ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institut für Technische Informatik
und Kommunikationsnetze
Computer Engineering and
Networks Laboratory

Prof. L. Thiele

Semester / Master Thesis:

Programming on the Parallella platform

Write a back-end to our process network based programming framework DAL for the "multi-core Raspberry Pi". Find out how to take advantage of the memory hierarchy.

The Parallella is a new and modern low-power platform similar to the Raspberry Pi. It features a dual-core ARM A9 processor running Linux, a powerful FPGA and a 16-core accelerator chip called "Epiphany III". In the Epiphany architecture all the cores are linked through an on-chip network mesh. Each core has its own local memory, which, however, can be accessed from all other cores as well. This ensures fast and reliable communication.

At our group, we have developed a high-level multi-core programming framework called "DAL". DAL is based on process networks, which are specified using C code for the processes and an XML description for the network topology. It supports many different target platforms, like UNIX PCs and PC clusters, the Playstation III processor, graphic cards and several others. However, the Epiphany chip is not supported yet.

Task: In this thesis, the focus will be on the Epiphany chip and how parallel programs can run on it efficiently. The goal would be to have a DAL back-end for the Epiphany chip such that from existing DAL process networks, code can be automatically generated that runs on the Epiphany chip. An additional result could be how the memory should best be organised for fast communication. All this would involve the following steps:

1. Understand the concepts of DAL.
2. Set up the platform and get a few "Hello World" programs running on it.
3. Formulate a concept of how process networks can be executed efficiently on the Epiphany chip and manually port a simple DAL application.
4. Write a code generator for DAL that automatically generates code similar to that created manually before.
5. Compare different implementation possibilities (concerning communication) for performance.

Requirements: You should be familiar with C, C++ and Java. Knowledge about low-level programming (e.g. from microcontrollers) would be an asset.

Interested? Please have a look at http://www.tec.ethz.ch/research.html **and contact us for more details!**

Contacts

- Andreas Tretter: andreas.tretter@tik.ee.ethz.ch, ETZ G81
- Lars Schor: lars.schor@tik.ee.ethz.ch, ETZ G78.1
- Lothar Thiele: thiele@ethz.ch, ETZ G87

Further Reading

- DAL: http://www.dal.ethz.ch/
- Parallella platform: http://www.parallella.com/