

Labo 6 sur QuartusII

VHDL : introduction

Le fichier disponible sur le WEB vous donne un exemple de définition d'un circuit en VHDL. Ce circuit vous permet d'afficher des carrés de diverses tailles et couleurs en fonction de la position de la souris. De plus, de la logique séquentielle vous permet de modifier le jeu des couleurs en cliquant sur le bouton gauche de la souris.

- 1) Réalisez une fonction qui dessine un carré de 25 pixels de côté centré sur le curseur de la souris. Le bouton gauche de la souris est utilisé pour changer la couleur du carré et le bouton droit de la souris est utilisé pour changer la couleur du fond d'écran.
- 2) Soit une forme inscrite dans un carré de 256 pixels de côté. On considère que chaque ligne est constituée d'une seule séquence continue de points colorés. De plus, la forme est symétrique par rapport à l'axe vertical. Pour chaque ligne, il suffit donc de connaître I_x , le premier index du point coloré. Le dernier se déduit en faisant $255 - I_x$. Les 256 index ont été mémorisés dans une ROM (LPM_ROM, fichier « figure.mif »). On demande :
 - a. Écrivez le code VHDL qui affiche la forme selon la position de la souris qui correspond au point (128,0) de la forme.
 - b. Utilisez le bouton de la souris pour réduire la taille de la forme dans les rapports suivants : $1/N$, N variant de 1 à 64. Par soucis de simplicité, considérez qu'il suffit de prendre un point sur N et une ligne sur N .
- 3) Écrivez le code VHDL du jeu suivant : L'écran est divisé en lignes et colonnes de 16 pixels de large.
 - a. Quand on clique avec la souris, la couleur de la ligne et de la colonne correspondante s'inverse (le carré à l'intersection s'inverse deux fois et ne change donc pas de couleur). Le bouton gauche de la souris utilise le rouge tandis que le droit utilise le bleu. Le vert est utilisé pour le fond d'écran.
 - b. En plus du point a), toutes les lignes multiples de la ligne de la souris et toutes les colonnes multiples de la colonne de la souris s'inversent elles aussi. Le jeu se joue à deux : la première personne place le système dans un état quelconque en cliquant au hasard et la deuxième essaie de retrouver l'état initial le plus rapidement possible. Un compteur de temps déclenché sur le bouton poussoir serait aussi utile.
 - c. On peut encore corser le problème en faisant périodiquement défiler le jeu d'une colonne ou d'une ligne (rotation).

Bon travail !