EECS 370
Final Exam
Winter 2006

Name:

University of Michigan uniqname:
(NOT your student ID number!)

Open book, open notes, calculators allowed. No laptops, PDAs, cell phones, etc. Questions vary in difficulty; it is strongly recommended that you do not spend too much time on any one question. For questions where a box is provided, please put your final answer in the box.

The rules of the Honor Code of the University of Michigan College of Engineering apply for this exam.

Honor code pledge: I have neither given nor received aid on this examination, nor have I concealed any violations of the Honor Code.

Signature:
(examinations without a signed honor pledge will not be graded)

1. True/False *(14 points)*

Circle TRUE or FALSE for each statement.

(a) The use of caller-saved registers should be avoided inside leaf functions (i.e., functions which themselves do not make function calls).

TRUE FALSE

(b) Given any two processors, it is possible to evaluate their relative performance by comparing simply their MIPs rating (million of instructions per second).

TRUE FALSE

(c) The LRU (least-recently used) replacement policy works because programs exhibit spatial locality.

TRUE FALSE

(d) Compulsory misses occur anytime memory is first touched, therefore they cannot be reduced by changing the cache configuration.

TRUE FALSE

(e) Doubling the associativity of a cache without changing its overall capacity or block size will increase the tag size by one bit.

TRUE FALSE

(f) When comparing two pipelines running the same program, the one with the lower CPI will always execute the program faster.

TRUE FALSE

(g) LRU cache replacement always performs better than random replacement.

TRUE FALSE

2. Design changes *(12 points)*

You have just been hired at CoolProcessors, Inc. The company is undergoing a major re-architecture of their main product, GeWeak4. To this end they are evaluating a wide range of design options and they need your help to fully understand the implications of each change they are considering.

For each of the numbered potential design changes listed here, indicate which of the properties labeled (A-H) at the bottom of this page apply. (List ALL that apply; for each numbered item, give the list of letters that are applicable).

(1) Changing from a CISC to RISC instruction set.

_____

(2) Replacing a ripple carry adder (that is on the design's critical path) with a carry-lookahead adder.

_____

(3) The processor is changed from a single-cycle implementation to a pipelined processor implementation.

_____

(4) Adding a pipelined multiplier and an assembly multiply instruction to a pipelined processor which can only emulate multiplications.

_____

(5) Changing the pipelined implementation of the processor from resolving branches in the EX stage to resolving them in the ID stage.

_____

(6) Doubling the associativity of a cache while keeping its size the same, assuming that the cache is on the critical path of the design.

_____

Properties:

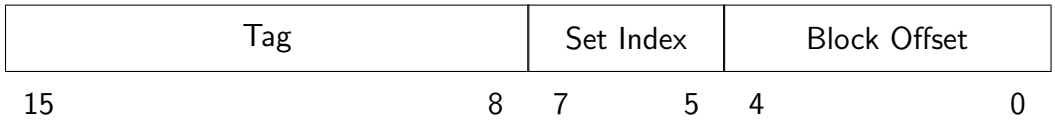
(A) Clock cycle gets longer.
(B) Clock cycle gets shorter.
(C) Average CPI decreases.
(D) Average CPI increases.
(E) Number of instructions per program increases.
(F) Number of instructions per program decreases.
(G) Design size in silicon area increases.
(H) Design size in silicon area decreases.

3. Caches *(15 points)*

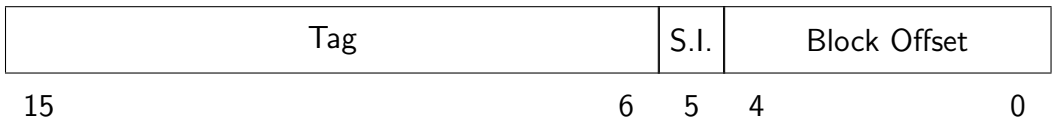
You are brought in as a consultant to analyze the cache performance of a 370Systems, Inc. computer. This is a byte-addressable computer with 16-bit addresses. The system uses separate instruction and data caches, and you are asked to only focus on the data cache performance. The cache is write-allocate and write-back. It is two-way set-associative with an LRU replacement policy. The cache size is 16 KB and the block size is 32 bytes.

