

科目：計算機結構 (A)

日期：97 年 7 月 24 日 第 1 頁 共 4 頁

請“✓”明 ✓不可看書 可看書

* 請將答案依題號順序寫入答案卷。(1-5 題請作答於 A 答案卷，6-9 題請作答於 B 答案卷)

* 答題時字跡需工整，否則不予計分。Write your answers legibly, otherwise you will get zero score.

1. (3%) What technological forces have caused Intel, AMD, Sun, and others to start putting multiple processors on a chip?
2. (5%) Compute speedups for a program that is 85% vectorizable for a system with 4, 8 and 16 processors. What would be a reasonable number of processors (achieves the most parallel efficiency) to build into a system for running such an application? (5%)
3. (6%) (a) What are the motivations and advantages of building a trace cache for instructions?
(b) What are the disadvantages of building a trace cache for instructions?
(c) Some recent processors have implemented trace caches (e.g., Pentium IV) and some have used conventional instruction caches. When, if at all, do you expect future chips to employ trace caches? Justify your answer.
4. (18%) Consider a single processor system with the following specification:
 - Data cache size is 1 KB (for data) and is 4-way set-associative. Its block size is 16 bytes. It is physically indexed and physically tagged. It uses LRU for replacement within a set and a write allocate write-back policy for writes.
 - 2-way set-associative TLB with 32 total entries and an LRU replacement policy.
 - Physical addresses of 24 bits.
 - Virtual addresses of 32 bits.
 - Byte addressable memory.
 - Page size is 64 KB.

For each field listed below, indicate the bits of the virtual address that correspond to it. Show your work (No point is given if no work is shown in deriving the answer).

- (a) The virtual page offset:
- (b) The virtual page number:
- (c) The TLB index:
- (d) The TLB tag:
- (e) The physical page offset:
- (f) The physical page number:
- (g) The cache block offset:
- (h) The cache index:
- (i) The cache tag:

科目：計算機結構 (A)

日期：97 年 7 月 24 日 第 2 頁 共 4 頁

5. (18%) Suppose that we have single-issue, *in-order* pipeline with one fetch stage, one decode stage, multiple execution stages (which include memory access) and a single write-back stage. Assume that it has the following execution latencies (i.e. the number of stages that it takes to compute a value): **multf** (4 cycles), **addf** (3 cycles), **divf** (6 cycles), integer ops (1 cycle). Assume full bypassing and **two cycles** to perform memory accesses, i.e. loads and stores take a total of 3 cycles to execute (including address computation). Finally, branch conditions are computed by the first execution stage (integer execution unit).
- (a) Assume that this pipeline consists of a *single linear sequence* of stages in which later stages serve as no-ops for shorter operations. You should do the following on your diagram:
1. Draw each stage of the pipeline as a box and name each of the stages. Stages may have multiple functions: i.e. an execute stage + memory op. You will have a total of 9 stages.
 2. Describe what is computed in each stage (e.g. EX1: Integer Ops, Address Compute, First stage of ...)
- (b) How many extra instructions are required between each of these instruction combinations to avoid stalls (i.e. assume that the second instruction uses a value from the first). Be careful!
- | | |
|--|---|
| Between a divf and a store : | Between a multf and an addf : |
| Between a load and a multf : | Between an addf and a divf : |
| Between two integer instructions: | Between an integer op and a store : |
- (c) How many branch delay slots does this machine have? Explain.
- (d) In the 5-stage pipeline shown in textbook, a load into a register followed by an immediate store of that register to memory would not require any stalls, i.e. the following sequence could run *without* stalls:
- ```
lw r4, 0(r2)
sw r4, 0(r3)
```
- Explain why this was true for the 5-stage pipeline.
- (e) Is this still true for your superpipelined processor? Explain.

