Tentamen i Datorarkitektur PB

Justin Pearson

13/6 2001 9-14

Cover Sheet

This sheet should be handed in together with the exam.

Each problem must be solved on a separate sheet. Write your name on each sheet. Indicate below which questions you have answered.

Problem no.	Solution provided	Max	Your points
1		9	
2		6	
3		9	
4		12	
5		20	
Total:		56	

Name : . . . .

Pers.no. : . . . .

Exam Rubric

All answers should be written in English or Swedish (English is preferred).
A mark of 60% is required for a G a mark of 85% is required for a VG.

Hjälpmedel:

Pen,pencil, ruler,rubber, dictionary. You are not allowed a calculator.

1. Simple MIPS coding

   (a) Code the following fragment of code in MIPS assembly language
       (be sure to add in comments):

       ```
       for(i=0 ; i!=100 ; i++) {
        A[i] = A[i] + B[i+1];
        B[i] = A[i] - B[i+1];
       }
       ```

       You should assume that the arrays A and B are integer arrays. In
       you answer use $s0 as the base address of the array A and $s1 as
       the base address of the array B. (**points 4**)

   (b) Explain how and why you would use the slt instruction. (**points 2**)

   (c) Make the following code more efficient by writing it to avoid the
       use of multiply instructions (you may alter the contents of the
       register $s0) and by removing the jump instruction: (**points 3**)

       ```
             li $t0,0
             li $t1,100
       loop: beq $t0,$t1,exit
             muli $t2,$t0,8
             add $t2,$s0,$t2
             sw $0,0($t2)
             addi $t0,$t0,2
             j loop
       exit:
       ```

2. **Question on I/O**

   (a) What is an interrupt? (**points 2**)

   (b) Explain what Polled and Interrupt driven I/O is. (**points 2**)

   (c) Explain the advantages and Disadvantages of Polled and Interrupt
       driven I/O. (**points 2**)

3. Processor Implementation

   (a) Explain how the longest path in a circuit effects the timing of a
       circuit. (**points 1**)

   (b) Explain the problems with a single-cycle implementation of a pro-
       cessor. (**points 2**)

(c) When implementing a processor using the multi-cycle approach a finite state machine is needed. First explain what a finite state-machine is, then explain how a finite state machine can be implemented using Microcode. (**points 4**)

(d) What are the advantages and disadvantages of microcode? (**points 2**)

4. Pipelines

(a) Explain in detail the principle of pipelining, in particular explain how it enhances the performance of a processor. (**points 6**)

(b) What are pipeline stalls and what causes them. (**points 4**)

(c) Assuming your processor has data-forwarding of arithmetic instructions identify all the pipeline stalls in the following piece of code:

```
lw $t0,4($s0)
sw $t0,0($t0)
lw $t1,0($s0)
addi $t1,$t1,4
lw $t1,0($t1)
```

(**1 points**)

(d) Rewrite the code to minimise the number of pipeline stalls. (**1 points**)

5. Caches and Virtual Memory

(a) What it a memory Cache? What principles does it rely on to increase the performance? performance. (**4 points**)

(b) Describe the memory access characteristics of programs that perform badly on cache based systems. (**2 points**).

(c) What are multi-level Caches? (**1 points**)

(d) Given a memory reference a Cache has to determine if the memory is in the Cache. There are a number of schemes for determining where a global memory reference should be placed in the cache memory. Describe the following Schemes: Direct mapped, n-way associative and fully associative. Describe the advantages and disadvantages of each different scheme. (**4 points**)

(e) Explain Virtual memory, explain what is is, the role of a page table and virtual addresses. (**points 4**)

(f) With a virtual memory system why is it inefficient to have a small page size in a virtual memory system.

(g) Explain what a Translation look-aside buffer (TLB) is and why it is needed. (**points 3**).