

SECTION – A

There are **FOUR** questions in this section. Answer **Question No. 1 (Mandatory)** and any **TWO** other questions.

1. (a) Assume pipelined execution for the following sequence of instructions.

ins 1: sub \$2, \$1, \$3

ins 2: and \$2, \$2, \$5

ins 3: or \$6, \$2, \$7

Assume that **no data forwarding mechanism** is available to you. Construct the execution pipeline for the given sequence along with the stalls required to produce correct output. Evaluate the total number of clock cycles required to execute all three instructions. (12)

(b) Adapt **data forwarding mechanism** for the instruction set in Question 1(a) to enhance parallelism. Reconstruct the execution pipeline for the given sequence showing all pipelined registers and the forwarding lines. Revise the evaluation of the total number of clock cycles required to execute all three instructions. Explain double-data hazard with the help of your constructed pipeline. (16)

(c) Formulate the data forwarding condition between ins 1 and ins 2 in your constructed pipeline for 1 (b). (7)

2. (a) Assume that there are two small caches, each consisting of eight (8) blocks. One cache is direct-mapped, and the other is a two-way set-associative cache. For each cache organization.

(i) Construct the following table for each cache organization for the following sequence of memory block addresses: 32, 36, 35, 32, 48, 32, 51 and 35. Note that you are allowed to subdivide the column “Cache content after access” as needed. Also, when it is necessary to replace an existing cache entry on a cache miss, follow the least recently used (LRU) replacement policy. (20)

Address of Memory Block Accessed	Cache Index	Hit or Miss	Cache content after access
----------------------------------	-------------	-------------	----------------------------

Table 1: Table for Question 2(a)

(ii) Compute the hit ratio from the constructed tables in (i) for each cache organization.

(b) How many total bits are required for an **eight-way set associative cache** of 4K blocks, a 4-word block size (i.e., 16 bytes), assuming a 32-bit address? [Show set number and tag bit calculations.] (15)

Contd P/2

CSE 309

3. (a) Suppose we have a processor with a base CPI of 2.0, assuming all references hit in the primary cache, and a clock cycle length of 0.5 ns. Assume a main memory access time of 150 ns, including all the miss handling. Suppose the miss rate per instruction at the primary cache is 5%.

(i) Calculate the total CPI for one-level of caching. How much faster the ideal CPU is? (25)

(ii) Now assume that we add a secondary cache that has a 15 ns access time for either a hit or a miss and is large enough to reduce the miss rate to 0.5%. The miss rate for primary cache remains 5%. Calculate the total CPI for two-level of caching. [Note that this secondary cache is accessed when there is a miss in the primary cache]

(iii) How much faster will the processor be with a two-level of caching compared to the processor with just one-level of caching?

(b) What is difference between a virtual address and a physical address of instructions of a program? What is the purpose of the virtual addressing of instructions of a program? (5)

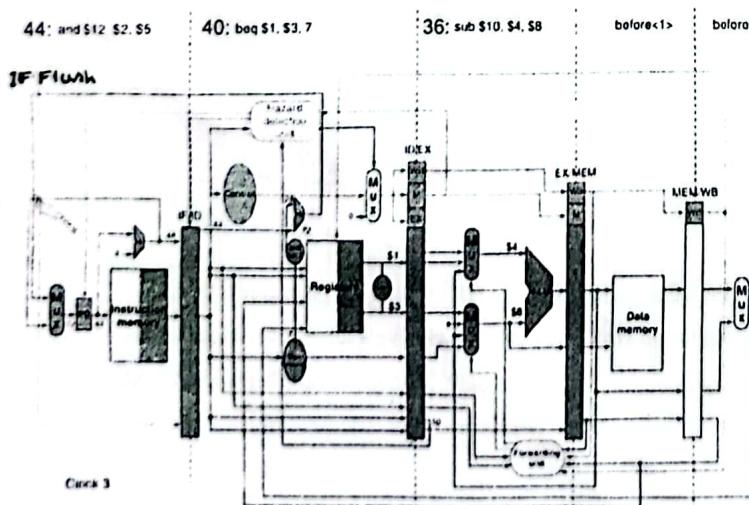
(c) Consider a mapping of a 64-bit virtual address and to a 30-bit physical address. How many entries are there in the page table if the page size is 8 KB. (5)

4. (a) Consider the following branch type instruction format given in Figure for Question 4(a) assuming 32 bit addressing. Construct the formula to calculate branch target address. Draw the single cycle datapath for branch-on-equal instruction assuming 32 bit addressing. [Include only the relevant components and control lines in your design. For example, you can exclude data memory from your design.] (16)

Branch	4	rs	rt	address
	31:26	25:21	20:16	15:0

Figure for Question 4(a): Branch Instruction Format in MIPS

(b) Four hardware components have been added in the ID stage of the pipelined datapath design as shown in Figure for Question 4(b) to reduce branch stalls. Write down the names of these four newly added components. Explain how the branch decision and the target address becomes available by the end of the ID stage of the *beq* instruction. (12)



Contd P/3

Figure for Question 4(b): Branch Stall Reduction Mechanism

CSE 305

Contd ... Q. No. 4

(c) Briefly explain the following concepts:

(7)

i. Shared Memory Multiprocessor (SMP)

ii. SISD vs MIMD

SECTION - B

There are **FOUR** questions in this section. **Question No. 5 is mandatory.** Answer any **TWO** questions from Questions No. 6, 7, and 8.

All relevant tables and figures are provided at the end of the question

5. [MANDATORY QUESTION]

(a) Rachel has a basic MIPS instruction set without support for multiple languages. In an attempt to add such support, she introduces four new instructions: lh (load halfword), lhu (load halfword unsigned), sh (store halfword), and shu (store halfword unsigned). However, Ross argued that she doesn't need all these instructions to achieve multi-language support. Do you agree with Ross? Justify your answer.

(7)

(b) Howard wants to demonstrate his engineering skills, but Sheldon sets a challenging constraint for him. Howard is tasked with designing a 4-bit Arithmetic Logic Unit (ALU), and the constraint is that he cannot use any parallel adder. Instead, he is only allowed to use parallel subtractors and logic gates for the ALU design. The ALU should have three selection variables, denoted as S₀, S₁, and S₂. These selection variables control the ALU to perform various operations on two 4-bit inputs, Labelled as A and B. Your task is to help Howard design the 4-bit Arithmetic Logic Unit (ALU) so that it can do the following operations:

(14)

S ₂	S ₁	S ₀	Operation
0	0	0	A - B
0	0	1	A + B
0	1	0	-A -B - 2
0	1	1	B - A
1	0	0	A XOR B
1	1	0	A AND B

(c) Assemble the two object files of Figure for Question 5(c)-1 and Figure for Question 5(c) -

(14)

2. Find the updated addresses of the first few instructions of the completed executable file. Note that all the necessary addresses and symbols must be updated in the link process.

[Use Figure for Question 5(c)-3 for reference]

6. (a) Joey claims that the number of cycles is identical to the number of instructions. Do you

(5)

agree with Joey's statement? Justify your answer.

(b) What support does the MIPS instruction-set architecture provide for synchronization?

(7)

Explain with an example.

CSE 305

Contd ... Q. No. 6

- (c) Write down the equivalent MIPS code for the following C code, considering registers \$t1 and \$t2 for variables x and y, respectively: (8)

```
if (x > y + 131075)
    x = 8;
```

- (d) Consider two different implementations (P1 and P2) of the same instruction-set architecture.

The instructions can be divided into four classes according to their CPI (class A, B, C, and D)

P1 with a clock rate of 2.7 GHz and CPIs of 2, 1, 3 and 2, and P2 with a clock rate of 3.2 GHz

and CPIs of 2, 3, 2, and 3.

(5+5+5)
=15

Leonard develops a program with a dynamic instruction count of 1.2 E6 instructions, which is divided into the following classes: 20% class A, 25% class B, 40% class C, and 15% class D.

(i) Find the number of clock cycles required in both cases.

(ii) Which implementation is faster and how much faster compared to the other one?

(iii) If Sheldon has improved 40 percent of the code for the program, determine the maximum possible improvement factor Sheldon can obtain for that program.

7. (a) Define response time and throughput. Are these two terms synonymous? (5)

(b) Chandler has used 40 variables in his program, whereas the MIPS architecture has only 32 registers. What does the compiler do in such a scenario? (7)

(c) Translate function f (defined below) into MIPS assembly language. If you need to use registers \$t0 through \$t7, use the lower-numbered registers first. Assume the function declaration for func is "int func(int a, int b);". The code for function f is as follows: (8)

```
int f (int a, int b, int c, int d) {
    return func(func(a,b), c+d);
}
```

(d) Write about the main difference between static linking and dynamic linking. With the necessary diagrams, explain how dynamically linked libraries (DLLs) are called. (15)

8. (a) Write down equivalent efficient sequence of MIPS instructions for the following C code fragment. Assume appropriate registers for each of the variables. (7)

```
float a, b, c;
double w, x, y, z;
if (a+b>c)
    x=(y*z) / w;
```

(b) Consider a 32-bit system where 19 bits are reserved for fraction, one bit for sign, and the rest for exponent. Answer the following questions: (2+3+3)=8

(i) What is the precision range of this system?

(ii) What are the smallest and the largest absolute values that can be stored in this system?

(iii) What is the smallest denormalized number that can be stored using this system?

(c) Consider two normalized floating point numbers f1 and f2 that are represented as IEEE single precision floats. This defines two 32-bit strings. (10+10)=20

(i) Suppose f1 is less than f2, that is, f1 < f2. If the same two 32-bit strings were treated as unsigned integers (call them i1 and i2, respectively), can we conclude that i1 < i2?

(ii) Similar to (i), if the two 32-bit strings were treated as signed integers j1 and j2, can we conclude that j1 < j2?

Object file header				
	Name	Procedure A		
	Text size	150 _{hex}		
	Data size	40 _{hex}		
Text segment	Address	Instruction		
	0	lw \$a0, 0(\$gp)		
	4	jal 0		
		
Data segment	0	(X)		
		
Relocation information	Address	Instruction type	Dependency	
	0	lw	X	
	4	jal	B	
Symbol table	Label	Address		
	X	-		
	B	-		

Figure for Question. 5(c)-1: Object File Header for Procedure A

Object file header				
	Name	Procedure B		
	Text size	350 _{hex}		
	Data size	50 _{hex}		
Text segment	Address	Instruction		
	0	sw \$a1, 0(\$gp)		
	4	jal 0		
		
Data segment	0	(Y)		
		
Relocation information	Address	Instruction type	Dependency	
	0	sw	Y	
	4	jal	A	
Symbol table	Label	Address		
	Y	-		
	A	-		

Figure for Question. 5(c)-2: Object File Header for Procedure B

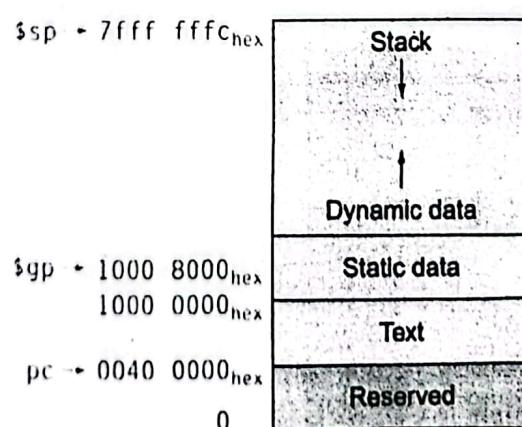


Figure for Question. 5(c)-3: The MIPS memory allocation for program and data.

