Title:

FPGA design from mathematical specifications

Abstract

It is widely recognized that mainstream hardware design languages (VHDL, Verilog) have poor abstraction mechanisms, turning FPGA design into a cumbersome activity. Most high level synthesis languages only partially solve this problem, mostly because they take a sequential perspective. CλaSH, on the other hand, starts from a mathematical perspective exploiting the abstraction mechanisms of the functional programming language Haskell, including polymorphism, type derivation, and higher order functions. Since every CλaSH specification is an executable Haskell program, testing and debugging can be done at top level.

Further advantages of a mathematical approach to hardware design are the possibility to transformations on a given design to get an optimal space-time performance.

We will illustrate the approach with several examples.

CV:

Jan Kuper studied logic and mathematics at the University of Twente, where he got his Master degree (with honours) in 1985. In 1994 he received his PhD degree under the supervision of Henk Barendregt on the foundations of mathematics and computer science. He developed a theory of partial functions and generalized it to a theory of binary relations, which both are strong enough to be a foundation of mathematics.

He worked as a lecturer at the University of Leiden, as a researcher at the University of Nijmegen, and he now is an assistant professor at the university of Twente. His main fields of interest are philosophical and mathematical logic, functional programming and hardware specification. Based on the functional programming language Haskell he initiated the design of a mathematical language as a specification language for computer architectures (called CλaSH).

In 2016 he co-founded (together with Dr. Christiaan Baaij) the company QBayLogic for hardware design, in particular FPGAs, with CλaSH.

Dr. Kuper published on the foundations of mathematics, lambda calculus, logic for artificial intelligence, specification languages for computer architectures, and on energy reduction for computer architectures.