TP 4 : Circuits séquentiels synchrones

Exercice 1 - Test du banc de registres

1.0 Récupérer l'archive

`http://www.ensiie.fr/~christophe.mouilleron/Teaching/ARMA/digproc_skel.tbz2`

et extraire le fichier `register.lgf`.

Vous pouvez aussi extraire les fichiers `combi.lgf` et `alu.lgf` dans le cas improbable où vous n'auriez pas fait toutes les questions taguées [⋆] dans les TP précédents.

1.1 Tester le composant `reg8`.

Exercice 2 - Compteur

Le but de cet exercice est de créer un circuit ayant les fonctionnalités suivantes :
- À chaque front montant de l'horloge, la valeur (entier N) en sortie augmente de 1.
- Un bit de contrôle `reset` permet de remettre N à 0.

On appelle compteur simple sur n bits un circuit réalisant uniquement le premier point avec N codé sur n bits.

2.1 Donner l'automate de Moore correspondant à un compteur simple sur 2 bits.

2.2 Grâce à un additionneur `add_8` et à un registre `df8`, faire un compteur simple sur 8 bits.

2.3 On va maintenant ajouter le support du bit de contrôle `reset`. Dire si le résultat sera un automate de Moore ou un automate de Mealy, et donner l'automate dans le cas $n = 2$.

2.4 Ajouter le support de `reset` au compteur simple sur 8 bits créé précédemment.

2.5 Ajouter ensuite le support des bits de contrôle suivant :
- `pause`, qui fige le compteur lorsque ce bit vaut 1 ;
- `back`, qui faire décroître le compteur lorsque ce bit vaut 1.
- `save`, qui permet de sauvegarder la valeur courante du compteur (toute sauvegarde écrase la précédente) ;
- `restore`, qui restaure la valeur lors de la dernière sauvegarde.

2.6 Que fait le circuit `count16` fourni dans le fichier `register.lgf` ?