SECTION - A

There are FOUR questions in this section. Answer any THREE questions.

- I. (a) Complete the provided incomplete diagram of the MIPS single cycle datapath for the branch if equal (beq) and jump (j) instructions. (10)
 (b) Consider the following MIPS code: (10)

```

lw $t1, 0($t0)
lw $t2, 4($t0)
add $t3, $t1, $t2
sw $t3, 12($t0)
lw $t4, 8($t0)
add $t5, $t1, $t4
sw $t5, 16($t0)
  
```

How many cycles are required to execute the above code? Can you reduce the number of cycles using code scheduling? If yes, what will be resulting number of cycles?

- (c) What do you mean by data hazard for branches? Give MIPS code examples with explanations for each of the following cases of data hazard for branches: (10)

- (i) Needs no stall to resolve
- (ii) Needs one cycle stall to resolve
- (iii) Needs two cycle stalls to resolve

- (d) Consider the following code: (5)

```

outer:
// some code
inner:
// some code
beq $t0, $zero, inner
// some code
beq $t1, $zero, outer
  
```

What is the problem with the above code if we use 1 bit branch predictor? How can you solve that?

CSE 305

2. (a) What do you mean by forwarding for data hazard? Design a complete forwarding unit in the provided diagram with necessary equations for the MIPS pipeline considering double data hazard. (15)

(b) Draw the block diagram of a dynamically scheduled multiple-issue super-scalar CPU. (10)

(c) Consider the following program: (10)

```
Loop:    lw $t0, 0($s1)
          addu $t0, $t0, $s2
          sw $t0, 0($s1)
          addi $s1, $s1, -4
          bne $s1, $zero, Loop
```

Schedule the above program for a dual-issue MIPS with the following configuration:

Two-issue packets, one ALU/branch instruction, one load/store instruction, ALU/branch then load/store.

What will be the Instructions per Cycle (IPC) for this scheduling? Which technique can increase the IPC even more?

3. (a) Suppose we have a processor with a base CPI of 1.0, assuming all references hit in the primary cache and a clock rate of 4 GHz. Assume a main memory access time of 200 ns, including all the miss handling. Suppose the miss rate per instruction at the primary cache is 4%. How much faster will the processor be if we add a second level cache that has a 10 ns access time for either a hit or a miss and is large enough to reduce the miss rate to main memory to 1%? How much faster will the processor be if we add a third level cache that has a 40 ns access time for either a hit or a miss and is large enough to reduce the miss rate to main memory to 0.5%? (10)
- (b) Explain with necessary diagrams how the page table works with respect to the virtual address, physical address, main memory, and disk. (10)
- (c) You are given the following hit/miss status of TLB, Page Table and Cache. (10)

Serial No.	TLB	Page Table	Cache
1	Hit	Hit	Hit
2	Hit	Miss	Hit
3	Miss	Miss	Hit
4	Miss	Miss	Miss
5	Miss	Hit	Hit

Table 3(c)

Explain which of the scenarios in Table 3(c) is possible or impossible and under what circumstance with proper reasoning.

- (d) Explain how snooping protocol solves the cache coherence problem with an example. (5)

4. (a) Assume there are three small caches, each consisting of four one-word blocks and uses LRU replacement policy. One cache is fully associative, a second is two-way set-associative, and the third is direct-mapped. Find the number of misses for each cache organization given the following sequence of block addresses: (15)
1, 3, 2, 4, 5, 1, 3, 1, 3, 5, 3, 2
(b) Show with block diagrams how contemporary PCs with Intel and AMD CPUs connect a separate GPU. How do the CPU and GPU access each other's memory? (10)
(c) What do you mean by kernel, grid, and thread block of the CUDA paradigm? Briefly describe how they map to the streaming processor (SP) cores and streaming multiprocessors (SM) of the unified GPU architecture of NVIDIA. (10)

SECTION – B

There are **FOUR** questions in this section. Answer any **THREE** questions.

5. (a) The following four design principles have been guiding the instruction-set designers to find a balance between the number of instructions needed to execute a program, the number of clock cycles needed by an instruction, and the speed of the clock. (3×5=15)
(i) Simplicity favors regularity
(ii) Smaller is faster
(iii) Good design demands good compromise
(vi) Make the common case faster

You are given several design features of MIPS instruction-set architecture in the following. Now, please identify which of the above principle(s) is behind each of these design decisions.

- (i) Keeping conditional branch's offset value in a range of 16 bits.
- (ii) Keeping 32 registers rather than many more.
- (iii) Consistent instruction format, i.e., always requiring three registers for R-type Instructions.
- (iv) Multiple instruction formats to allow flexibility.
- (v) Reduced instruction set, Complex instructions (i.e., pseudo instructions like bgt, bge, mov etc.) are translated into one or more basic instructions.

- (b) Assume that 2% of the runtime of a program is not parallelizable. This program runs on 24 cores of an Intel Xeon Gold processor. Under the assumption that the program runs at the same speed on all of those cores and there are no additional overheads, (4+6=10)

CSE 305

Contd.... for Q. No. 5(b)

- (i) What is the parallel speedup for 24 cores?
(ii) Is it possible to make the program 60 times faster? If yes, how can that be achieved? If not, mention the reasons.
(c) Discuss how can subword parallelism be used to accelerate common operations, such as matrix multiplication or image processing? (10)
6. (a) Write the MIPS assembly code for the C code shown in the following figure (Figure: 6.a). (15)
- ```
int fib(int n) {
 if (n==0) return 0;
 else if (n == 1) return 1;
 else return fib(n-1) + fib(n-2);
}
```
- Figure: 6.a
- (b) Suppose the program counter (PC) is set to 0x2000 0000. Explain your answer for the following questions. (5+5=10)
- (i) Is it possible to use the jump (j) MIPS assembly instruction to set the PC to the address 0x4000 0000?  
(ii) Is it possible to use single branch (beq/bneq) instruction to set the PC to this same address?  
(c) Draw the block diagram of 32-bit combined multiplication and division hardware which uses 32-bit ALU, and briefly mention functions of different components of this hardware. (10)
7. (a) Assume that you have four different threads as shown in figure (Figure: 7.a) below. Show how the four threads could be combined to execute on the processor more efficiency using the following three multithreading options. You only need to show the first 12 timeslots for each of the following options. (5+5+5=15)

- (i) A superscalar with coarse-grained multithreading  
(ii) A superscalar with fine-grained multithreading  
(iii) A superscalar with simultaneous multithreading

**CSE 305**

Contd.... for Q. No. 7(a)

| Time | Thread A | Thread B | Thread C | Thread D |
|------|----------|----------|----------|----------|
| T1   | A A      |          |          |          |
| T2   | A        |          |          |          |
| T3   | A A A A  |          |          |          |
| T4   |          |          |          |          |
| T5   |          |          |          | C C C    |
| T6   | A        |          |          | D D D D  |
| T7   | A A      | B B B B  | C C C C  | D D      |
| T8   | A A A    | B B      |          | D D D    |

Figure 7.a

- (b) Suppose you are using Floating Point Operations Per Second (FLOPS) as a performance metric for comparing processors. Your research partner believes that FLOPS might be a misleading metric for measuring the general performance of a processor, but it could still be a useful performance measure for a specific part of the processor or a specific type of processors. Do you agree or disagree with your research partner's statements? Provide a detailed explanation to support your position. (16)
- (c) With illustrative example(s), explain how vector processor differs from array processor. (10)
8. (a) Consider two different implementations (P1 and P2) of the same instruction set architecture. There are three classes of instructions A, B, and C. Suppose, a program has 200,000 instructions divided into three classes as 25% class A, 60% class B, and 15% class C. The clock rate and CPI of each implementation are given in the following table (Table: 8.a). (3+3+5=11)
- | Implementation | Clock Rate | CPI of Class A | CPI of Class B | CPI of Class C |
|----------------|------------|----------------|----------------|----------------|
| P1             | 2.7 GHz    | 3              | 3              | 2              |
| P2             | 3.5 GHz    | 2              | 4              | 1              |
- Table: 8.a
- (i) Find the total clock cycle required in both implementations.
  - (ii) Calculate the execution time required for both implementations and determine which implementation is faster.
  - (iii) How much must we improve the CPI of Class C instructions if we want the program to run two times faster in the better implementation determined in the previous question?
- (b) Consider representing the set of consecutive positive integers {1, 2, 3, 4, ..., n} using IEEE single precision floating point (8-bit exponent and 23-bit mantissa). Find the largest n such that every number in the above set can be represented. (14)
- (c) What are the different addressing modes of MIPS? Illustrate the calculation process for each addressing mode. (10)

