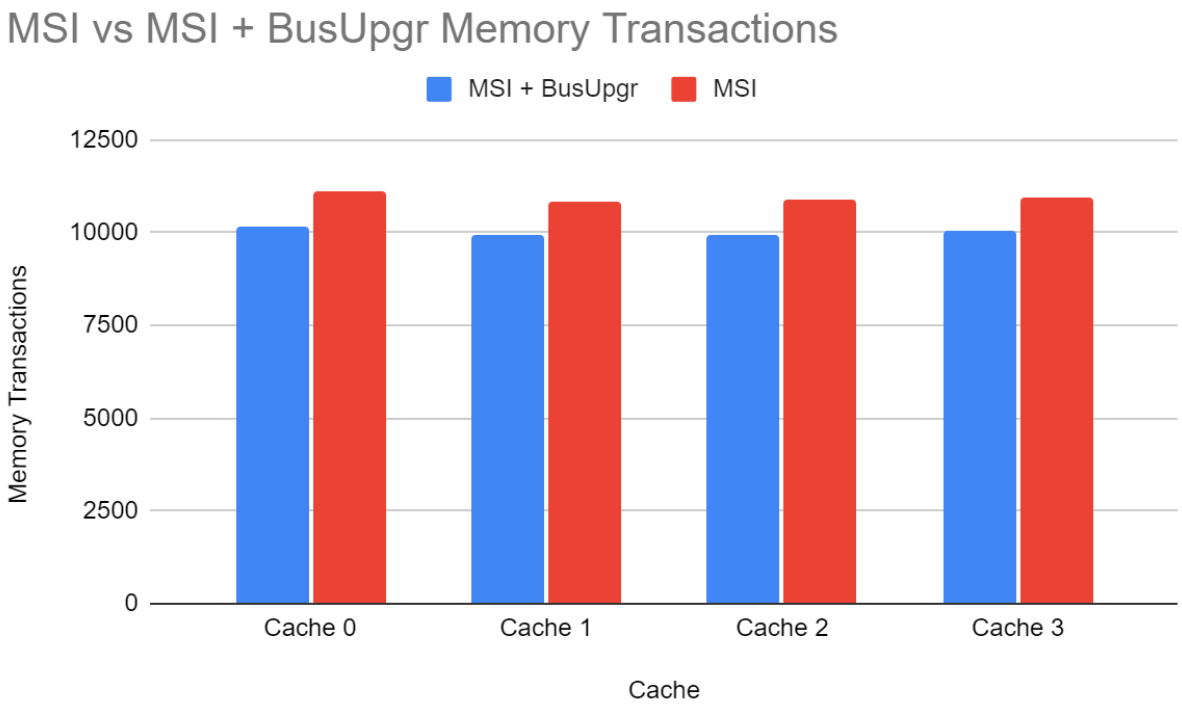
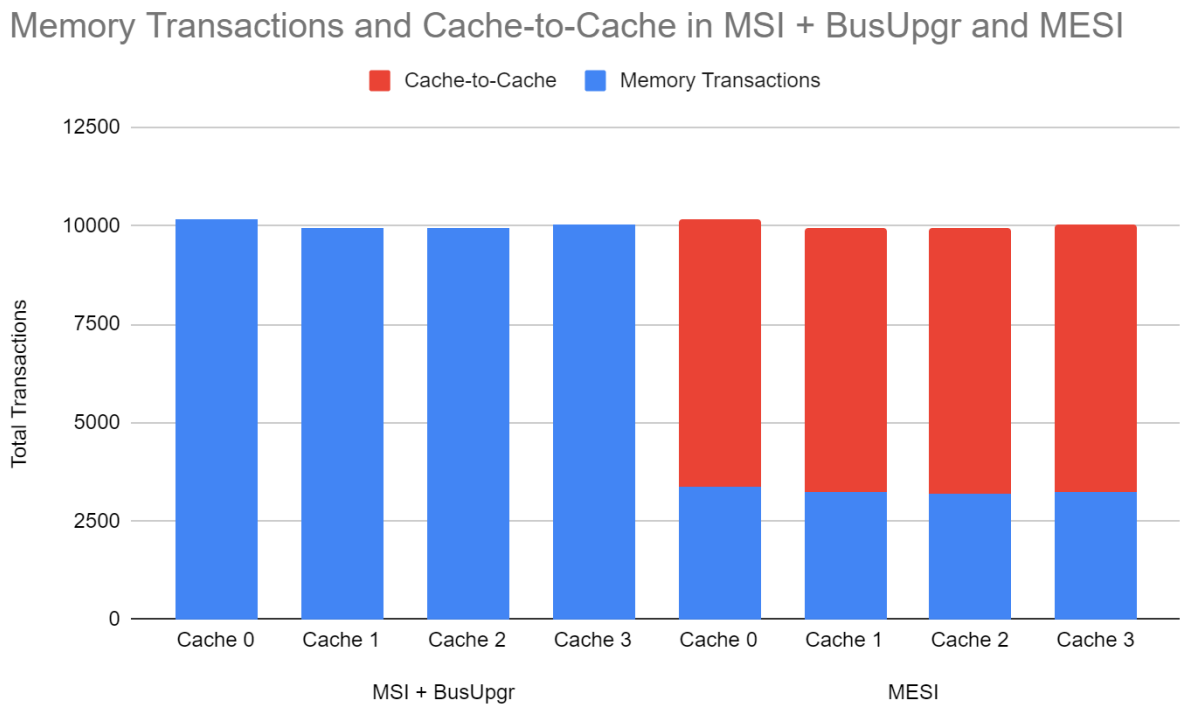
ECE 506
Project 2 Report
Luke Brown