科目：計算機結構 (B)

日期：97 年 7 月 24 日 第 3 頁 共 4 頁

\* 下列題目請作答於 B 答案卷。

6. (12 points) Given the code sequence:

```

DIV.D F0,F2,F4
ADD.D F6,F0,F8
S.D F6,0(R1)
SUB.D F8,F10,F14
L.D F6,F10,F8

```

please identify all **true data dependences**, **output dependences**, and **antidependences**. How to solve, if possible, these data hazards?

7. (12 points) Depict the purpose of correlating branch predictors. Draw a (3,2) branch-prediction buffer that uses a 3-bit global history to choose from among eight predictors for each branch address and explain it.

8. (12 points) Depict three-step improvements from superscalar to SMT (simultaneous multithreading).

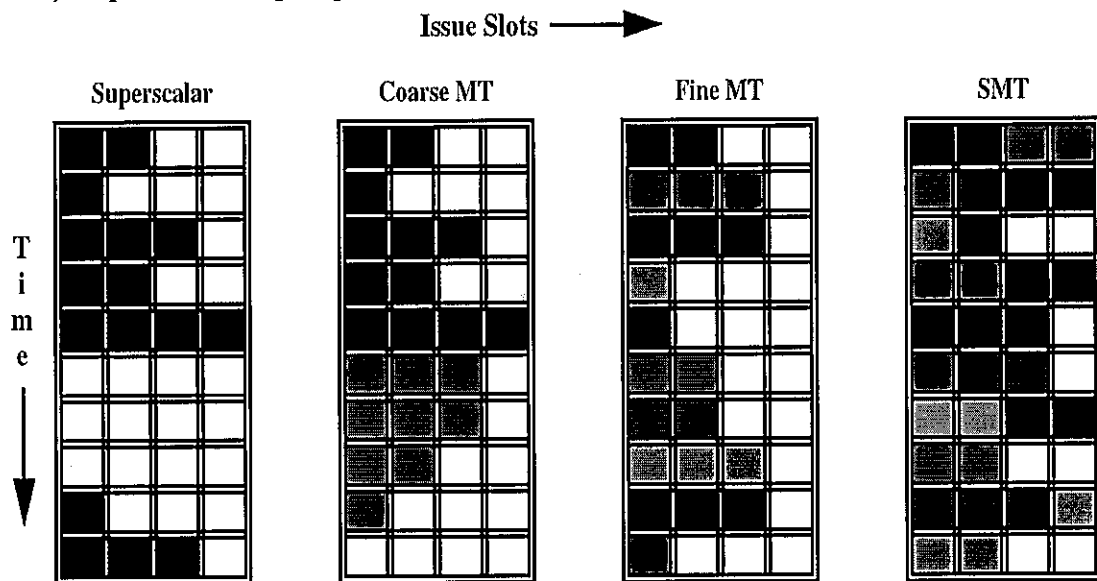


FIGURE 6.44 This illustration shows how these four different approaches use the issue slots of a superscalar processor. The horizontal dimension represents the instruction issue capability in each clock cycle. The vertical dimension represents a sequence of clock cycles. An empty (white) box indicates that the corresponding issue slot is unused in that clock cycle. The shades of grey and black correspond to four different threads in the multithreading processors. Black is also used to indicate the occupied issue slots in the case of the superscalar without multithreading support.

科目：計算機結構 (B)

日期：97 年 7 月 24 日 第 4 頁 共 4 頁

9. (14 points) Assume that words  $x_1$  and  $x_2$  are in the same cache block, which is in the shared state in the caches of both P1 and P2. Assuming the following sequence of events, identify each miss as a **true sharing miss**, a **false sharing miss**, or a **hit**. Any miss that would occur if the block size were one word is designated a true sharing miss.

| Time | P1          | P2          |
|------|-------------|-------------|
| 1    | Write $x_1$ |             |
| 2    |             | Read $x_2$  |
| 3    | Write $x_1$ |             |
| 4    |             | Write $x_2$ |
| 5    | Read $x_2$  |             |