Project 22703: SYRF

Deliverable 6.1.3

Translation to VHDL/Verilog

---

1 Approach

There are natural formal connections between synchronous reactive formalisms and models of synchronous hardware circuits (RTL netlists). These similarities have been recently exploited, both to apply optimisation techniques designed for circuits to synchronous processes, and conversely to give constructive causal meaning to circuits.

Now in the hardware community there already exist celebrated specification formalisms such as VHDL/VERILOG. Despite their numerous field-specific features (which synchronous formalisms currently lack), these languages suffer from informal and ambiguous semantics, specially for the cycle accurate description. So while very restricted subsets of Verilog/VHDL at the so-called "structural" level can be used for useful synthesis and analysis, the general-purpose "behavioural" level is most often only used for early stage simulation. Therefore a translation from synchronous formalisms (at DC level) to "synthesizable" VERILOG could be a mean to combine the advantage of high-level reactive programming and powerful analysis backends. Of course the aim is not so much to replace established languages that are successful in the development cycle with other formalisms, but rather to enjoy the benefits of hardware synthesis tools, even for software production.

2 INRIA Sophia-Antipolis contribution

In this third year we complemented our translation scheme by producing VHDL code similar to the previous Verilog translation. We tested the resulting code by entering several CAD design tools such as SMV, Synopsys FPGA Express or Altera's Max+PlusII.

We presented our translation schemes to a number of individuals from several companies, and this prompted reactions on possible refinement of the translation at so called "behavioural" level, taking advantage of some clear separation between control path and data path that takes place in some synchronous reactive formalisms (mostly Esterel). This work could result in further collaborations.

A technical description of our translation scheme is provided in annex [1]

References

[1] A. Ressouche Compiling synchronous reactive languages into Verilog/VHDL. *Internal Report, Syrf deliverable annex*, 1999.