(Pages : 4) **909**

Reg. No. : .....

Name : .....

First Semester M.Tech. Degree Examination, March 2014
(2013 Scheme)
Computer Science And Engineering
RCC 1005 : ADVANCED COMPUTER ARCHITECTURE

Time : 3 Hours Max. Marks : 60

***Instruction** : Answer **two** questions from **each** Module.*

MODULE – 1

1. a) Suppose we have made the following measurements :

   Frequency of FP operations (other than FPSQR) = 25%

   Average CPI of FP operations = 4.0

   Average CPI of other instructions = 1.33

   Frequency of FPSQR = 2%

   CPI of FPSQR = 20

   Assume that the two design alternatives are to decrease the CPI of FPSQR to 2 or to decrease the average CPI of all FP operations to 2.5. Compare these two design alternatives using the CPU performance equation. 5

   b) Suppose we make an enhancement to a computer that improves the speed of accessing stack by a factor of 10. After making this improvement, we notice that the amount of time the computer spends accessing the stack is 50% of the total execution time.

   i) What is the overall speedup obtained ?

   ii) What percentage of the original execution time has been affected by the speedup ? 5

P.T.O.

2. Given the code sequence as follows :

```
L1 :   LD      F2, 0(R1)
       MULTD   F6, F8, F10
       ADDD    F6, F2, F8
       ADDD    F8, F4, F5
       SD      F6, 0(R1)
       DADDUI  R1, R1, #-8
       BNEZ    R1, L1
```

i) Determine all true, anti and output dependencies in the code. Eliminate all name dependencies.

ii) Apply static scheduling and find the number of clock cycles required to complete one iteration of the above loop.

iii) Apply all possible static scheduling and loop unrolling method so that all stall cycles can be eliminated. **10**

3. Use the following code fragment :

```
Loop :  LD R2, 0(R1)          :   R2 = array element
        DADDUI R2, R2, #1     :   increment R2
        SD R2,0(R1)           ;   Store result
        DADDUI R1,R1, #8      ;   increment pointer
        BNE R2, R3, Loop      :   branch if not last element
```

a) Show the pipeline timing chart of this instruction sequence for the 5-stage pipeline without any forwarding or bypassing hardware but assuming that a register read and a write in the same clock "forwards" through the register file. Assume that the branch is handled by flushing the pipeline (2 stall cycles).

b) Show the pipeline timing chart of this instruction sequence for the 5-stage pipeline with full forwarding and bypassing hardware. Assume that the branch is handled by predicting it as not taken.

c) Show the pipeline timing chart of this instruction sequence for the 5-stage pipeline with full forwarding and bypassing hardware. Assume that the branch is handled by predicting it as taken. **10**

MODULE – 2

4. Show how the following code would look on MIPS.
   i) Unscheduled,
   ii) Scheduled,
   iii) loop-unrolled but unscheduled and
   iv) loop-unrolled and scheduled,

   including any stalls or idle clock cycles. Also compute the clock cycles required per element processing. Schedule for delays from floating-point operations, but remember that we are ignoring delayed branches.

```
Loop :  LD      F0, 0(R1)     :   F0 = array element
        ADD.D   F4,F0, F2     :   add scalar in F2
        S.D     F4,0(R1)      ;   Store result
        DADDUI  R1,R1,#-8     ;   decrement pointer 8 bytes (per DW)
        BNE     R1, R2, Loop; ;   branch R1! = R2
```

10

5. Given the following code sequence

```
Loop :  LD      :   F2, 0(R1)
        ADD.D       F4,F2, F3
        SUBD        F6, F4, F5
        S.D         F6, 0(R1)
        DADDUI      R1, R1, #-8
        BNE         R1, R2, Loop
```

   i) Show when the S.D F6,0(R1) of the second iteration of the above loop completes its execution (Write result to CDB) in a dual issue dynamically scheduled superscalar architecture.

   ii) Show when the S.D F6,0(R1) of the second iteration of the above loop completes its execution (Commit the result) in a dual issue dynamically scheduled superscalar architecture. 10

6. Explain the use of GPUs in detail. Compare it with vector architecture. 10

MODULE – 3

7. a) Assume that a directory based protocol for cache coherence and a block is now in the exclusive state at processor P1 and if the directory receives a request for the same cache block from processor p1. Is the above scenario possible ? If yes suggest a solution for the above problem. **6**

   b) Describe the cache coherence problems in data sharing and in process migration. **4**

8. a) Draw a 16-input Omega network using 2×2 witches as building blocks. **4**

   b) Show the switch settings for routing a message from node 1011 to node 0101 and from node 0111 to node 1001 simultaneously. Does blocking exist in this case ? **6**

9. a) Draw and explain the state transition diagram for write invalidate snoopy protocol developed for write-through and write-back caches. **8**

   b) Snoopy bus protocol require a share bus for its implementation. Justify. **2**