R-Stream 3.0: Technologies for High Level Embedded Application Mapping

Richard Lethin, Directing Engineer
Reservoir Labs, Inc.

R-Stream is a High Level Compiler being developed as part of the DARPA IPTO Polymorphous Computer Architecture Program. The compiler is targeted at the problem of mapping high performance embedded signal / knowledge processing applications, as exemplified by the Lincoln Labs Integrated Radar Tracker (IRT) reference, to polymorphous streaming hardware platforms, as exemplified by Stanford's Smart Memories and University of Texas TRIPS projects, and other commercially emerging chips which provide on-chip multiprocessing with distributed memory and explicit memory and communication through DMA operations.

Our 2.0 version, presented as a poster at HPEC last year, will be performing application mapping via the Streaming Virtual Machine interface to low level compilers (LLC). A separate abstract describing the overall software architecture in the PCA Morphware program is being submitted. This presentation will focus on the implementation and architecture of the high level compiler.

In particular, we will be able to present the performance results and insights from our 2.0 version for IRT running simulated on the reference PCA architectures. The performance results will be accompanied by details of the application transformations that 2.0 will be performing, including granularity selection, high level streaming transformations, and the manner in which we integrate the data parallel front end of IRT with the more task/thread parallel back end. Furthermore, we expect to be able to provide insight into the benefits and limitations of some aspects of the morphware software architecture, in particular the phased HLC/LLC compilation structure, the SVM interface, and the abstraction of architectures into the Streaming Machine Model (SMM) and Hierarchical Machine Model (HMM). We may further be able to accompany this with some performance results for the application mapped to commercial architectures that are emerging with similarities to the PCA architecture class.

The second major part of our presentation is to relate the transformations being performed in 2.0 to the more advanced compiler technologies being developed for the 3.0 version of R-Stream. In particular, while the 2.0 compiler will be performing streaming application transformations drawing on classic loop optimization technology, our 3.0 version will be based upon next generation program representations including Affine Partitioning and Dynamic Single Assignment. Such technologies subsume classic loop optimizations and increase their scope of applicability, albeit with substantial challenges in implementation. We expect to be able to comment on how such technologies must be adapted to the specific area of embedded computing, to the array comprehension extensions for Brook, the issues of dynamic application behavior, and to Polymorphous Hardware.