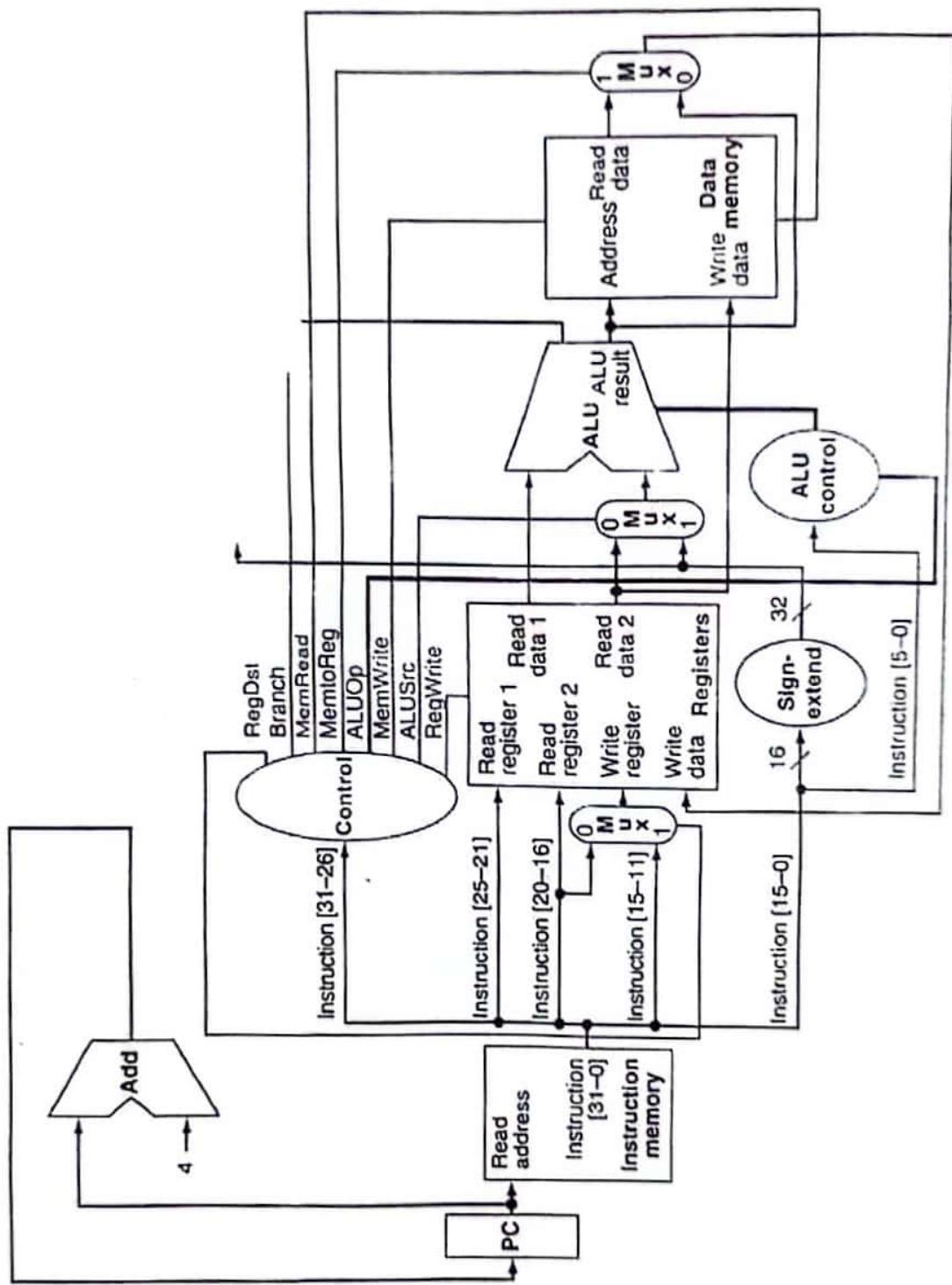


Figure for Question 1 (a)

Staple it with your answer script

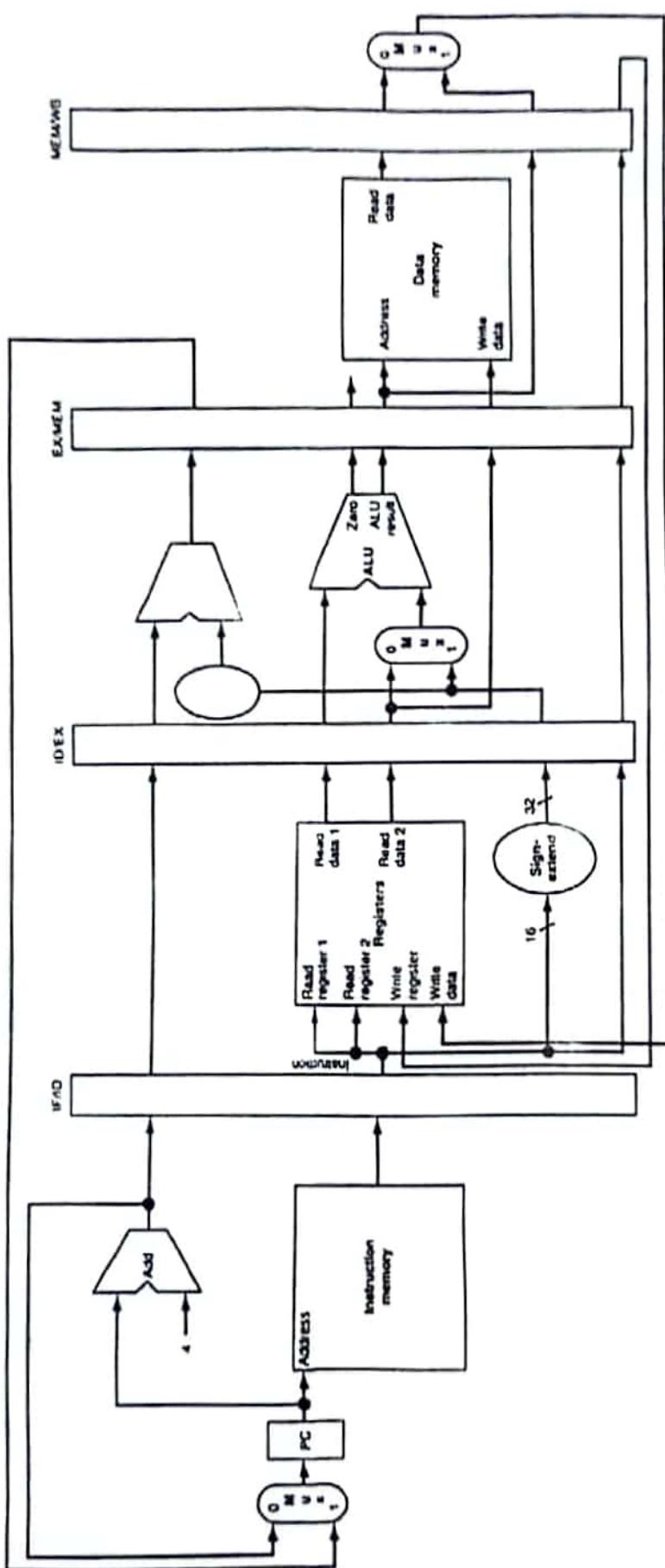


Figure for Question 2 (a)

Staple it with your answer script

**SECTION - A**

There are **NINE** questions in this section. Answer any **SEVEN** questions.

1. You have surely learnt about the eight big ideas of computer architecture. They are again mentioned below to brush up your memory: (15)
- Design for Moore's Law
  - Use abstraction to simplify design
  - Make the common case fast
  - Performance via parallelism
  - Performance via pipelining
  - Performance via prediction (a)
  - Hierarchy of memories (b)
  - Dependability via redundancy
- Now identify the statements below best relate to which of the above ideas. Each statement should relate to a single idea. Elaborate the justification behind your answer.
- (a) Code reordering is done to avoid the use of load result in the next instruction.  
(b) Different caches for data and memory can effectively increase bandwidth.  
(c) As CPUs have hit their physical performance barrier due to power limit issues, GPUs have started gaining increased popularity among the programming community.
2. What is your most favorite topic learnt from processor, memory hierarchy, and multiprocessors? Describe briefly. (15)
3. A full datapath with a control unit is given in Figure 1. Highlight the datapath elements that will be active for the `ori $t1, 100($t2)` instruction. Also, mention which control bits will be set. [You may highlight the given figure and attach it to your answer script.] (15)
4. Suppose you are maintaining a 5-stage pipeline in a MIPS datapath. You are given the following code snippet to execute. Identify all the data hazards that can occur using a multi-cycle pipeline diagram and mark where there is a need for forwarding. Also

write down the forwarding conditions to mitigate the hazards. Can all of the hazards in this snippet be addressed using forwarding? Defend your position. Also write down the forwarding conditions to mitigate the hazards. Can all of the hazards in this snippet be addressed using forwarding? Defend your position.

(15)

```

lw $t0, 0($t1)
add $t0, $t0, $t1
add $t0, $t2, $t0
add $t0, $t0, $t3
sub $t1, $t2, $t3
and $t4, $t5, $t6
sw $t0, 0($t6)

```

6. You are given the following code snippet and have been tasked to run multiple issues in parallel for faster execution. Refactor the code so that IPC is maximized. Also compute the effective IPC. [Hint: you do not have to limit the code to two issues.]

(15)

```

Loop: lw $t0, 0($$0)
 lw $t1, 0($$1)
 addu $t0, $t0, $t1
 sw $t0, 0($$1)
 addi $$0, $$0, -4
 addi $$1, $$1, -4
 bne $$1, $zero, Loop

```

6. Answer the questions below for the following memory address subdivision and a 2-way set associate LRU cache:

(15)

|           |           |          |         |
|-----------|-----------|----------|---------|
| 31 ... 20 | 19 ... 12 | 11 ... 2 | 1 ... 0 |
| 12-bits   | 8-bits    | 10-bits  | 2-bits  |

- (a) How many distinct blocks of data can there be in the cache?
- (b) In which cache index will the decimal address 24092022 map to?
- (c) If a cache index is valid, what is the probability that one of its elements will have to be replaced in the next request? Assume each memory address has equal likelihood of being requested and each element in a cache index has equal likelihood of being filled.

CSE 305

7. Suppose in a memory hierarchy there are four types of memories: (15)

Cache → RAM → SSD → HDD . Their local miss rates are 5%, 15%, 25%, and 0% respectively. Their access times are 1, 10, 100, and 1,000 cycles respectively. Now answer the following questions:

(a) What is the probability that a data will be fetched from the SSD?

(b) What is the expected time of a fetch?

(c) If the cache is separated into an I-cache and a D-cache, what will the miss rate of the I-cache be for a program that accesses data elements in 10% of its instructions if the overall miss rate remains constant (i.e., 5%) and the D-cache miss rate is 10%?

8. Suppose you have implemented an approximate LRU page replacement scheme using a reference bit for each page table entry. Your virtual memory has eight pages and page table has four entries. After every four time-steps, the reference bits are cleared. Assuming a fully set associative page table, simulate your page replacement scheme for the virtual memory reference below: (15)

0, 0, 2, 7, 0, 6, 1, 2, 6, 7, 1, 1, 7, 1, 4, 1

You have to print the page table after each page table reference. You also have to mention whether or not it is a hit or miss.

You want to perform two sums in a program: one is a sum of 8 scalar variables and another is a matrix sum of a pair of two-dimensional arrays of size 8 by 8. Assume you are doing these sums with 8 processors. (15)

(a) Assume only the matrix sum is parallelizable, what speed-up will you get with 8 versus 16 processors?

(b) Again assuming only the matrix sum is parallelizable, calculate the speed-up when the matrices grow to 16 by 16.

(c) Devise a recursive algorithm to speed-up the scalar sum and calculate the new speed-up.

SECTION – B

There are **FOUR** questions in this section. Answer any **THREE** questions.

