CS-307 Lab Project

Design phase 2: Designing coherent bus based multiprocessor.

Goal: design the hardware for a coherent bus-based multiprocessor interconnect. This design will implement a 2-state coherence protocol. Starting from the baseline bus-based non-coherent processor, you will add a write-through, write-no-allocate cache with a 2-state protocol. Please note that the write-back behavior of the cache will be dropped in this phase. It will come back in the next design phase, when you will design the cache with the MSI protocol.

Baseline: Bus based non-coherent NIOS2

The baseline design is the bus based non-coherent multiprocessor you designed and implemented in the previous phase of the project. The state machines and datapath diagrams for the cache and bus controllers of the baseline are attached to this handout.

New Design: NIOS2 Multiprocessor with 2-state coherence protocol

Figure 1 shows the 2-state coherence protocol that will be implemented. `PrRd` and `PrWr` indicate processor-issued reads and writes, respectively. `BusRd` and `BusWr` indicate operations snooped on the bus.

This is the same protocol presented in class in week 2. You will propose modifications to the baseline design in order to add coherence to it.

The bus protocol will change. Now, the bus will carry only `BusRd` and `BusWr` commands, and, in order to enable snoops, the bus will now notify other caches on the system whenever a command on the bus is valid. This will be done through the `busSnoopValid` signal, and this signal will be asserted whenever a valid operation is happening on the bus. The second modification will be the addition of a second state machine that invalidates the cache lines written by the other processor(s). This state machine will snoop the bus, probe the tag array to see whether the bus is operation on a cache line that is resident in cache, and invalidate that cache line if the other cache is issuing a `BusWr` to a resident cache line.

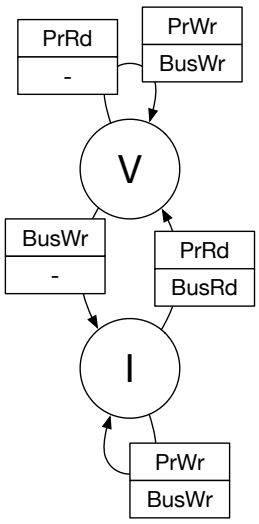

Figure 1. 2-state coherence protocol

Task

You have to create the design specification for the coherent bus-based NIOS2 multiprocessor. Starting from the baseline design you will present:

- Modifications to the datapaths and state machines
- Specification for the new state machine that performs snoops on the bus, and invalidates cache lines when necessary.
- A new interface for the tag array. The new tag array has to support cache line invalidations, which were not supported before. Furthermore, the new tag array has to be probed by 2 state machines, the cache controller and the snooping state machine.
- The registers that must remain and the ones that should be removed.

- You should show, in the state machines, how do you deal with corner cases. For instance, when an invalidation and a processor operation arrive in the same cycle.