(a) Which is the correct partition of address bits into tag, set index, and block offset? (choose one)

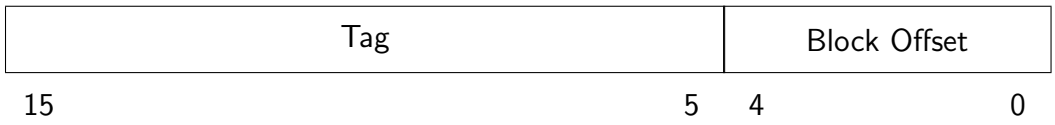
(a)



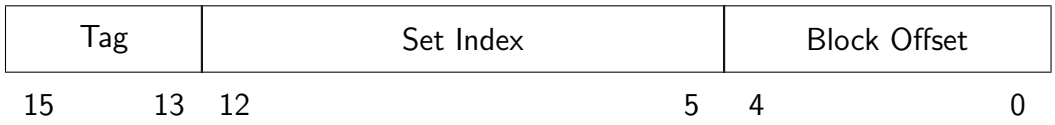
(b)



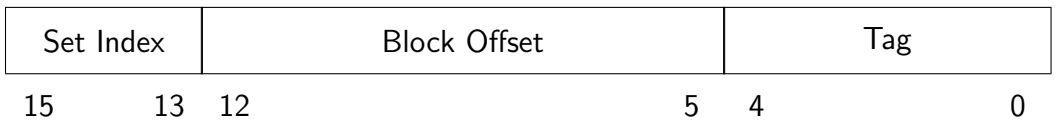
(c)



(d)



(e)



(b) Given the following memory operations (executed in the order shown), indicate which cause a data cache **hit** and which cause a **miss**. Classify each **miss** as one of three types: **capacity**, **compulsory**, or **conflict**. For each memory operation that causes an eviction, record the tag of the evicted block. The cache is initially empty.

Instruction	Hit/Miss	Type of miss	Tag of evicted block
lw 0xCE04			
sw 0x2E1C			
lw 0x8E00			
lw 0x8E0C			
lw 0xCE10			
sw 0x2E08			
lw 0x8E14			
sw 0x2E00			
sw 0xCE08			

(c) How much memory is required for bookkeeping information for this cache (everything but the actual data; tags, valid and dirty bits, and LRU information)

How many bookkeeping bits per set: _____

How many bookkeeping bits total: _____

(d) Assume that the memory operations in part (b) are representative of all the applications that will run on this processor. Based on your observances, what recommendation can you give 370Systems, Inc. to best improve their data cache performance? You can assume all else remains constant.

(a) Double the block size
(b) Double the cache size
(c) Make the cache virtual
(d) Increase the associativity
(e) No change is necessary

4. Virtual memory *(8 points)*

A new generation U-tanium processor has a 32-bit virtual address space and it can support up to 512 MB of byte addressable RAM. The machine has single level page tables and the page size is set to 4KB. Each entry of the page table contains an additional byte with control information (dirty, valid, etc.).

(a) What is the size of one page table entry in bytes? Remember that page table entries are byte aligned.

(b) How many pages does the page table occupy?

(c) How many programs can run simultaneously on this machine? Each program has its own page table which is kept in memory at all times. Assume at least 128MB of RAM must be devoted to data (not page tables).

(d) Increasing page size to 32KB will make the page table

(a) 4 times larger
(b) 8 times larger
(c) unchanged
(d) 8 times smaller
(e) 4 times smaller
(f) none of the above

5. Performance tradeoffs *(10 points)*

(a) The performance of a certain program running on an LC2 pipeline (exactly as in project 3) is observed to be exactly 1.5 CPI. The pipeline is modified to add caches; the instruction and data memories in the pipeline are replaced by separate instruction and data caches. Cache misses are handled by stalling the pipeline. The caches have the following characteristics

- 2-way set associative with 8 word block size
- Hit/miss is determined in one cycle, and data from a hit is available at the end of that cycle.
- Data cache is write allocate and write back, and has a total size of 256 words.
- Instruction cache is read only, and has a total size of 512 words.
- 1% of instructions produce an instruction cache miss.
- 2% of instructions produce a data cache miss.
- Starting after the initial cycle in which a miss is detected, a miss takes take 12 additional cycles if the evicted block is clean or 16 additional cycles if the evicted block is dirty.
- 25% of data cache misses require the eviction of a dirty block.

How much does the CPI increase as a result of cache misses? (Show your work for partial credit.)

(b) To improve performance, doubling the associativity of the data cache is considered. It is determined that this would decrease the fraction of instructions that miss in the data cache from 2% to 1%. However, it would also require a 10% increase in the machine's clock cycle time. Would this change produce a faster processor? (For partial credit, you must justify your answer with supporting numbers.)

