

Tronc commun 3A - Durée : 180mn Examen du Cours CE311 « Partie Matérielle »	Session 1 année 2015 – 2016 Un seul document autorisé : « VHDL résumé de syntaxe » Calculatrice interdite
<i>1 Les points donnés dans l'énoncé entre crochets [X] après chaque question représentent le barème et indiquent le temps à passer en minutes sur chaque question</i> <i>2 La qualité de la rédaction (lisibilité, orthographe) sera prise en compte</i>	

Rédiger vos réponses sur des copies séparées : une pour la partie matérielle et une autre pour la partie logicielle

I Partie cours [20/90]

- 1.1 Donnez 3 caractéristiques principales des FPGA et 3 caractéristiques principales des CPLD [2x3].
- 1.2 En VHDL, quel type **normalisé** doit être utilisé pour réaliser des opérations arithmétiques sur des **vecteurs binaires signés** [1]?
- Quels bibliothèques et paquetages doivent-êtré préalablement déclarés pour utiliser ce type [1]?
- Montrez en détaillant l'opération sur des nombres binaires signés en « complément à 2 » (de tailles suffisantes) comment le résultat de l'opération « -5 x 8 » est obtenu [6].
- 1.3 Dessinez le chronogramme du signal S suivant [3]:
- S <= '0', '1' **after** 10 ns, '0' **after** 25 ns, '1' after 30 ns;
- 1.4 En VHDL est-ce que l'ordre des processus dans le code a une importance ? Si non, dans quel ordre sont exécutés les processus ? [6]

II Partie exercices - Conception d'un « simple registre » [70/90]

2.1 Dessinez le schéma d'un **registre de n bits à chargement parallèle** [10] en utilisant uniquement des bascules D Flip-Flop dont l'entité est donnée ci-dessous :

- un vecteur d'entrée (din) de n bits
- une capture de (din) sur les fronts montants de l'horloge (clk)
- une entrée de données (data) pour le chargement parallèle de n bits de large
- une entrée d'autorisation de chargement (enable) active à l'état haut
- une sortie (dout) sur n bits
- une remise à zéro (reset) synchrone active à l'état haut.

library IEEE;

use IEEE.std_logic_1164.all;

entity bascule **is**

```
    port(    d, clk, reset, enable : in std_logic ;  
           q : out std_logic);
```

end entity bascule;

2.2 Ecrire en **VHDL RTL** l'entité générique (*registerN*) et l'architecture (*rtl*) du composant précédent [12]

2.3 Ecrire cette fois en **VHDL structurel** l'architecture (*struct*) de la même entité générique (*registerN*) précédente [14].

2.4 Ecrire en VHDL un testbench permettant de simuler l'**architecture structurelle struct** précédente pour n=8 [16].

Remarques : Votre testbench montrera sur **un seul vecteur de données** que chacune des fonctions « capture, chargement et remise à zéro » fonctionne correctement.

2.5 Dessinez les chronogrammes des signaux **din, enable, data, reset, clk, dout** correspondant très exactement à la simulation du testbench précédent [18].

Remarques : Pensez à indiquer clairement l'échelle des temps sur l'axe temporel