

Simulateur de netlists

Léon Sagaert

November 16, 2022

Pour simuler l'exécution d'une netlist le programme procède de la manière suivante :

L'utilisateur donne en argument le fichier .net de la netlist à simuler et le nombre d'étapes à exécuter. Les ROM et RAM sont initialisées au début de l'exécution, l'utilisateur doit alors écrire dans la console le chemin vers un fichier .txt pour chacune des mémoires (voir ExempleRAM.txt), ou ne rien écrire pour initialiser avec un tableau contenant que des 0 de bonne dimensions.

Après ça, la simulation peut commencer, et pour chaque étape, l'utilisateur précise la valeur des variables d'entrée, et le programme renvoie la valeur des variables de sortie.

En terme d'implémentation, les mémoires sont représentées par des tableaux, et ces tableaux sont stockés dans une table de hachage qui à chaque nom de variable désignant la sortie d'une lecture/écriture de mémoire associe le tableau représentant cette mémoire. Au début de l'exécution, on parcourt toutes les équations de la netlist pour initialiser les mémoires.

L'environnement est représenté par un dictionnaire qui à chaque nom de variable associe sa valeur.

Au cours d'une étape d'exécution, plusieurs étapes ont lieu :

- On crée un nouvel environnement vide.
- L'utilisateur fournit la valeur des variables d'entrées, qui sont alors ajoutées à l'environnement.
- Les équations sont parcourues dans l'ordre topologique, et on évalue puis ajoute la valeur de la variable de sortie à l'environnement. Si l'on doit écrire dans la RAM, on garde cette instruction en mémoire dans une pile.

- Une fois toutes les équations évaluées, on exécute toutes les instructions d'écriture dans la RAM.
- On peut alors afficher les valeurs des variables de sortie, et passer à l'étape suivante.

Pour ce qui est de la gestion des registres, on garde en mémoire à chaque étape l'environnement précédent, et lorsque l'on doit lire la valeur d'un registre, on la lit non pas dans l'environnement actuel, mais dans l'environnement précédent.

Les registres sont initialisés à 0.

Enfin, le simulateur identifie un tableau de VBit de taille 1 à un seul VBit, et autorise les binop entre tableaux de VBit de même taille.