

TP VHDL séquentiel

Exercice 1. Compteur de 2 bits

Ecrire le modèle VHDL d'un compteur synchrone à 2 bits, ayant une entrée reset synchrone (PB_1). Utiliser le signal PB_0 pour l'horloge, et le signal LED_10 pour l'affichage des deux bits de sortie.

- Le front montant de l'horloge est le front actif.
 - Le reset est actif au niveau bas (PB_1 = '1')
- a) Créer un nouveau projet et saisir le VHDL dans un fichier avec extension .vhd. N'oubliez pas le fichier de configuration.
 - b) Lancer la synthèse. Il est normal d'obtenir un avertissement au sujet du signal BP_0 qui sert d'horloge. Ouvrir le schéma « RTL » et le schéma « Technology » et constater le résultat de la modélisation. Les minimisations et optimisations de synthèse ont eu quels effets ? Qu'est-ce qui différencie les deux schémas ?
 - c) Simuler le circuit
 - a. Lancer le simulateur ISim
 - b. Affecter des valeurs logiques aux entrées. Affecter des impulsions à l'horloge par clic droit sur le signal PB_0 puis le menu « Force clock... ». Tester le reset d'une manière asynchrone et synchrone.
 - c. Interpréter les résultats. Que pouvez-vous conclure ?
 - d) Si vous êtes satisfait du résultat précédent, programmez le FPGA.
 - e) Constater le fonctionnement correct de la fonction.

Exercice 2. Détecteur de code

On souhaite réaliser un détecteur de code spécial sur une ligne de transmission série (de 1 bit) par une technique séquentielle synchrone. Dès que le code "11010" est détecté, le système doit le signaler en mettant une ligne Alarme à 1. Alarme doit rester inactive pour toute autre combinaison de codes. Les changements de niveau de l'entrée e sont synchronisés à l'horloge. Utiliser les signaux suivants : BP_0 pour l'horloge, SW_0 pour la ligne de transmission série et LED_0 pour le signal Alarme. On peut aussi affecter un signal à l'affichage de l'état. Utiliser le signal LED_7654 à cet effet.

- a) Modéliser ce détecteur en VHDL dans le cas où chaque fausse entrée oblige à tout recommencer.
- b) Saisir le VHDL et procéder à toutes les étapes de synthèse.
- c) Ouvrir le schéma « RTL » et développer tous les blocs. Donner :
 - i. Le nombre de bits d'état utilisés par le détecteur. Sous quelle forme sont-ils réalisés ?
 - ii. Les équations de génération de la sortie, en supposant que les bits d'état sont identifiés S_1 à S_n . Ces équations sont-elles cohérentes avec le code VHDL ? Pourquoi ?
- d) Simuler le circuit dans ISim. Si la simulation fonctionne correctement, programmer le FPGA.
- e) Avancer dans la machine à état avec les impulsions de PB_0. Contrôler les bits d'état pour vérifier l'avancement.

Exercice 3. Recommencer le VHDL pour l'exercice précédent mais pour le cas où une fausse entrée est négligée.

- a) Visualiser le schéma « RTL ».
 - i. le nombre de bits d'état est-il le même ?
 - ii. les équations de transition et de sorties sont-elles plus simples ?
- b) Quelles différences peut-on constater par rapport à l'exercice 2 ? Les bits d'état vous semblent-ils cohérents avec votre VHDL ?