CHALMERS

Exam in DAT 105 Computer Architecture

Time: December 18, 2008 before and after lunch in the V building

Person in charge of the exam: Thomas Lundqvist, 070-725 0935

Supporting material/tools: Chalmers approved calculator.

Exam Review: January 12, 2009 between 10-12. Location will be announced on the web.

Grading intervals:

- **Fail**: Result < 24
- **Grade 3**: 24 <= Result < 36
- **Grade 4:** 36 <= Result < 48
- **Grade 5:** 48 <= Result <= 64

Important note: Answers should be given in English

GOOD LUCK!

Thomas Lundqvist and Per Stenström

Version: 20081217c

General disclaimer: If you feel that sufficient facts are not provided to solve a problem, either (1) ask the teacher when he visits the exam, or (2) make your own additional assumptions. Additional assumptions will be accepted if they are reasonable and required to solve the problem. Always make sure to motivate your answers.

ASSIGNMENT 1

Consider the MIPS program in Figure 1 which operates on an array with 64-bit elements. The register R1 points to the beginning of the array from the beginning and R2 the end. The array always contains 1000 elements.

You are using this program as a benchmark when comparing two design options. Your **basic design** is a single-issue pipeline where each instruction executes in 1 cycle except LD, SD, DMUL, and BNE. The LD and SD takes 10 cycles extra penalty if a data cache miss occurs. The DMUL and the branch, BNE, instructions both takes 2 cycles to execute. The miss rate for the data cache has been measured to be 12.5%. No other stalls occur.

You are considering the option of **increasing the clock frequency** by 20%. This would however hurt some of the latencies. The BNE would be unchanged, but the DMUL would need yet an another extra cycle which would make it execute in 3 cycles instead. The data cache miss time would stay the same measured in absolute time (but the actual stall cycles for LD and SD would increase).

A) What is the expected speedup when using the high frequency design compared to the basic design? (4 points)

B) Explain Ahmdal's law and use it to calculate the additional speedup obtained if we could halve the data cache miss stall time in the high frequency design. (4 points)

C) Another option would be to reduce the miss rate by using a larger data cache. This would however influence the yield. Explain the term yield when manufacturing integrated circuits. What influences the yield and what is its relation to cost? (2 points)

D) Instead of increasing the clock frequency to get higher performance, one option would be to parallelize the program and run it on two processor cores with lower clock frequency instead. What is the general trade-offs between these options when looking specifically at power use? (2 points)

```
   ANDI  R3, R3, 0      # R3 = 0
LOOP:
   LD    R4, 0(R1)
   DMUL  R5, R4, R4
   DADD  R5, R3, R5
   SD    R5, 0(R1)      # new[i] = old[i-1] + old[i]*old[i]
   DADDI R3, R4, 0
   DADDI R1, R1, 8
   BNE   R1, R2, LOOP
```

Figure 1: Example MIPS program.

ASSIGNMENT 2

Consider again the MIPS program in Figure 1 (see previous page).

A) Generally, you can find three different types of dependencies in programs. Explain these different dependencies and give at least one example for each one in the program in Figure 1. (3 points)

B) For data hazards, you often find the terms RAW, WAR, and WAW. What does RAW, WAW, and WAR stand for? Try to give examples of each type of data hazard in the program in Figure 1. If you think there are no examples, state that explicitly. (3 points)

C) Two techniques often used by compilers are instruction scheduling and loop unrolling. Show how the loop in the program in Figure 1 can be unrolled and how the instructions can be rescheduled. In your new program, point to the specific changes you have made that you think will improve the performance and explain why the performance will be improved. (6 points)

ASSIGNMENT 3

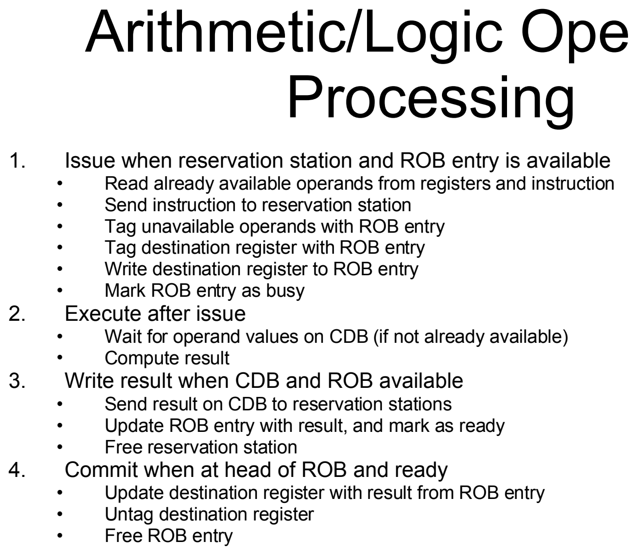
This assignment focuses on processors with dynamic scheduling, speculative execution using a reorder buffer, and dynamic branch prediction.

