

Homework #5 – Pipeline & Cache Performance

Due: at the start of lecture on Tuesday 07 August 2007

This homework covers pipeline and cache memory performance issues discussed in class. Please check the relevant sections of the Web pages (e.g., Section 5 for pipeline, Section 6 for memory) that contain the performance equations you need to use for this homework. Also, please show all work to get full credit.

Part I – Regular Problems

1. (15 pts) Assume we have a pipelined datapath having $N > 4$ segments (or stages), with a program of $M > N$ instructions, and 23 percent of those instructions are branches. Please calculate the CPI of this pipeline using three different strategies for branch handling that we discussed in class, namely:

- *No Intervention* – Branch Target Address calculated in Stage 2, branch taken or not taken in Stage 3 of the pipeline;
- *Branch Prediction as Branch-Not-Taken*, where we assume that the next instruction following the branch is the one that will be executed as a result of the branch; and
- *Branch Delay Slot*, where we assume that an independent instruction can be taken out of the program and placed in the pipeline after the branch, with a probability of $0 < f_{\text{BDS}} < 1$.

Assuming that there are *no* other effects that decrease pipeline performance (e.g., no stalls or exceptions), the pipeline CPI should be calculated using the following two cases (5 pts each):

(a) *Case 1*: $N = 5$; $M = 12,365$; fraction of branches mispredicted $f_{\text{BE}} = 0.46$; and $f_{\text{BDS}} = 0.53$

(b) *Case 2*: $N = 22$; $M = 12,365$; $f_{\text{BE}} = 0.46$; and $f_{\text{BDS}} = 0.37$

For an additional 5 points, compare and contrast the results of Case 1 versus Case 2, focusing on the effect of the branch delay slot versus the pipeline depth (i.e., number of pipeline stages).

2. (15 pts) Consider a 7-segment pipeline with possible exceptions thrown by the EXE stage at Stage 4, and I/O exceptions that can be thrown by the MEM stage at Stage 5. Given a program with $M = 1,429$ instructions, and the following parameters:

- Fraction of instructions that are branches $f_{BR} = 0.27$
- Fraction of branches mispredicted $f_{BE} = 0.5$
- How are branches dealt with? Branch Delay Slot
- Fraction of instructions able to be put in BDS $f_{BDS} = 0.46$
- Fraction of instructions that are exceptions in EXE $f_{EXE} = 0.002$
- Fraction of instructions that are exceptions in MEM $f_{EXM} = 0.008$
- Cost of handling an EXE-stage exception $H_E = 2$ cycles
- Cost of handling a MEM-stage exception $H_M = 15$ cycles
- Cost of flushing the pipeline for any exception $F = 1$ cycle

Please calculate the following:

- (a) pipeline CPI adjusted for control hazards resulting from branches only;
- (b) pipeline CPI adjusted for control hazards resulting from exceptions (MEM and EXE) only; and
- (c) pipeline CPI adjusted for all control hazards in a) and b), above. (Parts a-c 5 pts each).

3. (20 pts) Assume that a cache memory that stores $N_D = 2\text{Mbytes}$ with $L = 32$ words/block is between the CPU and the main memory, and that the main memory has an address of width $M = 32$ bits. Please answer the following questions (5 pts each), also show all your work and reasoning:

- (a) For *direct-mapped* cache, calculate the width N_T of the tag field in bits;
- (b) Calculate the cache address size in Problem 3a, above, assuming one *valid bit*.
- (c) For an *eight-way set associative* cache, calculate the width of the index, tag, and offset fields in the cache address;
- (d) Given the parameters N_D , L , and M stated at the beginning of this question, as well as memory address time $t_A = 1$ cycle, memory access time $t_M = 10$ cycles per word, and data transmission time $t_D = 1$ cycle / word, compute miss penalty C_{miss} (in cycles) for the direct-mapped cache in 3a, above.

Part II – Extra-Credit Problem

4. (25 pts) Suppose you have a cache memory with the fraction of *load* instructions that generate a cache miss given by $f_{CM} = 0.16$. Let this cache be designed according to parameters $N_D = 1\text{Mbytes}$ with $L = 8$ words/block, 4-way set associativity, and main memory having an address of width $M = 32$ bits. Further suppose that (i) the pipeline described in Problem 2, above, is used to generate an exception in the MEM stage whenever there is a cache miss, and (ii) each cache miss has a penalty C_{miss} calculated using t_A , t_M , and t_D as shown in Problem 3d), above.

>> Calculate the pipeline CPI if 20 percent of the instructions are loads (with f_{CM} of the instructions as cache misses), 18 percent are stores (with 0.02 percent of the store instructions generating an exception in the MEM stage, similar to Problem 2, above), and the remaining instructions are R-format MIPS instructions of which $f_{EXE} = 0.03$ percent generate an exception in the EXE stage.

Hint: Use the approach of Problems 1-3, above to set up the problem steps, and show all your work for each step. State any assumptions that you make, which are not in the description for this extra-credit Problem.