10. (a) Briefly explain the eight great ideas in Computer Architecture. (12)

(b) Suppose you want to design a 4-bit ALU where the inputs to the ALU are A ( $A_3A_2A_1A_0$ ) and B ( $B_3B_2B_1B_0$ ), and the selection variables are  $S_3$ ,  $S_2$ ,  $S_1$  and  $S_0$ . Also, consider  $X_i$ ,  $Y_i$  and  $Z_i$  as the inputs to the  $i$ -th parallel adder of the ALU. The

= 4 =

CSE 305

Contd.... for Q. No. 10

ALU must satisfy the following functional design specification (given in Table 10b) that has to be realized through the parallel adders. Your task is to derive the input equations ( $X_i$ ,  $Y_i$  and  $Z_i$ ) for the parallel adders of the ALU. You must design using the least possible number of gates.

(18)

| $S_3$ | $S_2$ | $S_1$ | $S_0$ | Required Functions     |
|-------|-------|-------|-------|------------------------|
| 0     | 0     | 0     | 0     | $F = A$                |
| 0     | 0     | 0     | 1     | $F = A - 1$            |
| 0     | 0     | 1     | 0     | $F = A + B + 1$        |
| 0     | 0     | 1     | 1     | $F = A + B$            |
| 0     | 1     | 0     | 0     | $F = A - B$            |
| 0     | 1     | 0     | 1     | $F = A - B - 1$        |
| 0     | 1     | 1     | 0     | $F = A + 1$            |
| 0     | 1     | 1     | 1     | $F = A$                |
| 1     | 0     | 0     | X     | $F = A'$               |
| 1     | 0     | 1     | X     | $F = AB'$              |
| 1     | 1     | 0     | X     | $F = A \odot B$ [XNOR] |
| 1     | 1     | 1     | X     | $F = A \vee B$ [OR]    |

Table 10(b): Required functional design specification

- (c) Briefly describe how to perform a signed multiplication operation.

Contd ..... P/S

11. (a) Suppose a hypothetical performance benchmark SPECBM101 runs on a computing system and the following data is obtained. (10)

| Description | Execution Time<br>(seconds) | Reference Time<br>(seconds) |
|-------------|-----------------------------|-----------------------------|
| Program 1   | 567.63                      | 9770                        |
| Program 2   | 527                         | 10490                       |
| Program 3   | 586                         | 12100                       |
| Program 4   | 109                         | 20720                       |
| Program 5   | 470                         | 7020                        |
| Program 6   | 275                         | 6900                        |

If the geometric mean of the benchmark is 28.74 seconds, what is the execution time of Program 1?

- (b) Consider the following figure (Figure 11b) of a processor unit employing a scratchpad memory. How many micro operations does it require to complete an addition operation with two operands, and store the result? Which changes and constraints are required to reduce the number of micro operations to only one? (2+6=8)

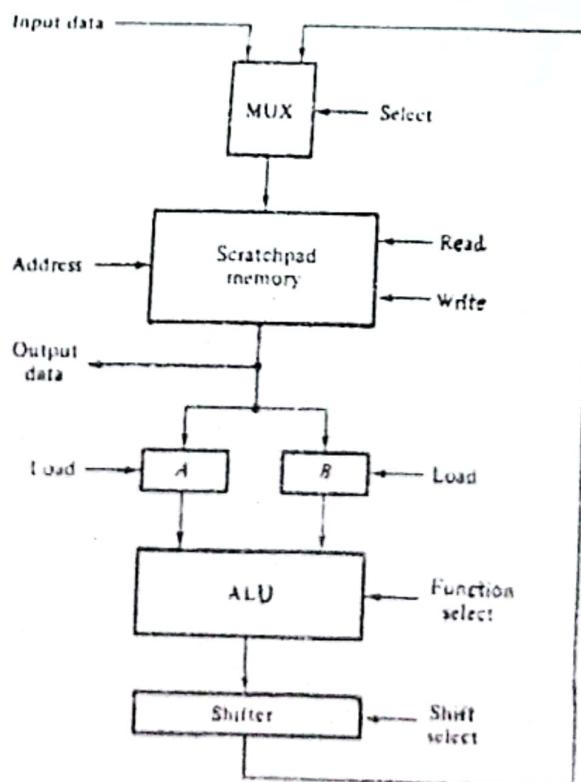


Figure 11(b). Processor unit employing a scratchpad memory

CSE 305

Contd.... for Q. No. 11

(c) Write the equivalent efficiency sequence of MIPS instructions for the following C code fragment.

(12)

```
int testFunc (int a, int b, int c)
{
 int i=0, result=0;
 do
 {
 if(i>b)
 result += c;
 i++;
 }while (i<a)

 return result;
}
```

Assume, \$s1 and \$s2 will carry the values of variable *i* and variable *result*, respectively.

(d) Provide an illustrative example for showing how MIPS architecture follows the design principle: "Good design demands good compromises."

(5)

12. (a) The data for the two programs A and B are given below.

(8)

| Instruction Type | Clock Cycle per Instruction (CPI) | Instruction Count in Program A | Instruction Count in Program B |
|------------------|-----------------------------------|--------------------------------|--------------------------------|
| Multiplication   | 3                                 | 2                              | 3                              |
| Addition         | 1                                 | 1                              | 2                              |
| Subtraction      | 2                                 | 4                              | 3                              |
| Division         | 4                                 | 2                              | 2                              |

Which program has the higher average clock cycle per instruction?

(b) Consider the following MIPS code:

(10)

L1: bne \$t0, \$t1, L3

L2: j LX

L3: addi \$t0, \$t1, 1

CSE 305

Contd.... for Q. No. 12(b)

Now, if the opcode of *j*, *bne* and *addi* are 2, 5 and 8 respectively, the register \$t0 and \$t1 are indicated by register values 8 and 9 respectively, and the addresses are

$$L1 \rightarrow 1000$$

$$L2 \rightarrow 1004$$

$$L3 \rightarrow 1008$$

$$LX \rightarrow 131072$$

then, write down the instruction format name and the instruction values for each of the above-mentioned instructions. The following example is provided as a hint.

**Example:** Consider the following MIPS instruction:

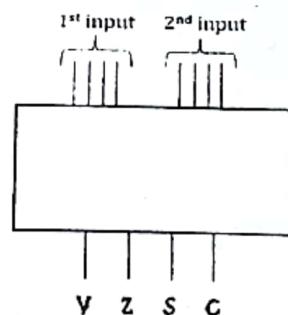
add \$s1, \$s2, \$s3

The instruction format is R-format and instruction values should be:

| Address | 6 bits | 5 bits | 5 bits | 5 bits | 5 bits | 6 bits |
|---------|--------|--------|--------|--------|--------|--------|
| 0400    | 0      | 18     | 19     | 17     | 0      | 0      |

(c) Briefly explain the importance of guard digit, round digit, and sticky bit during rounding an intermediate floating point number. (9)

(d) Consider the following module. (8)



This module takes two 4-bit inputs, and shows only the four status bits (V: Overflow, Z: Zero, S: Sign and C: Carry) considering the subtraction operation of the second input from the first input. If the first 4-bit input is expressed by A and second 4-bit input is expressed by B; draw the diagram of the complete circuit that can determine whether A>B, A=B, or A<B.

CSE 305

13. (a) Consider a 32-bit system where 24 bits are reserved for fraction, one bit for sign, and the rest for exponent. Answer the following questions. (4×4=16)

- (i) Why is bias used during the floating point representation?
- (ii) What should be the value of the bias for this system? Why?
- (iii) What are the smallest and the largest absolute values that can be stored in this system?
- (iv) What is the smallest denormalized number that can be stored using this system?

(b) How is address calculation performed in PC-relative addressing? (7)

(c) Write the equivalent efficient sequence of MIPS instructions for the following C code fragment. Assume appropriate registers for each of the variables. (7)

```
float a, b, c;
double w, x, y, z;
if (a+b>c)
 x=(y*z)/w;
```

(d) What are the main differences between x86 architecture and MIPS architecture? (5)

BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY, DHAKA

L-3/T-1 B. Sc. Engineering Examinations 2019-2020

Sub: CSE 305 (Computer Architecture)

Full Marks: 210 Time: 2 hours and 30 minutes (Sections A+B)

USE SEPARATE SCRIPTS FOR EACH SECTION

The figures in the margin indicate full marks.

**SECTION – A**There are **FOUR** questions in this section. Answer any **THREE**.

1. (a) Suppose we want to compare the performance of *Computer System A* with *Computer System B*. (15)

For this purpose, *Program 1* is run in both the computer systems and corresponding information (logs) are captured. The logs are presented below in Table 1.1 and Table 1.2.

| Operation                        | Required Clock Cycle(s) |
|----------------------------------|-------------------------|
| Multiplication 1 of Program 1    | 4                       |
| Addition 1 of Program 2          | 2                       |
| I/O overhead time required by OS | 5                       |
| Multiplication 2 of Program 1    | 4                       |
| Division 1 of Program 1          | 5                       |
| Multiplication 1 of Program 2    | 4                       |
| I/O overhead time required by OS | 2                       |
| Division 2 of Program 1          | 5                       |
| I/O overhead time required by OS | 4                       |
| Addition 1 of Program 3          | 2                       |
| Addition 1 of Program 1          | 2                       |

**Table 1.1: Logs from Computer System A**

| Operation                        | Required Clock Cycle(s) |
|----------------------------------|-------------------------|
| Multiplication 1 of Program 4    | 5                       |
| Multiplication 1 of Program 1    | 5                       |
| I/O overhead time required by OS | 1                       |
| Multiplication 2 of Program 1    | 5                       |
| Division 1 of Program 1          | 6                       |
| Multiplication 2 of Program 4    | 5                       |
| Addition 1 of Program 4          | 3                       |
| I/O overhead time required by OS | 3                       |
| Addition 2 of Program 4          | 3                       |
| Division 2 of Program 1          | 6                       |
| Addition 1 of Program 1          | 3                       |

**Table 1.2: Logs from Computer System B**

If the clock rates of *Computer System A* and *Computer System B* are 2 GHz and 2.5 GHz respectively, then compare their performances based on the execution time of *Program 1*.

- (b) What is the importance of Moore's Law? "To increase the overall performance of a computer by  $x$  amount, it is required to identify the most important aspect of the computer and improve its performance by  $x$  amount." Do you agree or disagree with the statement? Justify your answer with proper mathematical explanation. (4+8=12)
- (c) What are the advantages of using Dynamically Linked Libraries (DLL) during a C program execution? What is the role of JIT in Java program execution? (4+4=8)
2. (a) Suppose you want to design a 4-bit ALU where the inputs to the ALU are  $A$  ( $A_3A_2A_1A_0$ ) and  $B$  ( $B_3B_2B_1B_0$ ), and the selection variables are  $S_3, S_2, S_1$  and  $S_0$ . Also, consider  $X_i, Y_i$  and  $Z_i$  as the inputs to the  $i^{th}$  parallel adder of the ALU. The ALU must satisfy the following functional design specification (given in Table 2a) that has to be realized through the parallel adders. Your (20)

task is to derive the input equations ( $X_i, Y_i$  and  $Z_i$ ) for the parallel adders of the ALU. You must design using the least possible number of gates.