6. Pipelining Virtual Memory *(16 points)*

For all parts of this problem, SHOW YOUR WORK to be eligible for partial credit.

The ELC2 computer is an enhanced LC2 with virtual memory. This system is constructed by modifying the standard pipelined LC2. The instruction memory and data memory are replaced by physically addressed caches. Two new pipeline stages are added immediately before these cache stages to do the virtual to physical address translations. There are separate TLBs for instructions and data in these new stages. All misses are handled by stalling. All data hazards are handled by forwarding whenever possible. Branches are handled by predicting not taken and squashing if necessary. Thus, the ELC2 pipeline stages are as follows:

IT: Instruction translation - the instruction address is translated using the instruction TLB.
IF: Instruction fetch - the instruction is fetched using the instruction cache.
ID: Instruction decode - registers are read
EX: Execute - ALU operations are performed and branches are decided
AT: Address translation - the data memory address is translated using the data TLB.
MEM: Memory - the data memory is read or written, using the data cache.
WB: Write back - the destination register value is written.

(a) In the EX stage of the ELC2, when computing the address for a LW or SW instruction, should the ALU be working with a virtual or a physical address?

(b) Should the adder that computes branch target addresses for the ELC2 use virtual or physical addresses?

(c) What is the earliest stage in the ELC2 pipeline in which the adder that computes branch target addresses could be placed?

(d) A program is run on the ELC2 with the following characteristics:

- TLBs and cache miss rates are so low you can ignore them
- Stalls from LW followed by dependent instructions are so infrequent you can ignore them.
- 20% of instructions are branches
- 40% of branches are taken

What is the CPI?

(e) A program is run on the ELC2 with the following characteristics:

- TLBs and cache miss rates are so low you can ignore them
- Taken branches are so infrequent you can ignore them
- 50% of instructions are LW
- 10% of LW instructions are immediately followed by a dependent instruction (example: lw 0 6 data; add 6 6 6)
- 30% of LW instructions are not immediately followed by a dependent instruction, but they have a dependent instruction 2 instructions later. (example: lw 0 6 data; add 1 1 1; add 6 6 6)
- 20% of LW instructions are not followed by a dependent instruction 1 or 2 instructions later, but they have a dependent instruction 3 instructions later (example: lw 0 6 data; add 1 1 1; add 2 2 2; add 6 6 6)

What is the CPI?

7. More Pipelining Virtual Memory *(15 points)*

Consider again the ELC2 machine as described in problem 6.

(a) An ELC2 is built from components with worst case timings as follows:

- Register file read or write: 2 ns
- ALU operation: 1 ns
- Access time for any cache or TLB (for a hit): 5 ns
- Cache miss penalty (for either cache): 50 ns
- TLB miss penalty (for either TLB): 100 ns

What is the maximum clock speed of this processor in MHz?

(b) A program with the following characteristics is run on the ELC2 with timings as described in part (a).

- Taken branches are so infrequent you can ignore them
- No LW instructions are ever followed by dependent instructions that require stalling
- 1% of instructions produce an instruction TLB miss
- 2% of instructions produce a data TLB miss
- 3% of instructions produce an instruction cache miss
- 6% of instructions produce a data cache miss

How many MIPS will this program run at?

(c) If an ELC2 is built with a page size of 256 words and a cache size of 1024 words, what caches would have the property that only untranslated bits are used for the set index? Choose the ONE BEST CHOICE below:

(a) Only caches with a block size of 4 words
(b) All caches with a block size of no more than 4 words
(c) All caches with a block size of at least 4 words
(d) All direct mapped caches
(e) All caches with at least 4 sets
(f) Only 4 way set associative caches
(g) All caches that are n-way set associative, where n is at most 4
(h) All caches that are n-way set associative, where n is at least 4

(d) Will the performance of an ELC2 with the pipeline as described in this question benefit from having only untranslated bits used for the set index? (choose one)

YES NO

If your answer **YES**, estimate how large the performance improvement would be; you may state your answer in whatever units you think are appropriate. If your answer is **NO**, explain very briefly (a sentence or two) why not.

8. Floating point arithmetic *(10 points)*

(a) The decimal value 0.09765625 is loaded into a single precision floating point register on a MIPS processor. What is the representation of this number in the floating point register (show your answer in hexadecimal)? Hint: $25/256 = 0.09765625$

(b) A second single precision floating point point register on this processor contains the floating point number represented by the hexadecimal value 0xbdcb4400. These two floating point registers are added together. What is the hexadecimal representation of the sum?