

<p align="center">Examen CE311</p> <p align="center">Partie « conception numérique & VHDL »</p>	<p align="center">3^{ème} année 1^{ère} session 2019-2020</p> <p align="center">Durée indicative de la partie : 90 mn</p> <p align="center"><u>Document autorisé : syntaxe VHDL essentielle</u></p> <p align="center">Calculatrice interdite</p>
<p><i>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></p>	

Conception et validation d'un compteur à pré-chargement [94+6]

L'objectif de cette partie est de concevoir et valider un compteur à pré-chargement, c'est-à-dire un compteur qui compte à partir d'une valeur particulière ; valeur qui est pré-chargée à partir d'une valeur donnée en entrée. Ce compteur incrémente la sortie *C* à chaque front montant d'horloge *clk* quand *cnt* est à '1' et charge la valeur sur l'entrée *L* quand *ld* est à '1' (*ld* est prioritaire, et il y a chargement du compteur même si *cnt* est à zéro). *tc* est un signal en sortie qui indique que le compteur a atteint sa valeur max (après avoir atteint le maximum, la valeur du compteur *C* repasse ensuite à 0 lors du prochain front montant de *clk*). *clr* est un signal de remise à zéro synchrone prioritaire devant *ld* et *cnt*.

1 En considérant sur l'entrée *L* la valeur « 0111 », dessinez un chronogramme montrant l'évolution des signaux *clk*, *ld*, *cnt*, *clr*, *tc*, et *C* **pendant 100 ns** suite à un pré-chargement de l'entrée lors uniquement du 1^{er} cycle d'horloge puis une autorisation de comptage active. Dans ce chronogramme vous considérerez **une horloge *clk* de période 10 ns** [12].

2 Faire un schéma de ce circuit en utilisant les macro-composants suivant : un multiplexeur avec des entrées de 4 bits avec 2 entrées et une sortie, une porte OU logique à 2 entrées et une sortie, une porte ET logique à 4 entrées et une sortie, des bascules avec une entrée reset synchrone, un additionneur à 4 bits en entrées/sorties [16].

3 Commencez par décrire en VHDL RTL l'entité de ce circuit que l'on nommera *counter* [6].

Remarques :

- N'utilisez que des signaux de type *std_logic_vector*
- N'utilisez pas de port de mode inout
- Pensez à déclarer les bibliothèques nécessaires à l'entité et à l'architecture de la question suivante

5 Décrire en VHDL RTL l'architecture *rtl* de *counter* [18].

Conseil : Utilisez une valeur interne *C_int* pour pouvoir lire cette valeur qui correspond à la valeur de la sortie *C*

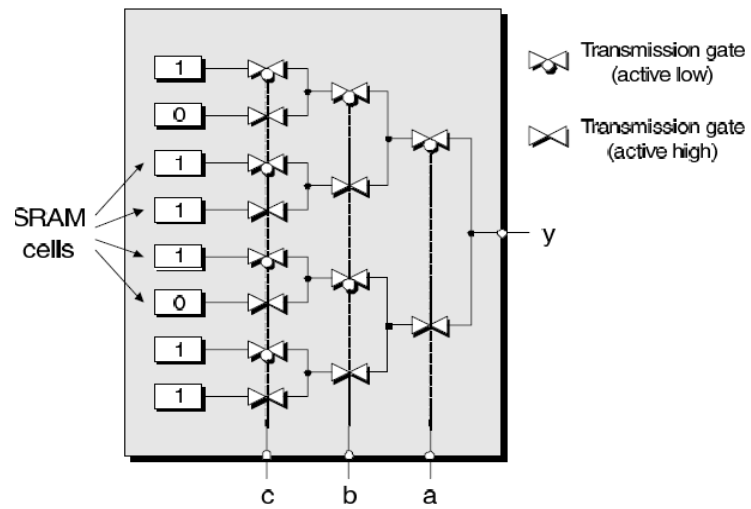
6 Combien de bascules générera la synthèse de ce circuit *counter* ? Expliquez pourquoi ce nombre et vérifiez que cela correspond au nombre de bascules figurant dans votre schéma précédent [6].

7 Décrire le testbench *tb_counter* (entité et architecture) permettant de retrouver **exactement le même chronogramme que celui que vous avez donné à la question 1** [12]

8 Modifiez votre code RTL du *counter* précédent de façon à ce que ce nouveau compteur, après avoir atteint sa valeur maximale, passe à la valeur *C* lors du prochain front d'horloge avec l'autorisation de comptage activée [16].

9 En utilisant une autre couleur, modifiez le schéma précédent pour qu'il corresponde à ce nouveau fonctionnement [8].

10 Optionnel [+2+4] : la figure ci-dessous représente une LUT



10.1 Que signifie l'acronyme LUT et où les trouve-t-on ? [+2]

10.2 Donnez la table de vérité de la fonction combinatoire correspondant à cette figure [+4].