

**Bengal Engineering and Science University, Shibpur**  
**B. E. Part-III (CST) 5<sup>th</sup> Semester Examinations, 2012**

Subject: Computer Architecture  
Time: 3 hours

Paper: CS-502  
Full marks: 70

Answer any four

- 1 a) Define RAW, WAR and WAW data hazards in a pipeline system. 5.5  
b) Let consider execution of the following code segment in a 5-stage pipeline shown in Figure 1a

```
LOAD R1, M(R2)
MULT R3, R1, R4
OR R4, R1, R5
AND R6, R1, R7
```

- i) Clearly indicate the data dependencies and their type.  
ii) If no forwarding is implemented in this pipelined processor, then indicate data hazards and show *Stalls* introduced to eliminate those hazards.  
iii) Indicate hazards and show *Stalls* introduced to eliminate those hazards when forwarding is implemented in the pipelined processor.  
iv) What speed-up is achieved in forwarding in comparison to without forwarding? 12

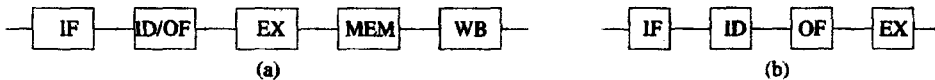


Figure 1

- 2 a) Define branch *delay slot* and *predict taken* branch prediction scheme. 5.5  
b) Refer to the execution of following fragment of code

```
L1: Load R1, M(R2)
    Sub R3, R3, R1
    Bnez R3 L1
    Store R4, M(R3)
```

in a 4-stage pipeline as shown in Figure 1b.

- i) Identify the branch delay slot.  
ii) Assuming *stall-on-branch* and no *delay slot*, what speed-up is achieved on this code if branch outcome is determined at the ID stage, relative to the execution where branch outcome is determined at the EX stage? Show the snapshot of pipeline execution in each case.  
iii) Modify the original program segment by moving an instruction *I<sub>x</sub>* to the delay slot. Evaluate the performance of modified program segment considering “predict taken” scheme. 12

3 a) Describe directory based protocol, with (i) *full directory* and (ii) *limited directory*, to ensure cache-coherence in a multi-processor system. 5.5

b) Explain the write-invalidate cache coherence policy with examples. 6

c) Evaluate performance of invalidation scheme in each of the following access patterns to variable *v*.

i) Repeat a number of times: processor P1 writes a new value into *v* and other 15 processors read the new value.

ii) Repeat a number of times: processor P1 writes 10 times into  $v$ ; this is followed by processor P2 reading the value of  $v$ . 6

4 a) Define the concept of ERCW and CREW shared memory SIMD architectures. Comment on the feasibility of concurrent write (CW). 7.5

b) A file 'file1' contains  $n$  records. Estimate the worst case search time to find a record in file1 with key  $x$  considering (i) ERCW and (ii) CREW SIMD m/c. 10

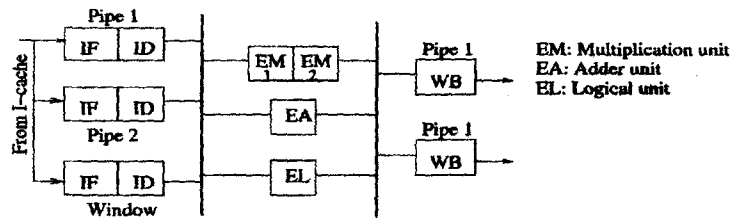
5 a) State the key features of *in-order issue, in-order-completion; in-order issue, out-of-order-completion; and out-of-order issue, out-of-order-completion* superscalar schemes. 5.5

b) Figure 2 describes a superscalar architecture. Consider the execution of following program segment in the superscalar processor-

- I1:  $R1 \leftarrow R2 * R3$
- I2:  $R4 \leftarrow \neg R4$
- I3:  $R5 \leftarrow R6 + R7$
- I4:  $R8 \leftarrow R10 + R5$
- I5:  $R11 \leftarrow R4 . R12$
- I6:  $R13 \leftarrow R9 \text{ xor } R14$

Find the time required to complete the execution and show detail status of the pipeline stages at each instant of time in

- (i) *in-order issue, in-order-completion* and
- (ii) *out-of-order issue, out-of-order-completion* (assume window). 12



6 a) Show how loop interchange technique can reduce the cache miss rate for the following nested loop

```

for j= 1 to 100 with increment 1
  for i= 1 to 3 with increment 1
    A[i][j] = B[j][1] * B[j+1][1];
  
```

Mention the assumptions taken. 5.5

b) Define the schemes *early restart* and *critical word first* employed in cache system design. Evaluate the performance of these two in reducing the cache miss penalty. 6

c) Describe VLIW execution for degree 3. Why does a VLIW machine need a good optimizing compiler? 6

7. Write short notes on the following 10+7.5

- a) Dynamic data flow computer.
- b)  $2^3 \times 2^3$  delta network.