TDT4295 Computer Design Project

Assignment Text

2011

Task: Construct a Massively Parallel Processor

The performance increase available from harvesting Instruction Level Parallelism (ILP) from the serial instruction stream is limited because we have reached the maximum power consumption that can be handled without expensive cooling solutions [1]. Consequently, there is a significant interest in single-chip parallel processor solutions (e.g. [2,3]). The processor cores in commercial multi-core chips are conventional designs and therefore reasonably complex. In this work, your task is to design a processor with massive parallelism and simple processing elements. A key trade-off is the simplicity of the processor cores compared to how easy the resulting machine will be to program.

Your massively parallel processor will be implemented on an FPGA, and you are free to choose your computer architecture. The system should be shown to work with a suitable application. Studying the architecture of the Goodyear MPP [4,5] might be a possible starting point.

Additional requirements

The unit must utilize an Atmel AVR micro controller and a Xilinx FPGA. The budget is 23.000NOK, which must cover components and PCB production. The unit design must adhere to the limits set by the course staff at any given time. Deadlines are given in a separate time schedule.

Evaluation

The project is evaluated by an external examiner based on the project report and an oral presentation of the work as well as a prototype demonstration. One grade will be given to the group as a whole, unless there are significant variations in the amount of effort put into the project.

References

1. Olukotun and Hammond; The Future of Microprocessors; ACM Queue; 2005
2. Bell et al.; TILE64 - Processor: A 64-Core SoC with Mesh Interconnect; ISSCC; 2008
3. Kongetira et al.; Niagara: A 32-way Multithreaded Sparc Processor; IEEE MICRO; 2005
4. Wikipedia; Goodyear MPP; http://en.wikipedia.org/wiki/Goodyear_MPP
5. K. E. Batcher; Design of a Massively Parallel Processor; IEEE Transactions on Computers, 1980