Proposed Title of Thesis

Noah Berner

Introduction

It is widely known, that the rate at which memory speed increases is smaller than the pace at which microprocessors improve. This has been true for several decades and is unlikely to change drastically in the future. Thus, memory bandwidth is expected to be an even bigger bottleneck in computation than it already is today.
A trend in recent years has been to look further than traditional multiprocessing and to explore the possibilities of stream processing on FPGAs. Intel, for example, acquired Altera, a leading FPGA manufacturer, in 2015 and NVIDIA has included Tensor Cores in their Volta architecture, which are streaming processors for matrix multiplication.
It is unlikely that FPGAs will replace traditional CPUs as their size limits their capabilities and running complete programs on FPGAs would make the architecture to complicated and thus ruining the benefits seen when running small kernels on them. They are more likely going to serve as separate accelerators which take over tasks that benefit from being offloaded to FPGAs, much like vector processors are used in today's architectures.
What is missing today however, is the possibility to compile an existing program in a high level language to a bitstream for a computer that houses an FPGA and a CPU such that the performance of the overall program is increased when comparing it to a traditional CPU-only run.

Description

In this thesis, I will analyse the performance of two SPEC-benchmarks on FPGAs.
First, the benchmarks will be analysed by running on a traditional CPU to get a baseline and to plan the design of the FPGA layout. In a next step, using HLS, the kernels of the benchmarks will be compiled to an FPGA bitstream and analysed on an FPGA. Once this analysis is done, a further improvement would be to get the whole program to run on a CPU and FPGA hybrid architecture and analyse the performance and data exchange between the CPU and FPGA, to further improve the design. If this analysis goes well, one could consider automating the process for similar types of programs, such that the process of writing a program for a CPU and FPGA hybrid architecture is simplified.

Objectives and goals

The main goal of this thesis is to get closer to a solution for easy programming of a CPU and FPGA hybrid architecture as this type of computer is likely to spread in the future, because the memory wall will force the data to be near the computation units if a performance increase is to be observed.
The thesis will analyse the possibility to port two SPEC benchmarks to such a computer and measure its performance when compared to a traditional CPU based run.

References

Timeline and To-Do List

- **February 2019:**
  Hand in proposal,
  Start analysis of benchmarks on CPU.
- **March 2019:**
  Finish analysis of benchmarks on CPU,
  Use HLS to compile kernel into bitstream for FPGA,
  Analyse performance of kernel on FPGA and tweak it.
- **April 2019:**
  Finish analysis of kernel-only performance on FPGA,
  Try running the whole benchmark on CPU and FPGA.
- **May 2019:**
  Port the running program to Enzian and figure out data-transfer between CPU and FPGA,
  Start automation process.
- **June 2019:**
  Finish automation,
  Start writing thesis.
- **July 2019:**
  Write thesis.
- **August 2019:**
  Finish writing thesis,
  Hand in thesis,
  Hold presentation.