

Sujet de PSTL : Ribbit sur FPGA avec Eclat

Encadrants : Emmanuel Chailloux et Loïc Sylvestre
Emmanuel.Chailloux@lip6.fr, Loic.Sylvestre@lip6.fr,

Objectifs

Le système Ribbit [1,2] est une implantation légère d'un compilateur Scheme (langage fonctionnel dynamiquement typé) vers une machine virtuelle nommée RVM (Ribbit Virtual Machine) et possédant une boucle REPL (Read-Eval-Print-Loop) pour les systèmes embarqués à faibles ressources comme les micro-contrôleurs pouvant tenir sur 4Ko.

Les FPGA ([3,4] *Field-Programmable Gate Arrays*) sont des circuits numériques configurables par logiciel pour implanter matériellement des calculs sur-mesure. Ces circuits sont particulièrement appréciés pour la spécialisation et la parallélisation de certaines tâches.

La programmation de FPGA se fait habituellement dans un langage de description de matériel (*Hardware Description Language*, HDL) comme VHDL ou Verilog. Les descriptions de matériel sont traduites, par *synthèse*, en un fichier de configuration qui peut alors être « flashé » sur une carte FPGA. Par exemple, un processeur *softcore* est une description de processeur synthétisée et « flashée » sur une carte FPGA ; le circuit résultant est alors capable d'exécuter des logiciels.

Parmi les outils de synthèse, on peut citer Yosis/next-pnr (qui est sont des logiciels libres), Quartus pour les FPGA propriétaires Intel et Vivado pour les FPGA propriétaires AMD.

Les HDL sont très éloignés des langages algorithmiques. Mais certains langages algorithmiques, comme le langage fonctionnel Eclat [5], sont compilés vers du code HDL. Cela permet d'implanter facilement des accélérateurs matériels sur FPGA. Par exemple, Eclat a été utilisé pour implanter la machine virtuelle OCaml et sa bibliothèque d'exécution sur FPGA [6] ; Cette implantation d'OCaml est 50 fois plus rapide qu'une implantation existante ciblant un processeur softcore sur FPGA.

L'objectif de ce PSTL est de porter le système Ribbit sur FPGA en utilisant le langage Eclat pour décrire la machine virtuelle RVM, sa boucle REPL et son compilateur. On pourra pour se faire utiliser des outils libres (Yosis/next-pnr sur une carte OrangeCrab) ou propriétaires (Quartus sur un carte DE-10 Lite) selon les contraintes de synthèse.

Tâches à réaliser

- Etudier le langage Eclat et le système Ribbit
- définir les schémas d'interprétation de la machine virtuelle RVB
- implanter en Eclat cet interprète, et le tester sur Orange Crab ou De-Lite 10
- définir un mode d'appel pour les fonctions externes en circuits
- charger la boucle REPL de Ribbit et son compilateur sur circuit pour interagir avec lui
- tester la chaîne complète de compilation

Prérequis

- Connaissances en programmation fonctionnelle (Ocaml ou Scheme)
- Connaissances de VHDL ou des FPGA sont des plus

Références

- [1] Samuel Yvon and Marc Feeley, A Small Scheme VM, Compiler, and REPL in 4K. In *Workshop on Virtual Machines and Intermediate Languages (VMIL@SPLASH'21)*, 2021.
- [2] Leonard Oest O'Leary and Marc Feeley, A Compact and Extensible Portable Scheme VM. In *MoreVMs Workshop (MOREVMS@PROGRAMMING'23)*, 2023
- [3] Jocelyn. Sérot. La programmation des circuits FPGA et le langage VHDL. Une introduction pour les programmeurs et par l'exemple. Ellipse, 2019.
- [4] cours VLSI 1 de M1 SESI (cours 4) : <https://www-soc.lip6.fr/master-sesi/ue-vlsi1/>
- [5] Loïc Sylvestre, Emmanuel Chailloux, Jocelyn Sérot : "[Work-in-Progress: mixing computation and interaction on FPGA](#)", International Conference on Embedded Software (EMSOFT 2023), 2023.
- [6] [Loïc Sylvestre, Jocelyn Sérot, Emmanuel Chailloux. Hardware implementation of OCaml using a synchronous functional language.](#) In. 26th International Symposium on Practical Aspects of Declarative Languages (PADL'24), 2024.