Computer Science 703

Advance Computer Architecture

2010 Semester 1

Lecture Notes

13May10

Hardware Transactional Memory

James Goodman

Department of Computer Science

CS703 Project

- Extend Bochs (functional) simulator to make simple performance measures
  - http://bochs.sourceforge.net/
- Simulate the multicore system from A1 running your programme
- Compare simulation results to actual execution time

Alternative assignment

- Select two Multicore chips and compare/contrast them, e.g.,
  - Target applications (what they would do well, what they would not do well)
  - Number of cores & threads
  - Memory organization: caches, levels, sharing, coherence
  - Interface to outside world
  - Support for multiple chips
- Maximum grade: B

Today's Recommended Readings

- D Dice, Y Lev, M Moir, D Nussbaum, "Early experience with a commercial hardware transactional memory implementation" *ASPLOS09*, pp. 157-168, (**March 2009**).
  - http://research.sun.com/scalable/pubs/ASPLOS2009-RockHTM.pdf
  - See especially section 2: Background
- AMD Advanced Synchronization Facility: Proposed Architectural Specfication, Advanced Micro Devices, (**March 2009**).
  - http://developer.amd.com/assets/45432-ASF_Spec_2.1.pdf

Generic "Best-Effort" HTM

- Atomically executes transactions
  - Speculate, then commit or abort
- Failure **IS** an option (success may not be)
  - Some transactions may always fail
  - Not necessarily predictable which will succeed
  - Specification is difficult/imprecise
  - Litmus test guarantee:

If you write your code as specified in the example, it will succeed

ROCK

- SPARC multicore processor
  - Multiple cores, multiple threads
  - Highly speculative (e.g., *hardware scout*)
- Designed and built
  - Never marketed
  - Never cancelled

ROCK Instructions

- CHKPOINT
  - Begin speculation
  - Looks like a branch instruction but speculates not taken
    - Branch is taken on abort
- COMMIT
- (ABORT)

ROCK Guarantees

Forward progress is guaranteed(?) if

- Cache line read limit not exceeded
- No more than 32 writes
- No interrupts/exceptions
  - No system calls
  - No TLB misses
- No conflicts

AMD: Advanced Synchronization Facility (ASF)

- A proposal for TM support for AMD processors
- Open for public discussion/criticism/proposals

ASF Instructions

- SPECULATE: begin transaction
  - Special status code register can be tested
- COMMIT
- ABORT
- LOCK MOVx, LOCK PREFETCH, LOCK PREFETCHW (first time in atomic section)

2010 YEAR PRESENTATION The University of Auckland | New Zealand

ASF Guarantees

Forward progress is guaranteed if

- Four 64-byte cache lines limit not exceeded
- No interrupts/exceptions
  - No system calls
  - No TLB misses
- No conflicts

2010 YEAR PRESENTATION The University of Auckland | New Zealand

Contention/Conflicts

Who Should Win?

- Oldest transaction (FCFS)
- Most urgent
- Most "Karma"
- Most disruptive

ROCK: Most aggressive

- Requester always wins

2010 YEAR PRESENTATION The University of Auckland | New Zealand

What about Locks?

- Locks are usually a library function
- Replace `lock_acquire()` code with CHKPOINT
  - Must handle case where speculation fails
- Replace `lock_release()code` with COMMIT