

High-performance Signal and Data Processing: Challenges in Astro- and Particle Physics and Radio Astronomy Instrumentation



Contribution ID: 0

Type: **not specified**

An intuitive Parallel Programming Tool-flow for Software Defined Radio Signal Processing on Heterogeneous Architectures

We present a domain specific language (DSL) for software defined radio (SDR), referred to as SDR-DSL. The main objective of SDR-DSL is to provide an intuitive parallel programming tool-flow (PPTF) for digital signal processing (DSP) algorithms, which combines SDR and parallel programming domains to implement a PPTF that matches high level signal processing abstraction to generic parallel executable patterns. As languages that contain constructs for specific problem spaces, DSLs generally provide substantial gains in expressiveness and ease of use for even complex heterogeneous architectures. We demonstrate our approach in tackling parallel DSP programming complexity issues using an embedded DSL compiler with code generation kernels for multiple heterogeneous targets. We also discuss how a high-level DSP algorithms implementation using SDR-DSL can be validated prior to hardware implementation using software design cycles and functional verification tools.

Summary

SDR-DSL is intended for intuitive yet scalable design and assembly of parallel DSP routines for SDR. Through its data types and execution semantics, SDR-DSL is expected to reflect the desired abstraction hierarchy in SDR. The SDR-DSL compiler, based on Delite DSL framework, provides a constraint system which is used to drive creation of new DSP datapaths for SDR easily. SDR-DSL accomplishes these while being part of a larger tool-chain which supports validation, code generations for heterogeneous architectures, and physical device assembly.

Primary author: Mr MOHAPI, Lerato (University of Cape Town)

Co-authors: Dr MISHRA, Amit (University of Cape Town); Prof. INGGIS, Michael (University of Cape Town); Dr WINBERG, Simon (University of Cape Town)

Presenter: Mr MOHAPI, Lerato (University of Cape Town)