4. Fachgebiet Architektur eingebetteter Systeme
Advanced Computer Architectures SoSe2017

Assignment start: 27.06.2017 Submission deadline: 17.07.2017

Assignment 4 - Simultaneous Multithreading 25 Points

In the final exercise you will investigate the effect of executing two threads simultaneously on the same physical core capable of Simultaneous Multithreading (SMT). SMT allows better utilization of core resources by selecting independent instruction from multiple threads simultaneously. The effectiveness, however, varies depending on the workload the threads are performing. With SMT most core resources are shared (e.g. functional units, reservation stations, reorder buffers, branch predictor entries, etc), only the architectural register files are duplicated. Because the core resources are shared, when threads content for the same resources performance of both threads reduce.

In this assignment one thread is calculating the 40th fibonacci number. Your task is to produce code for the other thread that has *as little as possible influence* on the performance of the fibonacci thread. The code sequences can use all the tricks you can think of except they must be active and running. This means no `sleep`, `printf`, or other system calls that can lead to a blocked thread. Furthermore, no inline assembly is allowed, and the fibonacci thread must not be modified.

The SMT code is available on the course ISIS page. The experiments should be performed on a machine with an Intel Core-i7 3930K. This processor has the Sandy Bridge microarchitecture and the main specification can be found at `http://ark.intel.com/products/63697/Intel-Core-i7-3930K-Processor-12M-Cache-up-to-3_80-GHz`.

To balance the load on the available cores, you will get only one account per group. To get an account on this machine you have to give the student assistant a tubit username of one of the group members during the lab. The student assistant will then create you an account on this machine on which you can login. Also the student assistant will assign you a core number to minimize the sharing of cores during the lab. This core number should be filled as the `CPUID_SMT0` in the source file. The machine is accessible via ssh using: *ssh -p 20222 username@lianli.aes.tu-berlin.de*. To copy the smt.zip file for example you can use: *scp -P 20222 smt.zip username@lianli.aes.tu-berlin.de:/˜*

The deliverable is the source file of the assignment. In the file the `tf_smt1` should be documented with a brief explanation of the choices you made. The grade for this exercise will be determined by the performance of the fibonacci thread (15 points) and your explanation (10 points).