

<p align="center"><b>Examen CE311</b></p> <p align="center"><b>Partie « conception numérique &amp; VHDL »</b></p>	<p align="center"><b>3<sup>ème</sup> année 1<sup>ère</sup> session 2018-2019</b></p> <p align="center"><b>Durée indicative de la partie : 90 mn</b></p> <p align="center"><b><u>Document autorisé : syntaxe VHDL essentielle</u></b></p> <p align="center"><b>Calculatrice interdite</b></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>	
<p><i>Rédigez sur des feuilles séparées les 2 parties de cet examen.</i></p>	

## Conception et validation d'un compteur/décompteur [90]

L'objectif de cette partie est de concevoir et valider un compteur/décompteur (nommé *count\_up\_down*), c'est-à-dire un circuit qui peut soit incrémenter ou décrémenter une valeur C à chaque front d'horloge clk.

- *cnt\_up* permet l'incrémement du compteur (quand *cnt\_up* vaut '1', le circuit incrémente C)
- *cnt\_down* permet la décrémementation du compteur (quand *cnt\_down* vaut '1', le circuit décrémement C)
- *clk* est l'horloge du système
- *clear* est un signal de « remise à zéro » synchrone de C (C devient « 0000 »)
- *set* est un signal de « mise à un » synchrone de C (C devient « 1111 »)
- C est la sortie sur 4 bits du circuit

**Si 2 ou plus entrées de contrôle (*cnt\_up*, *cnt\_down*, *clear* ou *set*) sont à 1 alors le circuit conserve sa valeur courante sur C.**

1 Dessinez un schéma représentant sur un rectangle nommé *count\_up\_down* toutes les entrées et les sorties de ce circuit [2], puis écrivez en VHDL l'entité *count\_up\_down* de ce circuit [4].

Remarques :

- Utilisez uniquement des signaux de type *std\_logic* ou *std\_logic\_vector*
- Pensez à déclarer les bibliothèques et paquetages nécessaires aux types et aux opérations futures

2 Dessinez des chronogrammes montrant l'évolution de tous les signaux d'entrées et de sorties. Commencez par remettre la sortie C à '0' avec *clear*, puis incrémentez C pendant 3 cycles d'horloge, puis mettez C à '1' avec *set*, décrémentez C pendant au moins 3 cycles d'horloge et finalement représentez l'effet de 2 signaux de contrôle ou plus simultanément à '1' [18]

Remarque : dans ce chronogramme vous considérerez **une horloge clk de période 10 ns.**

3 Dessinez un schéma de *count\_up\_down* en utilisant les macro-composants suivants : un bloc de logique combinatoire (Comb, avec en entrée les signaux de contrôle *cnt\_up*, *cnt\_down*, *clear* et *set*), un multiplexeur (Mux, avec des entrées/sorties sur 4 bits et 3 bits de sélection), un incrémenteur (Incr), un décrémenteur (Decr), et un registre de 4 bits à chargement parallèle (Reg) [16].

4 Donnez la table de vérité du bloc de logique combinatoire Comb [6].

5 Combien de bascules FF ce circuit utilise-t-il? [4]

6 Décrivez en VHDL RTL l'architecture *rtl* de ce circuit [20].

7 Combien de bascules générera la synthèse de votre architecture précédente ? Justifiez [4].

Remarque : si vos réponses aux questions 5 et 7 sont différentes expliquez pourquoi !

8 Donnez une description du testbench *tb\_count\_up\_down* permettant de retrouver **exactement le même chronogramme que celui que vous avez donné à la question 2** [16]