

Projet Microprocesseur (2016-2017)

RUFFEL Martin

Ordonnanceur de net-list

- Le code de l'ordonnanceur est dans `scheduler.ml`.
- On traitera les registres séparément. Ils ne sont donc pas considérés dans le tri topologique, puis rajoutés de manière arbitraire.
- Pour les opérations de type $x = Eram(_)$ ou $Erom(_)$, on ne considère que l'adresse de lecture pour le graphe de dépendances, les opérations d'écriture se faisant à la fin du cycle.

Simulateur de net-list

J'ai implémenté le simulateur en OCaml.

- Le code du simulateur est dans `simulateur.ml` ; l'exécutable est `simulateur.byte`. Il prend en argument `-n` le nombre de cycles, par défaut 1, puis le nom du fichier au format `.net`.
- On utilise une table de hachage (*Hashtbl*) pour représenter l'environnement (valeur des variables).
- Le simulateur demande les entrées à chaque cycle.
- Lors des opérations sur les nappes de fils (`SELECT`, `CONCAT`, `SLICE`), on convertit d'abord l'entrée en type *TBitArray*, afin de rendre ces opérations possibles sur le type *TBit*.
- On traite les registres séparément du reste des instructions. On n'effectue les changements de valeur dus aux registres qu'à la fin de chaque cycle, une fois les autres opérations effectuées. On crée alors une liste de changements, qu'on applique ensuite. Cela garantit bien que les affectations sont correctes (dans le cas $y = Ereg(x)$ et $z = Ereg(y)$ par exemple). Enfin, on commence par initialiser les valeurs des variables définies par des registres avec *False* (ou une nappe de *False* selon le type).
- Pour représenter la RAM/ROM, j'ai utilisé un tableau de taille fixée (nommé *ram*) par la taille des adresses (nommée *sa*) et la taille des objets (nommée *sw*) supposées uniques au sein d'un programme.

Ainsi, il suffit de convertir l'adresse donnée sous la forme *VBitArray* en décimal, puis d'accéder aux cases souhaitées dans la RAM.

Lors de l'évaluation d'une équation de type $x = Eram(_)$, on se contente de lire la valeur à l'adresse passée en argument, et on sauvegarde dans la liste *changes* les opérations d'écriture, que l'on évalue et effectue à la fin de chaque cycle.

- Une erreur de type (*TBitArray* à la place de *TBit*, ou vice-versa) est signalée par *Failure*(« *unsupported operand type* »)