

Exercice: Langage Abel

1. Ecrire un programme Abel où décrire les fonctions suivantes à l'aide d'équations

- $S1 = E1.E2 + E3$
- $S2 = \overline{S1}$
- $S3 = \overline{E3}$

2. Ecrire un programme Abel où décrire la fonction $S = \overline{A}.B$ à l'aide d'une table de vérité

3. Ecrire un programme Abel où décrire les fonctions suivantes à l'aide d'un algorithme (structure when)

- $S1 = \overline{A}$
- $S2 = \overline{A} + ABC$

Solution : Langage Abel**1.**

<pre> Module logique1 declarations logique1 device 'P22V10' ; E1, E2, E3 pin 2, 3, 4 ; S1, S2, S3 pin 14, 15, 16 istype 'com' ; Equations S1 = E1&E2 # E3 ; S2 = !S1 ; S3 = !E3 ; End logique1 </pre>	<p>La directive module commence le programme et précise le nom du fichier</p> <p>Déclaration du PLD utilisé par la directive Device</p> <p>Déclaration des variables d'entrée et affectation aux broches du PLD</p> <p>La directive Istype configure la cellule en sortie du PLD, Com pour combinatoire</p> <p>On définit les bits en sortie sous forme d'équations</p> <p>La directive End termine le programme</p>
---	--

2.

```

Module logique2
declarations
  logique2 device 'P22V10' ;
  A, B pin 3, 4 ;
  S pin 16 istype 'com' ;
Truth_table ([A, B] -> S)
  [0, 0] -> 0 ;
  [0, 1] -> 1 ;
  [1, 0] -> 0 ;
  [1, 1] -> 0 ;
End logique2

```

3.

```

Module logique3
declarations
  logique3 device 'P22V10' ;
  A, B, C pin 3, 4, 5 ;
  S1, S2 pin 16, 17 istype 'com' ;
Equations
  When (A==0) then S1 = 1 ;
  Else S1 = 0 ;

  When (A==0) then S2 = 1 ;
  Else S2 = B&C ;

End logique3

```