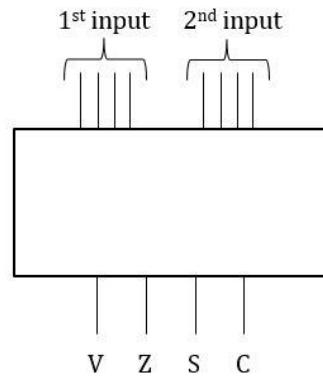
| $S_3$ | $S_2$ | $S_1$ | $S_0$ | Required Functions     |
|-------|-------|-------|-------|------------------------|
| 0     | 0     | 0     | 0     | $F = A'$               |
| 0     | 0     | 0     | 1     | $F = A' + 1$           |
| 0     | 0     | 1     | 0     | $F = A' + B$           |
| 0     | 0     | 1     | 1     | $F = A' + B + 1$       |
| 0     | 1     | 0     | 0     | $F = A' - B - 1$       |
| 0     | 1     | 0     | 1     | $F = A' - B$           |
| 0     | 1     | 1     | 0     | $F = A' - 1$           |
| 0     | 1     | 1     | 1     | $F = A'$               |
| 1     | 0     | 0     | x     | $F = (AB)'$ [NAND]     |
| 1     | 0     | 1     | x     | $F = A \odot B$ [XNOR] |
| 1     | 1     | 0     | x     | $F = A \oplus B$ [XOR] |
| 1     | 1     | 1     | x     | $F = A \vee B$ [OR]    |

**Table 2a: Required Functional design specification**

- (b) Suppose, you want to implement a hardware unit that would produce the value of  $F$  using the following logic (the logic is given in C syntax). (15)

```
if ((A>=B && C!=D) || X==0)
 F=1;
else
 F=0
```

Here, each of the variables  $A, B, C, D$  and  $X$  is of 4-bits where the variable  $F$  is of 1-bit. After searching the hardware stationaries, you bought some basic gates. Also, you bought some status-generator modules (see Figure 2b) each of which takes two 4-bit inputs, and shows only the four status bits (V: Overflow, Z: Zero, S: Sign and C: Carry) considering the subtraction operation of the 2<sup>nd</sup> input from the 1<sup>st</sup> input.



**Figure 2b: Status-generator module**

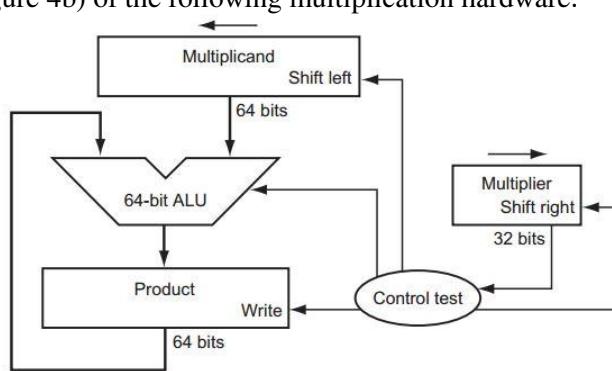
Now, (just) draw the circuit diagram of the full hardware unit using basic gates and status-generator modules.

3. (a) Write down the equivalent MIPS code for each of the following the C code fragments. (6+5+4)

- i. Code Fragment 1:

```
sum = 0;
x = 100;
while (x>=-5)
{
 sum = sum + x;
 x = x-1;
}
```

- ii. Code Fragment 2:
- ```
if (x>y+131075)
    x=1;
```
- iii. Code Fragment 3:
- ```
x = A[131075]+5;
```
- (b) With an appropriate example, briefly describe how pseudoinstructions work. Suppose, there is a critical section in your code where performance matters. Do you think it is always a good idea to write the assembly code by yourself for such type of scenario? Justify your answer. (6+2+4 =12)
- (c) Why does MIPS multiplication operation always strictly demand for three operands? Why has the number of registers in MIPS architecture been restricted to 32? (4+4=8)
4. (a) Suppose, you are working in a system where 16 bits are used to represent a floating-point value. The largest absolute floating-point value that can be represented by the system is  $1.11 \dots 1_{two} \times 2^{63}$  while the smallest absolute floating point value that can be represented by the system is  $1.00 \dots 0_{two} \times 2^{-62}$ . (6+5+4 =15)
- Show the floating-point representation format (bit distribution among sign, exponent and fraction portion) of this system.
  - Show the floating-point representation for the number  $-110.11_{two} \times 2^1$  in this system assuming an appropriate bias.
  - What is the smallest absolute denormalized floating-point value that can be stored in this system?
- (b) See the diagram (Figure 4b) of the following multiplication hardware. (4+4+4 =12)



**Figure 4b: Multiplication Hardware**

Briefly describe the reasons why this hardware is inefficient for a multiplication operation. What changes are required to make this hardware efficient? Draw the diagram of the efficient hardware.

- (c) Consider the following C code fragment. (8)

```
if (a>b)
 c = c- (c*e) /d;
else
 c = c+d;
```

If  $a, b, c, d$  and  $e$  are variables of type *double* and their values are already loaded in registers \$f2, \$f4, \$f6, \$f8 and \$f10 respectively, then write down the equivalent MIPS assembly code for it. Use the least number of MIPS instructions possible.

BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY, DHAKA

**L-3/T-1** B. Sc. Engineering Examinations 2019-2020

Sub: **CSE 305** (Computer Architecture)

Full Marks: 210 Time: 2 hours and 30 minutes

USE SEPARATE SCRIPTS FOR EACH SECTION

The figures in the margin indicate full marks.

---

**SECTION – B**

There are **FOUR** questions in this section. Answer any **THREE**.

5. (a) Assume pipelined execution for the following sequence of instructions. The corresponding C code is  $A = B + C; D = A + D$ . (15)

```

lw $t1, 0($t0) // load B
lw $t2, 4($t0) // load C
add $t3, $t1, $t2 // add B & C
sw $t3, 12($t0) // store A
lw $t4, 8($t0) // load D
add $t4, $t3, $t4 // add A & D
sw $t4, 8($t0) // store D

```

Draw the execution pipeline for the given sequence of instructions and clearly show the stalls and the forwarding paths. Note that data forwarding is available if needed.

- (b) How many stalls and clock cycles are required in question 5(a)? (5)

- (c) Is it possible to rearrange the code in question 5(a) to reduce the required number of stalls? If not, then explain why it is not possible. If yes, then write down the rearranged code block and indicate the number of stalls and the total number of clock cycles required after the rearrangement.

6. (a) Assume there are *three* small caches, each consisting of eight (8) one-word blocks. (20)  
 One cache is **direct-mapped**, a second is **two-way set-associative**, and the third is **four-way set-associative**. For each cache organization, complete the following table for the following sequence of memory block addresses: **0, 12, 16, 18, and 0**. You are allowed to subdivide the third column "Contents of cache blocks after reference" as needed.

Note that when it is necessary to replace an existing cache entry on a cache miss, follow the least recently used (LRU) replacement rule to determine which cache entry to replace and show the cache content after the replacement.

| Address of memory block accessed | Hit or Miss | Contents of cache blocks after reference |
|----------------------------------|-------------|------------------------------------------|
| <b>0</b>                         |             |                                          |
| <b>12</b>                        |             |                                          |
| <b>16</b>                        |             |                                          |
| <b>18</b>                        |             |                                          |
| <b>0</b>                         |             |                                          |

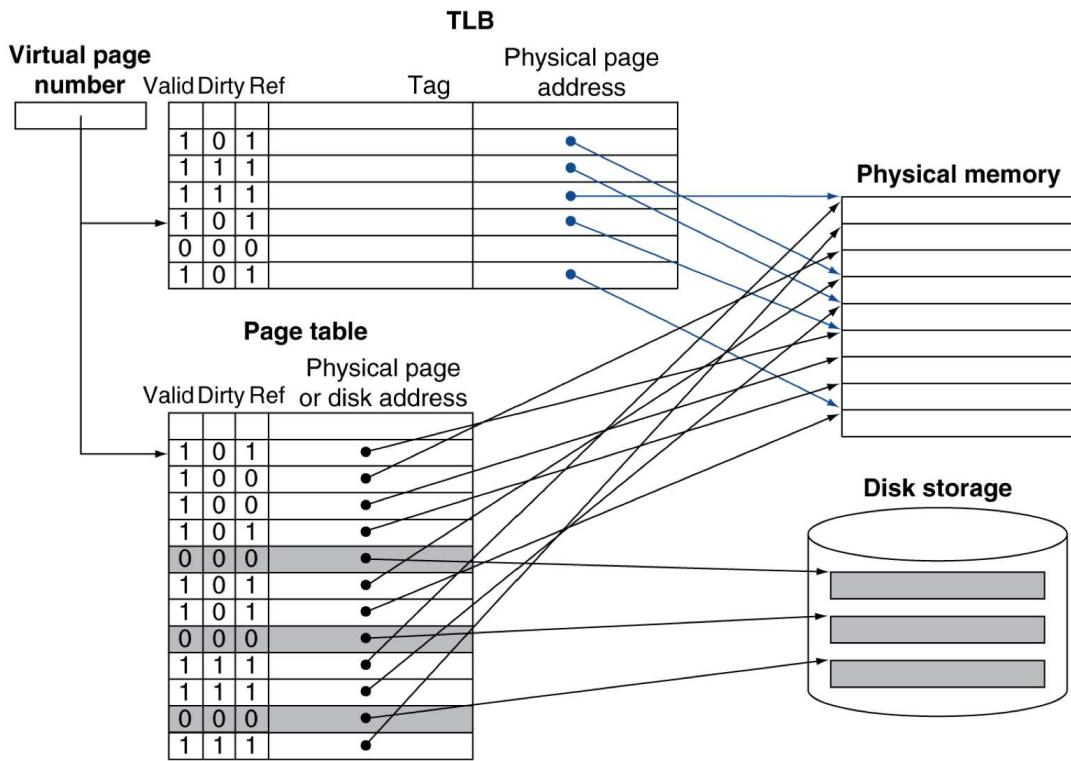


Figure 7b: TLB acts as a cache of the page table

- (b) Assuming a cache of 4K blocks, a 4-word block size, and a 32-bit address, find the **total** number of tag bits for a cache that is **eight-way set associative**. (15)

7. (a) How many *total* bits are required for a **direct-mapped** cache with 64 KB of data and 8-word blocks (i.e., 32 bytes), assuming a 32-bit address? (20)
- (b) The TLB acts as a cache of the page table for the entries that map to physical pages only as shown in Figure 7b. Write down the purpose of the tag bits in TLB. (5)
- (c) Explain how a TLB miss is handled. (10)

8. (a) Figure 8a shows a single cycle datapath implementation of a subset of MIPS (**5x3=15**) instruction set. For each of the following three instructions: **add**, **sw**, **beq**, determine the values (0/1) of the following control signals shown in Figure 8a.

