ECEn 528

Study Guide – Advanced caching

* Read Sections 2.1­2.2 and 2.5­2.6 of H&P  Things to focus on

■ Advantages and disadvantages of the 10 advanced optimizations. Why they work and when they do not work.

 Clarifications

■ Two more techniques:

* If data can be read from a write buffer of a writeback cache, we call it a victim buffer
* If the victim buffer is generalized to also hold clean evicted lines, we call it a victim cache ­­ the effect is to “extend” the associativity of the cache for sets which have high conflicts.

 Answer the following questions:

1. List the advantages and disadvantages of each optimization:

|  |  |  |
| --- | --- | --- |
| *Optimization* | *Advantages* | *Disadvantages* |
| Small and simple | Smaller hit time/power | It’s smaller, larger hit rate |
| Way prediction | Smaller hit time/power | Increased size/complexity |
| Pipelined access | Increased bandwidth, easier to incorporate high associativity | Greater miss penalty, more cycles between load and use (worse hit time) |
| Nonblocking caches | Increased bandwidth, lower miss penalty | Complex, hard to determine miss penalty |
| Banked caches | Increased bandwidth, reduced power consumption | Relies on accesses naturally spreading themselves across banks |
| Critical word first/early restart | Reduced miss penalty | Only benefits large cache blocks, hard to calculate miss penalty |
| Merging write buffers | Reduced miss penalty | I/O address cannot allow write merging |
| Compiler optimizations | Reduced miss rate, no hardware changes | Have to make a better compiler! |
| Hardware prefetching | Reduce miss penalty/rate | Aggressiveness can reduce performance for some applications, increase power |
| Compiler­controlled prefetching | Reduce miss penalty/rate | Requires nonblocking cache, incurs an instruction overhead |

1. Why are “small and simple” designs particularly appropriate for first­level caches?

It keeps hit time small for the L1 caches that are close to the processor.

1. Why is it hard to evaluate the performance of nonblocking caches?

A cache miss doesn’t necessarily stall the processor, so it’s difficult to judge the impact of a single miss.

1. For what kinds of cache access patterns would merging write buffers be particularly effective?

Access patterns with high locality.

1. What happens if you perform blocking with a submatrix size which does not fit in the cache? What if the submatrix size is only 50% of the cache size?

If the submatrix doesn’t fit in the cache, the submatrix values will start to overwrite themselves in the cache. If the submatrix is only 50% of the cache size, then the cache might as well be half the size.

1. Why might hardware prefetching cause a loss of performance?

If the prefetched data isn’t actually used, it’s a waste of time/power

1. Why are non­faulting software prefetches generally preferred to faulting software prefetches?

They turn into no-ops if they would normally result in an exception

1. Why do most of the scientific codes have lower I$ misses and higher D$ misses than most of the integer codes (in Figure 5.20)?