

| | |
|---|--|
| 3A/3App - Durée : 90 mn Examen du Cours CE311/CE317 NOM : | 2ème session Année 2018 – 2019 Un seul document autorisé : « VHDL résumé de syntaxe » Calculatrice interdite PRENOM : |
| <p>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</p> <p>2 La qualité de la rédaction (lisibilité, orthographe) sera prise en compte</p> | |

I Partie cours [40/90]

1.0 Dans toute la suite, les entrées et les sorties des entités seront toujours de type *std_logic* ou *std_logic_vector*. Quelles déclarations permettent d'utiliser ces types. Donner le code VHDL [4]

Conseil : dans toute la suite, il ne sera plus nécessaire de rappeler ces déclarations

1.1 Ecrire la déclaration de l'entité *memory* d'une mémoire dont les ports d'entrée et de sortie sont : [4]

addr: 12 bit address input

wr_n: 1-bit write-enable control signal

oe_n: 1-bit output-enable control signal

data: 8 bit bidirectional data bus

1.2.1 Soit *a* et *y* deux vecteurs de type *unsigned* ; expliquer à quelle opération arithmétique l'expression suivante peut être assimilée. [4]

y <= "000" & a(7 downto 3);

1.2.2 En utilisant l'instruction & précédente, décrire l'entité et l'architecture en VHDL d'un *compteur de Johnson* modulo 16 (simple registre à décalage avec un rebouclage avec un inverseur) [20].

Contraintes :

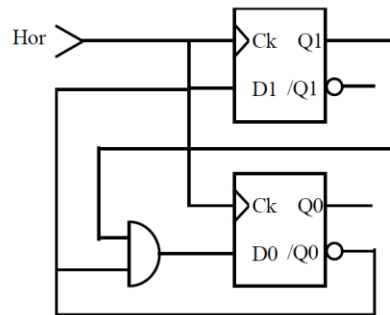
- Utiliser une entrée reset synchrone (actif au niveau haut)
- Ne pas utiliser de ports en mode inout

1.3 A quel(s) composant(s) correspond le code ci-dessous suivant que *b* et *q* sont de simples *std_logic* ou des *std_logic_vector* de largeur *n*. Expliquer. [4+4]

```
process ( a , b)
begin
    if a='1' then
        q <= b ;
    end if ;
end process;
```

II Partie exercices : [50/90]

2.1 Décrire en VHDL RTL (*synthétisable et sans instanciation de composants*) l'entité et l'architecture correspondant au schéma ci-dessous. [20]



2.2 Compteurs modulo m [22]

Attention aux types des signaux utilisés dans les opérations arithmétiques :

2.2.1 Décrire en VHDL l'entité *mod10* et l'architecture *rtl* d'un compteur synthétisable modulo 10 comptant de 0 à 9 indéfiniment (le compteur compte de 0 à 9 puis se répète indéfiniment). [20]

Contraintes :

- Utiliser un signal reset *rst* asynchrone pour initialiser le compteur cpt à 0.
- Ne pas utiliser de ports en mode inout

2.2.2 Combien de bascules seront générées lors de la synthèse de votre compteur mod-10?. Justifiez [8]
Combien au minimum en faut-il théoriquement? Expliquer l'écart éventuel. [2]