RegDst  
Branch  
MemRead  
MemtoReg  
MemWrite  
ALUSrc  
RegWrite

- (b) Draw the finite state machine for a 2-bit dynamic branch prediction scheme. (8)

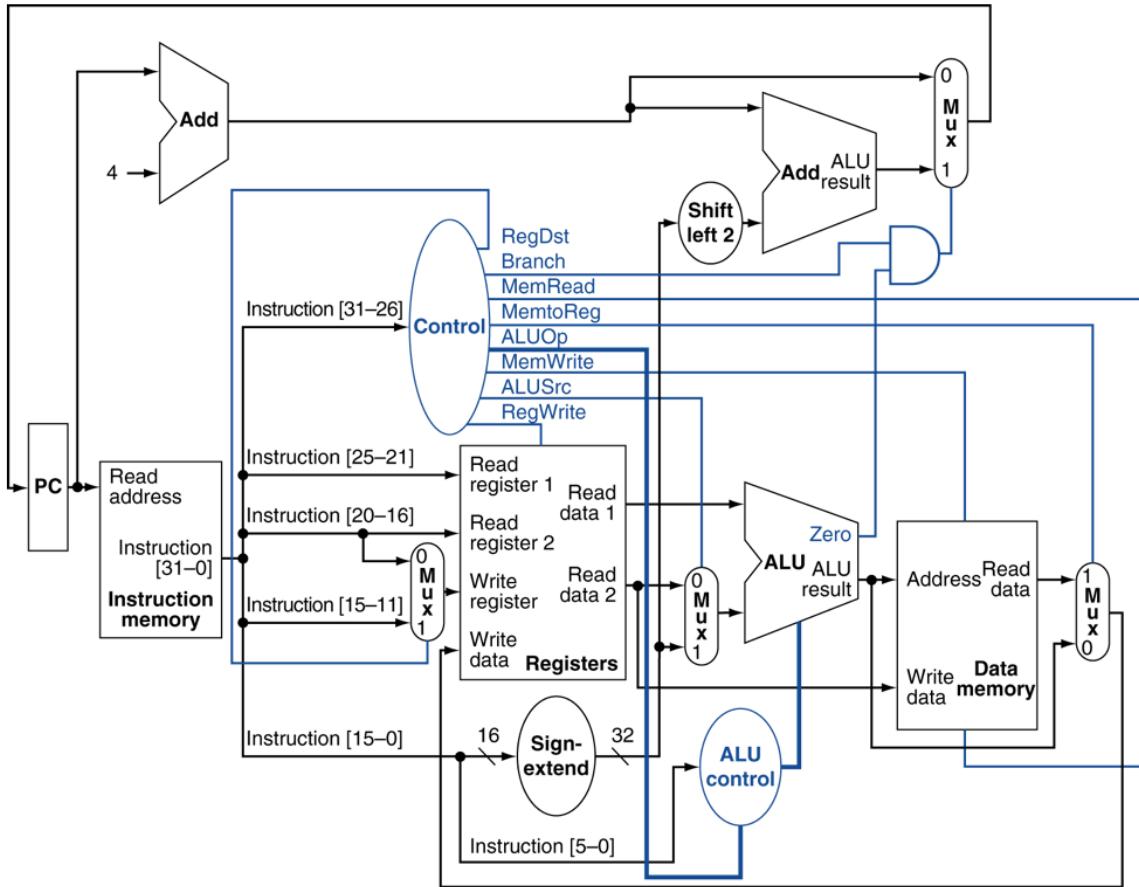


Figure 8a: Single cycle datapath with control signals

- (c) Figure 8c shows a pipelined execution of three instructions **sub**, **and**, and **or**. The **(6x2=12)** pipeline registers are denoted respectively as IF/ID, ID/EX, EX/MEM, and MEM/WB. The data forwarding path between **sub** and **and** has been shown in *green* color and the data forwarding path between **sub** and **or** has been shown in *yellow* color. For each of these two data forwarding cases, write down the data forwarding conditions.

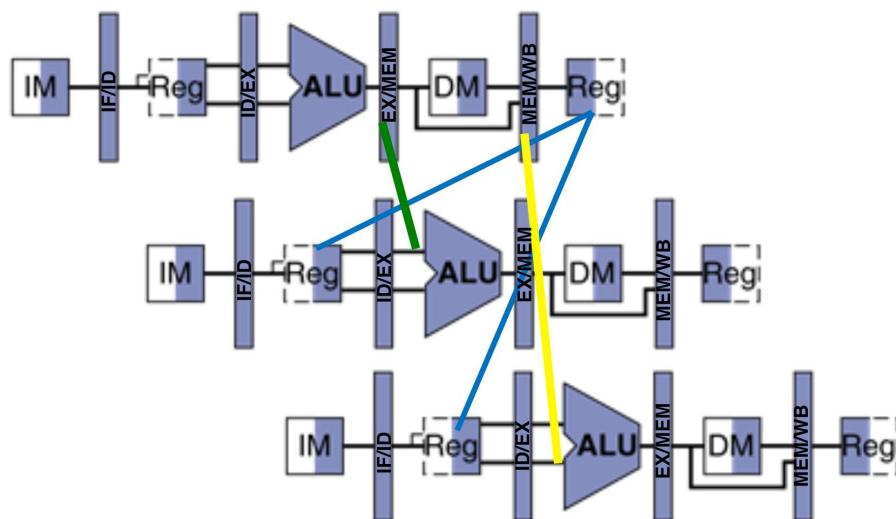


Figure 8c: Pipelined execution with data forwarding

The figures in the margin indicate full marks.

USE SEPARATE SCRIPTS FOR EACH SECTION

### SECTION - A

There are **FOUR** questions in this section. Answer any **THREE**.

*You may choose to answers parts of some questions in the figures provided with the question script. In that case, you must securely attach that page to your answer script after having that duly signed by the honourable invigilator.*

1. (a) Consider the single cycle datapath shown in Figure 1.A. Update the figure to implement the following MIPS instructions: bne, j, jr, jal, slt, lui. Briefly explain your updates. You can make the changes in Figure 1.A and have that page stapled with your script. The question page containing Figure 1.A must then be signed by the invigilator. (30)

(b) Why are single cycle implementations not very efficient? (5)

2. (a) Consider the following code snippet: (10)

```
beq $1, $2, TARGET # branch is/will be taken for this one
```

```
lw $3, 40($4)
```

```
add $2, $3, $4
```

```
sw $2, 40($4)
```

TARGET: or \$1, \$1, \$2

Assume that branch decision is checked using the ALU zero signal (i.e., no separate hardware is placed to decide early). Now can an attempt to flush and stall the pipeline occur simultaneously? If yes, then will there be an issue? Explain with the help of a single cycle illustration. If there is an issue, you need to provide a solution. If there is no issue, you need to clearly explain why not.

- (b) Discuss what are the issues and how can they be handled in the following code snippet. Assume that using extra hardware branch decision is checked as early as possible. Use multicycle illustrations to explain. How many cycles will be needed to complete the code snippet? (10)

```
lw $1, 100($2)
```

```
add $1, $1, $2
```

```
beq $1, $2, TARGET
```

- (c) Discuss how a TLB miss is handled by MIPS architecture. Consider the pipelined datapath in Figure 2.c. that is capable of handling undefined instruction and overflow exceptions. Now, discuss how TLB miss exception may be handled there. You may update Figure 2.c if you feel necessary. You can make the changes in the question page containing Figure 2.c and have that stapled with your script. The question page must then be signed by the invigilator. (15)

## CSE 305

3. (a) With the help of a flowchart show how a real read or write request is processed in a system with a TLB interacting with an appropriate cache and virtual memory. (8)
- (b) Consider a virtual memory system with 40-bit virtual byte address, 16KB pages and 36-bit physical byte address. All the virtual pages are in use and there are valid, protection, dirty and use bits. Disk addresses are not stored in the page table. There is a 2-way set associative TLB with a total of 256 TLB entries. Now answer the following questions: (4+3+12=19)
- What is size of the page table?
  - What will be the size of the page table if we use the inverted page table concept?
  - With the help of an appropriate illustration show how the virtual-to-physical mapping is done. Assume a 16 KB direct-mapped cache with 16 words block.
  - You have designed a processor with 16-entry TLB and 4 KB pages and have exhausted your budget for implementing the system. During trial you have found that programs are accessing at least 2MB of memory at a time. How would your system perform and why? Can you suggest how can you improve the system performance of the system if no more money is available? What if more money is available? (8)

4. (a) Consider three processors with different cache configurations and data as follows: (4+3+12=19)

| Processor/Cache | Processor cycle time | Organization            | Block size (Word) | Instruction Miss rate | Data Miss rate |
|-----------------|----------------------|-------------------------|-------------------|-----------------------|----------------|
| 1               | 420 ps               | Direct-Mapped           | 1                 | 4%                    | 6%             |
| 2               | 420 ps               | Direct-Mapped           | 4                 | 2%                    | 4%             |
| 3               | 310 ps               | Two-way Set Associative | 4                 | 2%                    | 3%             |

For these processors, one half of the instructions contain a data reference and cache miss penalty is 6+ block size in words. The CPI of this workload was measured on Processor 1 (i.e., with cache 1) and was found to be 2.0. Now, answer the following questions.

- Which processor spends the most cycle on cache misses?
- Which processor is the fastest and which one is the slowest?
- For processor 3, show the cache organization and how the cache will be accessed with the help of a detailed illustration.

## CSE 305

### Contd..... Q. No. 4

(b) Suppose you are working on a computer architecture with the following properties:

**(4+6+6=16)**

- a. 32-bit address
- b. byte addressable RAM
- c. 256KB cache
- d. 64 words block
- e. Random replacement algorithm
- f. Write-through scheme

Now answer the following questions:

- a. What is the size of the cache?
- b. What is the overhead if the cache is:
  - i. direct-mapped
  - ii. 2-way set associative
  - iii. fully associative
- c. Consider the current value of \$2 is 1000 in decimal. Assuming direct-mapped organization, find the cache address for the following load word instruction: lw \$1, 18(\$2).

## SECTION – B

There are **FOUR** questions in this section. Answer any **THREE**.

5. (a) Draw the hardware block diagram and the flowchart of the sequential version of the multiplication algorithm. Also, draw the flowchart of the refined version of the algorithm.

**(12)**

(b) Consider a 16-bit floating point representation system, where the exponent is 4 bits long. With this system, represent the floating point numbers  $(-1.25)_{10}$  and  $(0.375)_{10}$  into a bit string of length 16 using appropriate bias.

**(8)**

(c) Determine the decimal value of the floating point number represented by the hexadecimal value  $(DF80\ 7500)_{16}$  according to IEEE 754 single precision floating point format.

