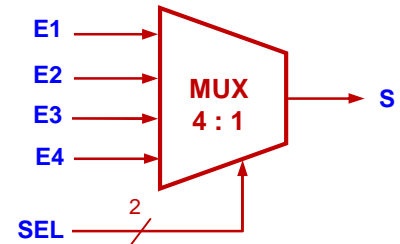
Exercice 2 : instruction concurrente WHEN ... ELSE ... (affectation conditionnelle ; ex. Multiplexeur 4 : 1)

En utilisant l'instruction **When ... else ...** (sans utiliser process), écrire le code de description VHDL qui permet de réaliser un multiplexeur 4 vers 1 (MUX 4 : 1) ; comme le montre la figure.

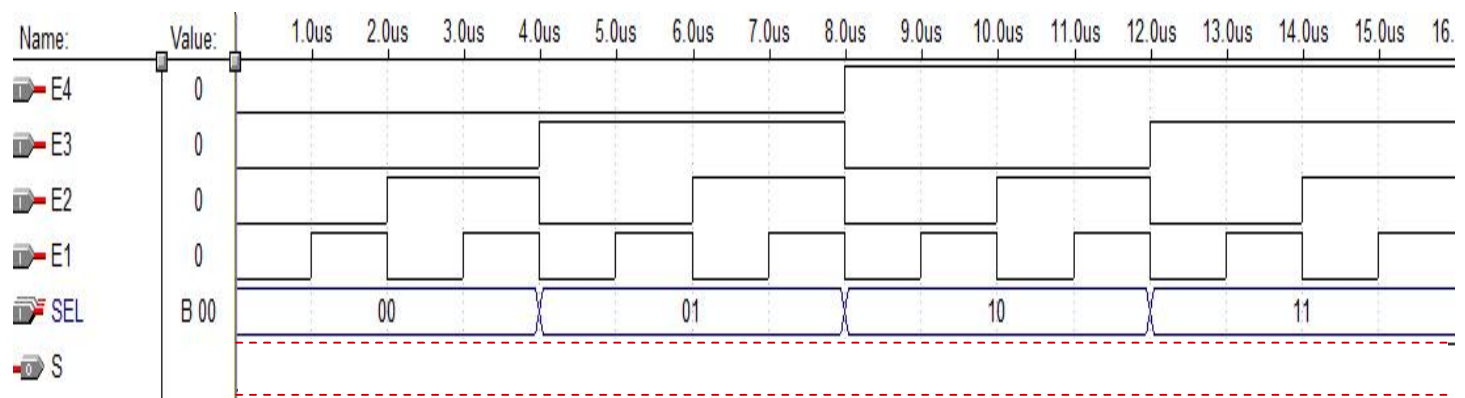
1- Saisir le code, Sauvegarder, Compiler, Corriger les erreurs si nécessaire.

2- Simuler et Vérifier les résultats via Waveform editor ?

SEL = 00 : S E1 SEL = 01 : S E2 SEL = 10 : S E3
SEL = 11 : S E4



3- Visualiser les chronogrammes des entrées / sorties (temps de 0 à 16 µs avec un pas 1 µs) ?



Donner le code de description VHDL de ce circuit ?

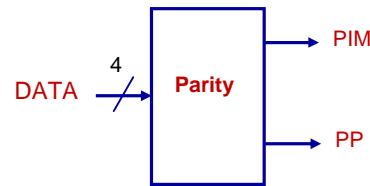


Présenté par : Dr. S. ABADLI.

Corrigés des exercices

Exercice 1

D3	D2	D1	D0	PIM	PP
0	0	0	0	0	0
0	0	0	1	1	0
0	0	1	0	1	0
0	0	1	1	0	1
0	1	0	0	1	0
0	1	0	1	0	1
0	1	1	0	0	1
0	1	1	1	1	0
1	0	0	0	1	0
1	0	0	1	0	1
1	0	1	0	0	1
1	0	1	1	1	0
1	1	0	0	0	1
1	1	0	1	1	0
1	1	1	0	1	0
1	1	1	1	0	1



Signal $S = PIM\ PP$

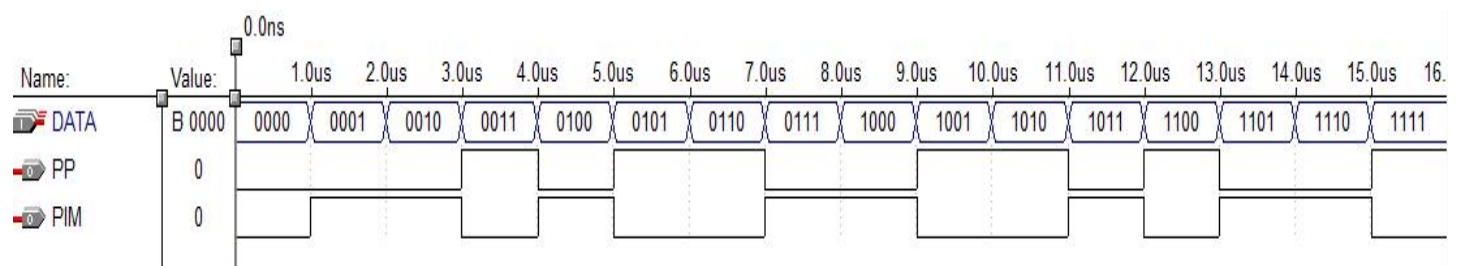
With ...Select ...Else ...

$$PIM = D_0\bar{D}_1\bar{D}_2\bar{D}_3 + \bar{D}_0D_1\bar{D}_2\bar{D}_3 + \dots$$

$$PP = D_0D_1\bar{D}_2\bar{D}_3 + D_0\bar{D}_1D_2\bar{D}_3 + \dots$$

```

Library ieee;
use ieee.std_logic_1164.all;
entity tp3_1 is
    port(DATA: in std_logic_vector (3 downto 0);
          PP,PIM : out std_logic);
end tp3_1;
architecture ARCH1 of tp3_1 is
    signal S : std_logic_vector (1 downto 0);
    Begin
    with DATA select
        S<= "00" when "0000",
            "10" when "0001",
            "10" when "0010",
            "01" when "0011",
            "10" when "0100",
            "01" when "0101",
            "01" when "0110",
            "10" when "0111",
            "10" when "1000",
            "01" when "1001",
            "01" when "1010",
            "10" when "1011",
            "01" when "1100",
            "10" when "1101",
            "10" when "1110",
            "01" when "1111",
            "00" when others;
        PIM<=S(1);
        PP<=S(0);
    end ARCH1;
    
```



Exercice 2

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

entity tp3_2 is
    port (SEL : in std_logic_vector (1 downto 0);
          E1, E2, E3, E4 : in std_logic ;
          S : out std_logic);
end tp3_2;

architecture ARCHI of tp3_2 is
begin
    S <= E1 when ( SEL = "00") else
          E2 when ( SEL = "01") else
          E3 when ( SEL = "10") else
          E4 when ( SEL = "11") else '0';
end ARCHI;
```

