Carnegie Mellon University
School of Computer Science
Computer Science Department

CS740: Computer Architecture

Project Milestone Report

FACS: FPGA Accelerated Multiprocessor Cache Simulator

Michael Papamichael [papamix@cs.cmu.edu]
Wei Yu [wy@andrew.cmu.edu]
Yongjun Jeon [yongjunj@andrew.cmu.edu]

Pittsburgh, PA, November 2007

Group Info:

- **Michael Papamichael** [papamix@cs.cmu.edu]
- **Wei Yu** [wy@andrew.cmu.edu]
- **Yongjun Jeon** [yongjunj@andrew.cmu.edu]

Project Web Page:

http://www.cs.cmu.edu/~mpapamic/projects/facs.html

Major Changes:

We did not have to make any major changes to our original schedule or to our project goals.

Accomplishments So Far:

- Achieved original milestone goal
  - FACS supports coherent L1 caches
  - FACS can be fed static memory reference traces that reside on on-chip memory
- Partially achieved 100% goal
  - FACS replicates the functionality of TraceCMPFlex (the SW cache simulator)
- Partially achieved 125% goal
  - We have verified that FACS replicates the exact behavior of TraceCMPFlex in hardware for small traces (1000s of memory references) that fit in on-chip memory
- Use of BlockRAM (FPGA on-chip memory) instead of discrete registers for keeping statistics
- Fully configurable design with support for checkpointing
  - Cache sizes, associativity, number of processors and other parameters can be easily changed
- Designed with 150% goal in mind
  - We have designed FACS with checkpoint support in mind, but that feature has not been fully implemented or verified yet.

We have also posted some preliminary results on the project website.

Meeting Milestone:

We have already surpassed our milestone and are well on schedule.

Surprises:

We have not encountered any major surprises so far.

Revised Schedule:

At this point we do not need to make any major revisions to our schedule. Our first priority is to finish with the last part of the 100% goal, which is to be able to feed FACS memory reference traces that reside on off-chip DRAM memory. This will enable us to test our cache model with larger traces. After reaching the 100% goal we will aim for the 125% goal and hopefully the 150% goal as well.

Resources Needed:

We have all the necessary resources to complete our project:

- Software Tools
    - Simics simulator
    - Flexus simulator source code
    - TraceCMPFlex source code
    - Modelsim 6.3c
    - Xilinx ISE 9.2i
    - Xilinx EDK 9.2i
    - Xilinx Chipscope 9.2i

- Hardware Resources
    - Xilinx XUP Board
    - BEE2 Board

- Our own Espresso machine ☺