CSE471 Homework Assignment #3, Part #1: Cache Coherence

Due Tuesday, May 12th, before lecture

Introduction

The purpose of this assignment is to further your understanding of cache coherence. 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 that utilizes the fourth value to implement a fourth state that improves multiprocessor performance in some way.

In this project you are going to define, justify, implement, and evaluate a 4-state protocol. The protocol states in the 3-state protocol are (1) Exclusive (dirty); (2) Shared (clean); and (3) Invalid. You will be modifying this 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 two parts to this project with separate due dates.

Defining and Justifying the Protocol

The first part requires you to add the fourth state to the 3-state protocol. You'll submit a mini-report in which you do several things:

Protocol Diagram. Draw the 4-state protocol finite state diagrams (FSM). You must show two diagrams, one for transitions resulting from actions performed by a CPU, and another for transitions resulting from snooping requests on the bus. An example of a pair of diagrams like this for the 3-state protocol can be found in our class slides and in Chapter 5 of your textbook (Figure 5.6 in the 5th Ed.).

Design Justification. Hypothesize why such a state might provide a 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 a 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 diagrams showing a sequence of instructions being executed by two processors that has different performance characteristics under the two protocols. Along with the diagram, be sure to point out exactly what is different and why it matters.

Additional Information. Being clear and thoughtful with your design now is to your benefit. In part 2 of the assignment, you will be implementing your new coherence state in a simulator and evaluating it on real programs. Getting all the arcs in your FSM correct now will make implementing and debugging your design so much easier. And having well reasoned hypotheses about the impact of your protocol change will help guide your analysis of the data you generate in the simulator.