**(5)**

(d) Design a 4-bit ALU with three selection variables  $S_0$ ,  $S_1$  and  $S_2$  that performs the following operations on two 4-bit inputs (A and B):

**(10)**

## CSE 305

### Contd..... Q. No. 5(d)

| S <sub>2</sub> | S <sub>1</sub> | S <sub>0</sub> | Operation                    |
|----------------|----------------|----------------|------------------------------|
| 0              | 0              | 0              | Addition                     |
| 0              | 0              | 1              | Transfer A                   |
| 0              | 1              | 0              | Add 1's complement of B to A |
| 0              | 1              | 1              | Increment A                  |
| 1              | 0              | 0              | XOR                          |
| 1              | 0              | 1              | NOT                          |
| 1              | 1              | 0              | AND                          |
| 1              | 1              | 1              | OR                           |

Draw the logic diagram of one typical stage of your designed ALU.

6. (a) What are the different addressing modes of MIPS? Describe and illustrate the calculation process for each addressing mode. (15)
- (b) Suppose you have a 4-bit ALU and you are required to perform a division operation, where the dividend is 11001001 and the divisor is 1011. Show the division process step by step along with the value of each relevant register at each step. (15)
- (c) How many instruction formats are there in MIPS? Write down the name of each instruction format along with the length of each field therein. (5)

7. (a) Consider the following MIPS assembly language code and assume that the code fragment is loaded at location 60000 in the memory: (10)

```

Loop: s11 $t1, $s2, 2
 add $t1, $t1, $s6
 lw $t0, 0($t1)

 beq $t0, $s5, Lp1
 j Exit

Lp1: addi $s2, $s2, 1
 j Loop

```

Exit:

Write down the MIPS machine code with appropriate decimal values for the above assembly code.

- (b) Assume that a processor with 4 GHz clock rate is executing a program that requires the following instruction types. (20)

## CSE 305

### Contd..... Q. No. 7(b)

| Instructions                        | FP  | INT | L/S | BRANCH |
|-------------------------------------|-----|-----|-----|--------|
| Instruction Count ( $\times 10^6$ ) | 160 | 110 | 10  | 16     |
| CPI                                 | 2   | 1   | 5   | 2      |

- (i) Calculate the execution time of the program.
- (ii) If you want the program to run two times faster, then how much should you improve the CPI of **FP** instructions? Explain your finding.
- (iii) If you want the program to run two times faster, then how much should you improve the CPI of **L/S** instructions? Explain your finding.
- (iv) By how much is the execution time changed if the **CPI** of **INT** and **BRANCH** instructions are increased by 15% and the **CPI** of **FP** and **L/S** instructions are decreased by 20%?
- (c) What is the use of Bias in floating point representation? Give the encoding of the floating point "infinity". (5)

8. (a) Consider the following C code which calculates  $a^b$  (a raised power b), (20)

```
int fun (int a, int b)
{
 if (b == 0)
 return 0,
 else if (b % 2 == 0)
 return fun (a*a, b/2);
 return fun (a* a, b/2) *a;
```

Write down the equivalent MIPS assembly code for it. Use a minimal number of MIPS instructions.

- (b) What are the steps of floating point addition? Describe with an example. Also, write down the condition of overflow and underflow during a floating point addition. (10)
- (c) What is the problem in the following code snippet? How can you solve the problem? (5)

| Address | Instruction           |
|---------|-----------------------|
| 005000  | beq \$s0, \$s1, 30500 |

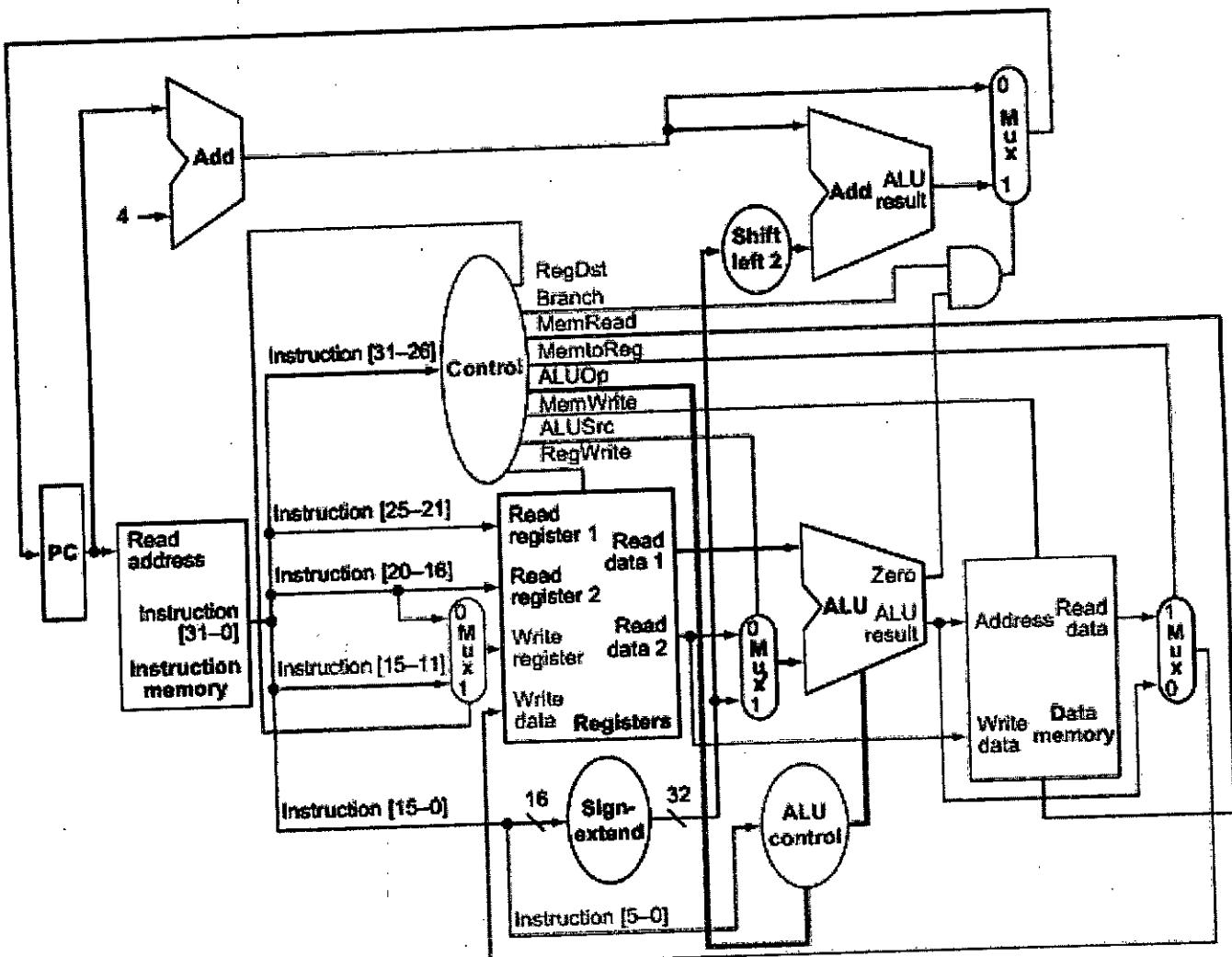


Figure 1.A

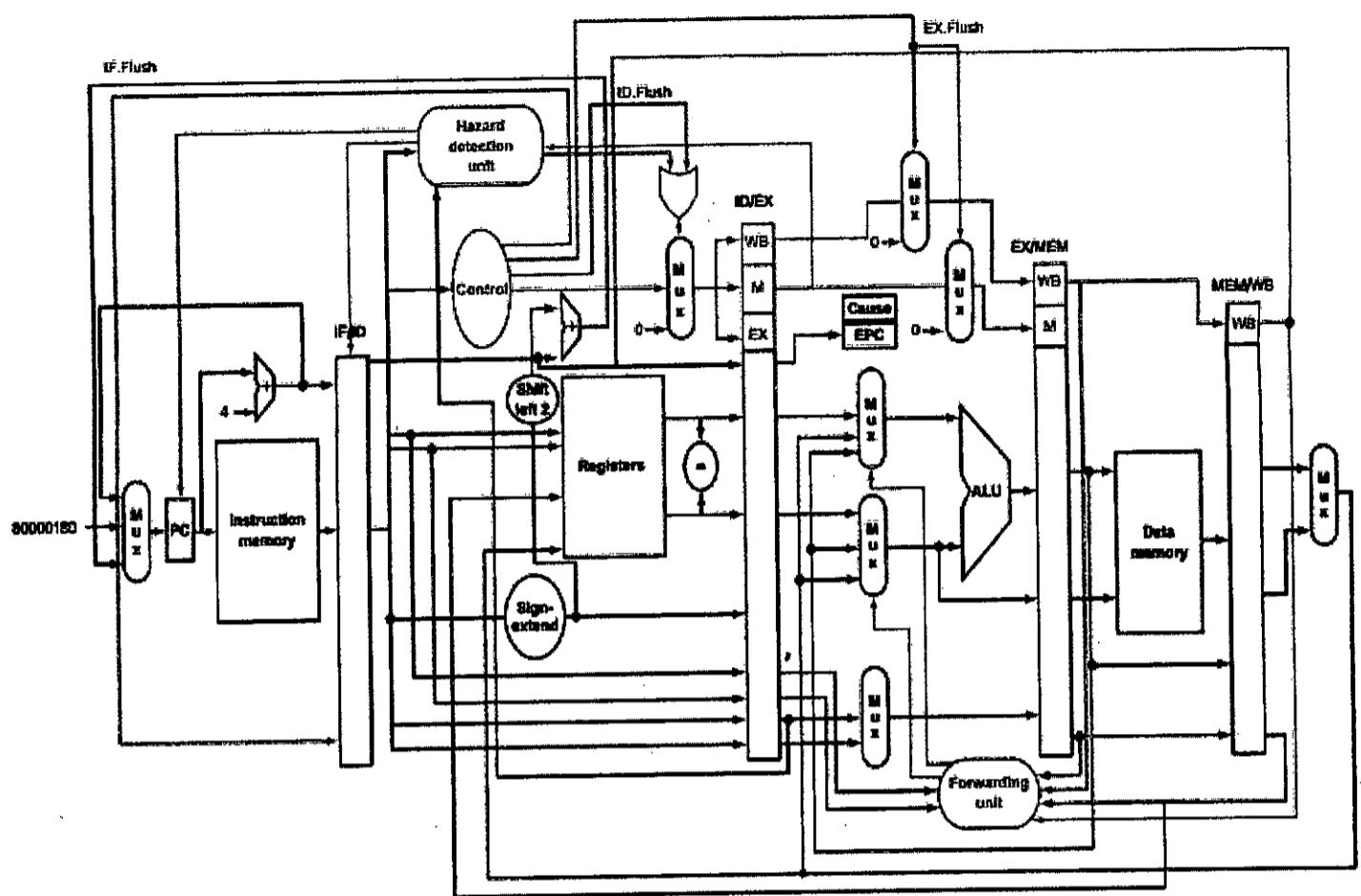


Fig 2.c

**SECTION – A**

There are **FOUR** questions in this section. Answer any **THREE**.

