



Master Thesis – Compiler LLVM for x86

Description

The goal of this project is to make it possible to run a digital signal processing (DSP) dialect of C on the x86 architecture. The project will modify an existing compiler backend for x86, to make it as compatible to the DSP C dialect as possible.

A DSP processor is designed to be used for specialized applications and typically has some features that general purpose processors (e.g. x86) lack, such as fixed-point arithmetic, and different memory spaces. The DSP used by Ericsson also has other differences, such as 16-bit bytes (addressable unit of data storage in C), 16-bit pointers, and big-endian data layout.

A drawback of these differences is that, for most tests, it is not possible to run directly on the machines running the development environment, typically Linux-x86. Tests must be run either in a DSP simulator, on the DSP hardware, or using some tricks to make the C code x86-compatible.

The project will modify the LLVM x86 backend from llvm.org, and use it together with parts (mainly the frontend) from our internal LLVM based compiler, supporting this DSP C dialect. For example, modifications are required to support 16-bit bytes, 16-bit pointers, and big-endian data layout. Some parts, such as fixed-point arithmetic, may be handled by connecting existing emulation libraries.

To support execution on x86 through the LLVM IR interpreter may also be interesting. The project will evaluate the result by comparing with previous solutions for execution of this DSP C dialect on x86.

Qualifications

This is a technically advanced project and you will need the following in order to succeed:

- Knowledge of how a compiler backend work
- Knowledge of x86 assembly.
- Knowledge of C and C++ programming in Linux.

Contact persons:

Patrik Hägglund patrik.h.hagglund@ericsson.com, software developer, Ericsson AB, Linköping +46730436142 or Anders Nilsson: anders.n.nilsson@ericsson.com, section manager, Ericsson AB, Linköping +46730435786