

TP2 : Réalisation d'un clicker

Le but de ce TME est d'utiliser une grande partie des outils de la chaîne de conception de circuit Alliance pour la réalisation d'un petit circuit. L'archive [clicker.tgz](#) contient les fichiers nécessaires pour la réalisation du clicker à 1 afficheur, vous allez devoir étudier cette archive et créer un clicker à 4 afficheurs en utilisant le moins de broches possibles.

Questions

Le fichier Makefile fourni contient les appels des outils d'Alliance pour la simulation et la synthèse logique, mais aussi pour le placement et routage qui seront détaillés dans le cours de Jean-Paul Chaput. L'intérêt de faire le placement et le routage est que vous allez pouvoir voir le circuit. Pour chaque outil d'Alliance, vous avez un manuel (p. ex. `man boom`) Si les manuels ne sont pas installés ils sont sur : [?https://manpages.debian.org/testing/alliance/](https://manpages.debian.org/testing/alliance/)

Le Makefile n'est pas conventionnel dans le sens où il ne tient pas compte de la date des fichiers produits et des fichiers source pour permettre une reconstruction partielle si un fichier source change. C'est plus ici un ensemble de script permettant de réaliser les étapes dans l'ordre. L'appel de `make` sans argument donne une aide.

Les fichiers présents dans l'archive :

__ clicker.vhd	: modèle comportemental du clicker 1 segment
__ clicker.h	: fichier .h préprocessé pour produire la netlist core-plot
__ help.mk	: include pour le help du Makefile
__ Makefile	: Make de construction et de validation
__ patterns	: répertoire contenant les patterns de validation
__ clicker.c	: générateur de patterns
__ clicker.pat	: Patterns écrits à la main pour la validation
__ clickerx.pat	: Même patterns mais le registre interne n'est pas nommé pareil
__ mygenpat.h	: fichier include pour le générateur de patterns

- En étudiant le Makefile, représenter le graphe du flot de conception et pour chaque outil indiquer quels sont les fichiers d'entrée et de sortie. Décrire en quelques mots ce à quoi sert chaque outil.
- Créer un clicker avec 4 segments. Vous devez créer un compteur BCD (Binary-Coded Decimal), c'est à dire que chaque chiffre du compteur est dans un intervalle 0 à 9. Vous ne pourrez pas avoir autant de sortie que de segments, alors vous allez devoir ajouter un compteur de 0 à 3 qui définit le numéro du segment à afficher.
- Essayez de placer-router le nouveau circuit. Vous devrez modifier le fichier `clicker.h`