Certified Compilation of Scade/Lustre

Postdoctoral position at INRIA Futurs, Proval Team
LRI, Université Paris-Sud, Orsay

March 2006

Since its definition at the beginning of the eighties, the synchronous language Lustre has been used by several companies for the implementation of critical real-time systems in various industrial domains (nuclear systems, avionics, public transportation and automotive). All these developments have been done with Scade, a graphical environment whose semantics is based on Lustre and is developped by the Esterel-Technologies company.

The precisely defined semantics of Lustre, its verification tools and its efficient compilation into sequential code make it well suited to the domain of critical embedded systems.

The use of Scade in certain fields like energy or avionics called for a *certification* (or *qualification*) of its code generator (norme DO-178B). This certification is extremely important since it avoids various testing to establish the equivalence between the Scade code and the generated C code, everytime the source code is modified. Nonetheless, this certification is not "formal" (on the mathematical sense) but rather based on the developpment process: description of the whole life-cycle, specification and verification, detailed documentation and the ability to locate errors in the code. It aims at increasing long term maintenability (e.g., 30 years for avionics).

Synchronous languages and their associated compilation techniques are now sufficiently mature to be formally certified by a computer. A certifed Scade/Lustre compiler is in the scope of the current evolution of the DO norm with more formal certification requirements.

The goal of this postdoctoral position is to write a small but realistic compiler of a synchronous data-flow kernel such as Lustre for a (safe) subset of C inside the proof assistant Coq. This work will rely in particular on the work of Boulmé and Hamon [1] on the shallow embedding of Lustre inside Coq. The compilation techniques will be based on the one used in the Scade compiler and formalised in [2].

Directeurs : Christine Paulin (`Christine.Paulin@lri.fr`) and Marc Pouzet (`Marc.Pouzet@lri.fr`)

References

[1] Sylvain Boulmé and Grégoire Hamon. Certifying Synchrony for Free. In *International Conference on Logic for Programming, Artificial Intelligence and Reasoning (LPAR)*, volume 2250, La Havana, Cuba, December 2001. Lecture Notes in Artificial Intelligence, Springer-Verlag. Short version of *A clocked denotational semantics for Lucid-Synchrone in Coq*, available as a Technical Report (LIP6), at `www.lri.fr/~pouzet`.

[2] Paul Caspi and Marc Pouzet. A Co-iterative Characterization of Synchronous Stream Functions. In *Coalgebraic Methods in Computer Science (CMCS'98)*, Electronic Notes in Theoretical Computer Science, 28-29 March 1998. Extended version available as a VERIMAG tech. report no. 97–07 at `www.lri.fr/~pouzet`.