Digital circuits in CλaSH

Christiaan Baaij, Jan Kuper, Gerard Smit - Compute Architectures for Embedded Systems group, University of Twente, Enschede, The Netherlands

CλaSH (`http://clash-lang.org/`) is a novel compiler system for generating digital circuits as described by a mathematical/functional specification of the architecture. The specification language is a subset of the functional programming language Haskell (`http://haskell.org`) and offers:

Polymorphism: a hardware component that is useful for different types of data has to be specified only once.

Higher-order functions: regular structures can be expressed very concisely and are well readable.

Pattern matching: definition-by-cases has a compact format and is well readable.

Functional composition: composition of components exploits signals directly without being bothered by port mappings.

Type derivation: types of input and output signals of nested components can be derived automatically from the types of the outer component. Consequently, in almost all cases, only the top-level component needs a type annotation.

We will demonstrate several (interactive) applications written in CλaSH, which are programmed on an embedded device called an FPGA. These applications include:

Tunneling ball device: A ball bearing is dropped down a long shaft, where at the bottom there is a fast spinning metal disk with two holes in it. With a *minimal* amount of acceleration, the metal disk is either sped up or slowed down so that the ball bearing can fall through one of the metal disk's two holes.

Music synthesizer and spectrum analyser: An audio CODEC samples music being played from either an MP3 player or a computer. The audio spectrum is analysed and displayed on a monitor. We can apply several digital filters which affect the music. The effects of these filters can be both seen on the monitor, and heard through the speakers connected to the FPGA board.

Multi-processor system: A multi-processor system developed completely in CλaSH. The system is used in a compiler construction course, where the compiler is written in the Haskell (the language of which CλaSH is a subset). Because CλaSH is proper subset of Haskell, students can build and experiment with the compiler and the multi-processor system in the same environment.

References

[1] Baaij, C.P.R. **Digital circuits in CλaSH: Functional Specifications and Type-Directed Synthesis** (`https://doi.org/10.3990/1.9789036538039`). PhD Thesis, University of Twente, Enschede, The Netherlands, January 2015.