A) Explain how dynamic branch prediction works when using a 2-bit prediction scheme. Use the program in Figure 1 to demonstrate how the prediction works. Also, describe how the prediction works for a program containing two or more branches? (4 points)

B) In what way does a reorder buffer help speculation? What is the key points when introducing support for speculation? (2 points)

C) In a processor with dynamic scheduling according to Tomasulo's algorithm, hardware based speculation using a reorder buffer (ROB), and dynamic branch prediction, execution of each instruction follows a sequence of steps that is somewhat different depending on the type of instruction. Figure 2 shows the steps carried out for an ALU instruction. What are the corresponding steps for handling a store instruction? (3 points)

D) Explain how RAW hazards are resolved in the basic Tomasulo's algorithm. (3 points)



Arithmetic/Logic Operation Processing

1. Issue when reservation station and ROB entry is available
   - Read already available operands from registers and instruction
   - Send instruction to reservation station
   - Tag unavailable operands with ROB entry
   - Tag destination register with ROB entry
   - Write destination register to ROB entry
   - Mark ROB entry as busy
2. Execute after issue
   - Wait for operand values on CDB (if not already available)
   - Compute result
3. Write result when CDB and ROB available
   - Send result on CDB to reservation stations
   - Update ROB entry with result, and mark as ready
   - Free reservation station
4. Commit when at head of ROB and ready
   - Update destination register with result from ROB entry
   - Untag destination register
   - Free ROB entry

Figure 2: The steps needed for an ALU instruction (used in assignment 3).

ASSIGNMENT 4

A) Describe the advantages and disadvantages with having a virtual indexed first-level cache. (3 points)

B) Describe two cache memory optimization techniques that may reduce miss rate, and define the miss type (compulsory, capacity, conflict) that is primarily affected by each technique. Try to use the program in Figure 1 to explain the optimization techniques. (4 points)

C) Why can the virtual memory system be a performance problem when using virtualization techniques to run many guest operating systems on the same physical machine. (3 points)

D) What does RAID stand for? Explain the technique. (2 points)

ASSIGNMENT 5

A) As Figure 3.1 below shows, the ILP available in many applications can be fairly high. Nevertheless, in practical processor implementations it can be hard to exploit this available ILP. List at least three factors that might limit the exploitable ILP and explain why they limit the ILP. (3 points)

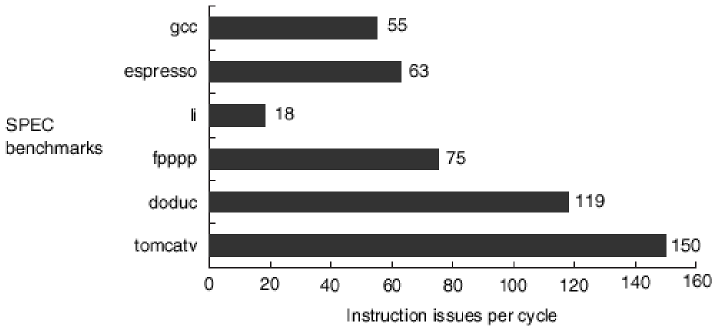


Figure 3.1 ILP available in a perfect processor for six of the SPEC92 benchmarks. The first three programs are integer programs, and the last three are floating-point programs. The floating-point programs are loop-intensive and have large amounts of loop-level parallelism.

(picture from book Computer Architecture, A Quantitative Approach)

B) What structures in a superscalar processor must be replicated to realize a simultaneous multithreaded processor? (2 points)

Shared-memory multiprocessors is an important class of architectures that form the basis for multi-core microprocessors. The memory model is such that all processors access the same memory.

C) What is cache coherence? Give an example of what can happen if cache coherence is missing. (3 points)

D) A commonly used cache coherence protocol relies on snooping and invalidations. Below (next page) you find a list of requests that arrive to the cache coherence mechanism. Connect all requests, A-N, with the correct cache action and explanation, 1-14. Hint: each request matches exactly one action/explanation. Your answer should be a table listing all connections like A-3, B-2, C-8, etc... (4 points)

Request	Source	State of addressed cache block	
Read hit	Processor	shared or modified	**A**
Read miss		invalid	**B**
Read miss		shared	**C**
Read miss		modified	**D**
Write hit		modified	**E**
Write hit		shared	**F**
Write miss		invalid	**G**
Write miss		shared	**H**
Write miss		modified	**I**
Read miss	Bus	shared	**J**
Read miss		modified	**K**
Invalidate		shared	**L**
Write miss		shared	**M**
Write miss		modified	**N**

Assignment 5D: List of requests.

	Type of cache action	Function and explanation
1	normal hit	Write data in cache.
2	coherence	Place invalidate on bus.
3	coherence	Attempt to write block that is shared; invalidate the cache block
4	normal hit	Read data in cache.
5	replacement	Address conflict miss: write back block, then place write miss on bus.
6	normal miss	Place read miss on bus.
7	replacement	Address conflict miss: write back block, then place read miss on bus.
8	normal miss	Place write miss on bus.
9	coherence	Attempt to write shared block; invalidate the block.
10	coherence	Attempt to write block that is exclusive elsewhere: write back the cache block and make its state invalid
11	coherence	Attempt to share data: place cache block on bus and change state to shared.
12	replacement	Address conflict miss: place write miss on bus.
13	replacement	Address conflict miss: place read miss on bus.
14	no action	Allow memory to service read miss.

Assignment 5D: List of actions and explanations

****** GOOD LUCK! ******