CSE471 Homework Assignment #2: Cache Coherence

Due Tuesday, May 3rd before lecture

Introduction

The purpose of this assignment is to further your understanding of cache coherence and to hone your data analysis and technical writing skills. For this assignment, you will go through the process of designing, implementing and evaluating a new cache coherence protocol.

In class we studied a simple 3-state snooping cache coherency protocol. Although two bits are needed to encode the coherency state in this protocol, only three of the four possible values are used. This begs for a coherency protocol which utilizes the fourth value to implement a fourth state that improves multiprocessor performance in some way. In this project you are going to design, justify, implement, and evaluate a particular 4-state protocol.

The states in the 3-state protocol are (1) Exclusive (one-cache, dirty); (2) Shared (possibly many caches, clean); and (3) Invalid (no caches). You will be modifying the 3-state protocol by splitting the Exclusive state into two states: Modified, which is a private, dirty state; and Exclusive, which is a private, clean state.

There are three milestones to this project with separate due dates. You should pick a different partner for this assignment than you had for the last and stick together for all three. The first milestone is explained below.

Milestone 1: Defining and Justifying the Protocol

The first milestone requires you to design the fourth state by augmenting the finite state diagrams for a 3-state protocol. You will submit a mini-report in which you do several things:

Protocol Diagram. Draw the 4-state protocol diagrams. You must show two transition diagrams, one for transitions resulting from actions performed by the local CPU, and one for transitions resulting from requests on the bus. An example of a pair of diagrams like this for the 3-state protocol can be found in Chapter 4, Section 2 of your textbook (Figure 4.6 in the 4th Ed.) and in our class slides.

Design Justification. Describe why such a state provides any performance benefit. For this, it is important to think about situations in which the 3-state protocol differs from the 4-state protocol. Do these situations always lead to performance improvement? Are there situations in which the 4-state protocol is detrimental? This section of the mini-report is intended to be a justification for making the design change to the coherence protocol. You should argue as though you are trying to convince a design team that this change is a good or a bad idea.

Illustrative Example. Finally, you must illustrate one or more situations in which the proposed change to the protocol will have an impact, using snippets of program text. You don't have to write any real code for this part. Instead, draw a diagram showing a sequence of instructions being executed by two processors that has different performance characteristics due to the protocol change. Along with the diagram, be sure to point out exactly what is different and why it matters. Brandon will give you an idea of what this diagram is in section.

A Preview of the Next Milestones. Being clear and thoughtful with your design now is to your benefit – in Milestone 2 you will be implementing your new coherence state in a simulator. Having well-reasoned hypotheses about the impact of your protocol change in this part of the assignment will also benefit you later – Milestone 3 requires you to evaluate the 4-state protocol on real programs and characterize the impact of the new design.