MSI vs MSI + BusUpgr:



There are fewer memory transactions over all caches in MSI + BusUpgr compared to basic MSI. This is because memory transactions are avoided when moving from Shared to Modified states with the added BusUpgr bus transaction. Instead of reading from the memory and then writing the block is just written to and everyone else is notified.

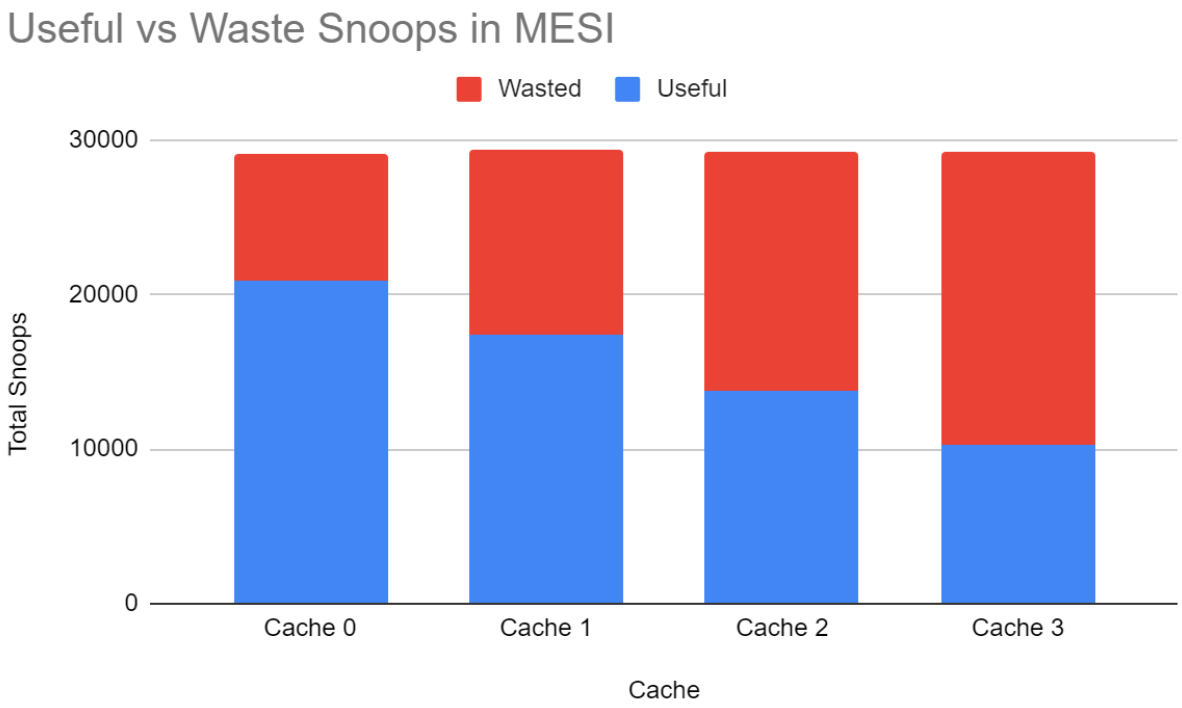
MSI + BusUpgr vs MESI:



There are considerably fewer memory transactions in MESI when compared to MSI + BusUpgr. This is because instead of contacting memory the caches in a MESI protocol have access to cache-to-cache sharing which can be much quicker than contacting main memory. As can be seen in the chart above more than half of the memory transactions were replaced with quicker cache-to-cache transfers. Quicker in a real life scenario, time was not measured in this simulation.

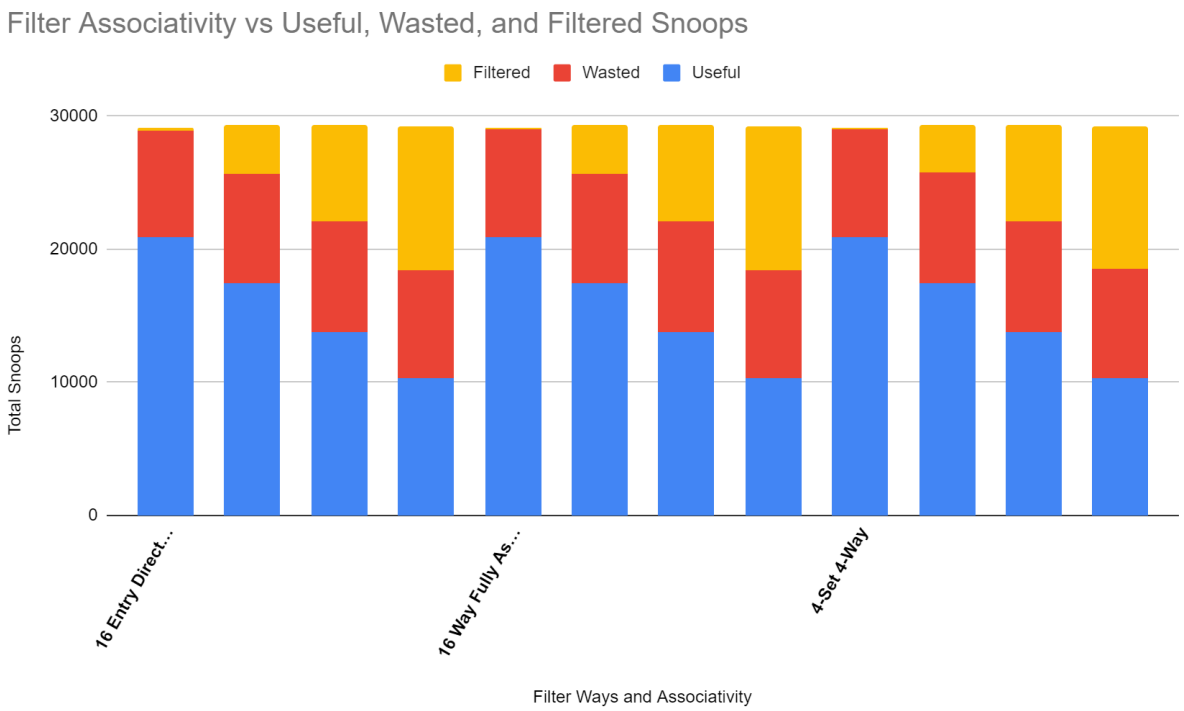
MESI Snoop Filter:

Useful vs Wasted Snoops:



As can be seen in the chart above there are quite a few wasted snoops in a regular MESI protocol. Wasted snoops being those where a cache checks itself and does not contain the requested block.

Associativity vs. Types of Snoops



As can be seen in the chart above the different associativities of the filter seem to have very little impact on how efficient the filter was. Having a filter in general is very useful as can be seen in yellow(filtered snoops). It filters out many snoops and saves wasted time. The most effective filter is the 16-entry Direct Mapped version, followed by the 16-Way Fully Associative version, and finally the 4-set 4-Way version. However, there is so little difference between the effectiveness it would probably be best just to implement the one that was most easily created in hardware.