1. (a) Consider the following C code: (10)

$$A = B + C; D = A + E;$$

All the above are memory instructions. The base address is in register t0 and the offsets are A (0), B (4), C (8), D (12), E (16). How many cycles are required to execute the above? Can you reduce the number of cycles using code scheduling? If yes, what will be resulting number of cycles? Show each step.

- (b) Consider the following set of instructions: (15)

add, lw, beq

Draw the complete single cycle data path (not pipelined) and control that can execute the above set of instructions.

- (c) Consider the following information for load word (lw) instruction: (5)

| Instr. | Instruction Fetch | Register read | ALU op | Memory access | Register write |
|--------|-------------------|---------------|--------|---------------|----------------|
| lw     | 200 ps            | 100 ps        | 100 ps | 200 ps        | 100 ps         |

Consider the following program:

lw \$1, 100(\$0)

lw \$2, 200(\$0)

lw \$3, 300(\$0)

What is the speedup for the pipelined datapath over the single cycle datapath for this program? What will be the speedup if we add 3,000,000 more load word (lw) instructions?

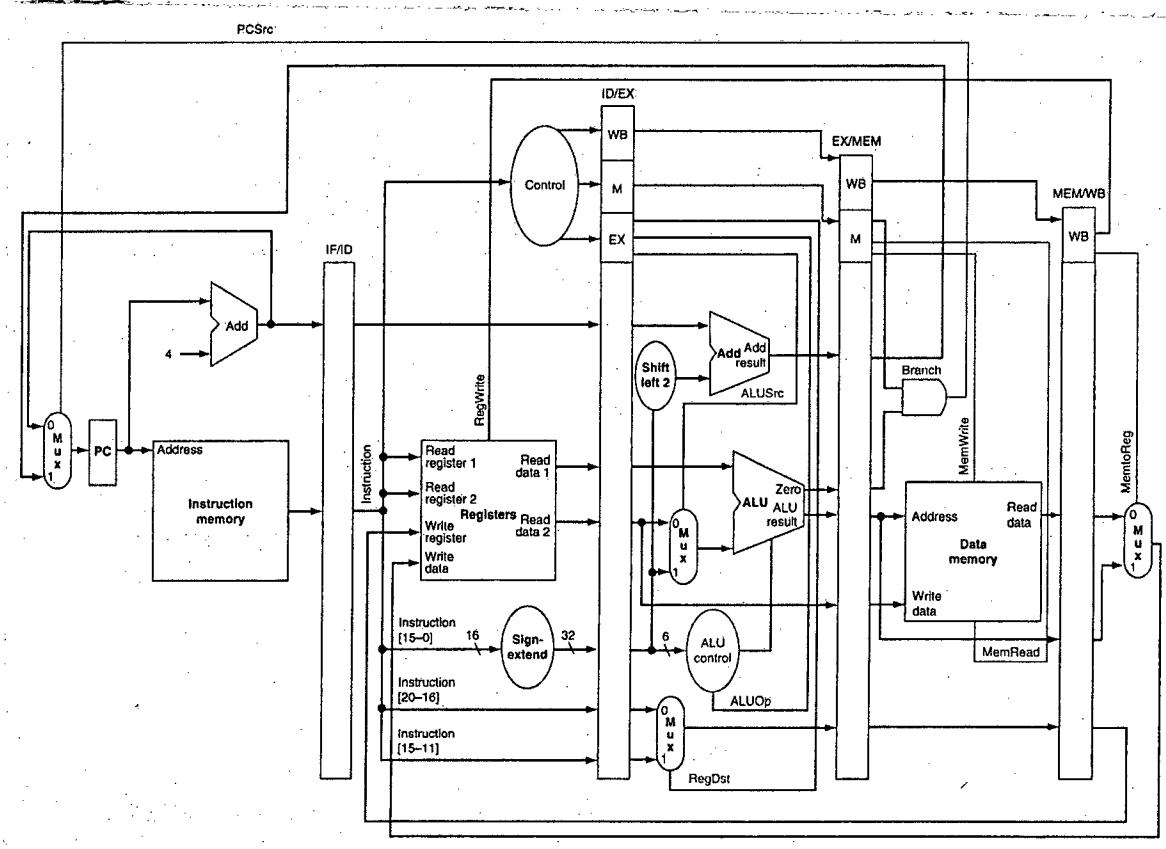
- (d) "Even with a perfect branch predictor, 1-cycle penalty for a taken branch". Why is that? How to solve this? (5)

2. (a) Briefly explain double data hazard with example. How MIPS detect hazard due to the use of load instruction and how MIPS stall the pipeline? (10)

- (b) Design a complete forwarding unit with necessary equations for MIPS considering double data hazard. Draw the complete forwarding unit using the provided figure below. (15)

## CSE 305

Contd... Q. No. 2(b)



(c) Give the MIPS codes for each of the following cases of data hazard for branches: (5)

- (i) Needs one cycle stall to resolve
- (ii) Needs two cycle stalls to resolve

(d) Consider the following code: (5)

outer:

// some code

inner:

// some code

beq \$t0, \$zero, inner

// some code

beq \$t1, \$zero, outer

What is the problem with the above code if we use 1 bit branch predictor? How can you solve that?

3. (a) What are the three sources of cache misses? The specification of an ideal disk is as follows: (10)

sector size: 1 MB, rotational latency: 10000 rpm, average seek time: 5 ms, transfer rate: 100 MB/s, and controller overhead: 0.5 ms. What is the average read time for the disk? If the average seek time can be reduced to 2 ms, then how much improvement can be possible on average read time? (10)

## CSE 305

### Contd... Q. No. 3

- (b) What is meant by spatial and temporal locality? Briefly explain write-through and write-back scheme for cache. How do write-through and write-back schemes handle write-miss? (15)
- (c) Briefly explain how page table works with respect to virtual address, physical address, main memory and disk. (5)
- (d) Assume the miss rate of an instruction cache is 3%, the miss rate of the data cache is 5%, and the frequency of all loads and stores is 40%. If a processor has a CPI of 2 without any memory stalls and the miss penalty is 200 cycles for all misses, determine how much faster a processor would run with a perfect cache that never missed. Also, determine the amount of execution time spent on memory stalls. (5)
4. (a) Suppose we have a processor with a base CPI of 1.0, assuming all references hit in the primary cache and a clock rate of 4 GHz. Assume a main memory that has an access time of 200 ns, including all the miss handling. Suppose the miss rate per instruction at the primary cache is 4%. How much faster will the processor be if we add a second level cache that has a 10 ns access time for either a hit or a miss and is large enough to reduce the miss rate to main memory to 1%? How much faster will the processor be if we add a third level cache that has a 40 ns access time for either a hit or a miss and is large enough to reduce the miss rate to main memory to 0.5%? (10)
- (b) Assume there are three small caches, each consisting of four one-word blocks and uses LRU replacement policy. One cache is fully associative, the second is two-way set-associative, and the third is direct-mapped. Find the number of misses for each cache organization, given the following sequence of block addresses: (15)
- 1, 3, 5, 1, 3, 1, 3, 5, 3, 1, 5, 3
- (c) What is meant by TLB? Why TLB is necessary in a virtual memory system? (5)
- (d) Explain (with flow chart) the interaction between TLB and data cache for only write operation. (5)

### SECTION – B

There are **FOUR** questions in this section. Answer any **THREE**.

5. (a) Suppose there are three classes of instructions A, B and C in a particular instruction set architecture with CPIs 1.2, 2 and 2.5, respectively. The number of instructions from each class in two separate programs P1 and P2 are as follows: (8+7=15)

## CSE 305

### Contd... Q. No. 5(a)

| Programs | Instruction Classes |    |    |
|----------|---------------------|----|----|
|          | A                   | B  | C  |
| P1       | 40                  | 10 | 16 |
| P2       | 12                  | 13 | 40 |

- (i) Which program will execute faster and by how much?
- (ii) By how much should we improve the CPI of Instruction Class B to make P1 execute two times faster? Explain your finding.
- (b) Consider a half precision floating point representation system where each floating point is represented by 16 bits in which the exponent is 4 bits long. Now convert the two floating points  $(0.75)_{10}$  and  $(1.25)_{10}$  into the mentioned representation system and then demonstrate the addition of them step by step along with the values of exponent and fraction fields at every step. (15)
- (c) Determine the decimal value of the floating point represented by the hexadecimal value  $(7F80\ 0005)_{hex}$  according to IEEE 754 single precision floating point format. (5)
6. (a) Design an ALU with three selection variables  $S_2$ ,  $S_1$  and  $S_0$  that performs the following operations on two 4-bit inputs (A and B): (25)

| $S_2$ | $S_1$ | $S_0$ | Operation                    |
|-------|-------|-------|------------------------------|
| 0     | 0     | 0     | Addition                     |
| 0     | 0     | 1     | Transfer A                   |
| 0     | 1     | 0     | Decrement A                  |
| 0     | 1     | 1     | Add 1's complement of B to A |
| 1     | 0     | 0     | XOR                          |
| 1     | 0     | 1     | AND                          |
| 1     | 1     | 0     | NOT                          |
| 1     | 1     | 1     | OR                           |

Draw the logic diagram of one typical stage of your designed ALU.

- (b) For each arithmetic operation listed in question 6(a), find out the conditions under which the output carry will be equal to 1. (10)

## CSE 305

7. (a) The following C code segment can sort an array of integers in ascending order. Write down the equivalent MIPS assembly code for it. Assume appropriate registers for the variables.

(25)

```
void swap(int v[], int k)
{
 int temp;
 temp = v[k];
 v[k] = v[k+1];
 v[k+1] = temp;
}

void sort (int v[], int n)
{
 int i, j;
 for (i = 0; i < n; i += 1) {
 for (j = i - 1; j >= 0 && v[j] > v[j + 1]; j -= 1) {
 swap(v, j);
 }
 }
}
```

- (b) Suppose you are required to jump to the instruction at address  $(4000\ 0001)_{hex}$ .

Write down the MIPS assembly code for this task.

(10)

8. (a) Suppose you have a 4-bit ALU and you are required to perform a division operation, where the dividend is 11010101 and the divisor is 1011. Show the division process step by step along with value of each relevant register at every step.

(15)

- (b) Write down the equivalent MIPS assembly code for the following function, written in C programming language. Assume appropriate registers for the variables.

(10)

```
float convert (float cels)
{
 float ferh = (9.0 * cels) / 5.0;
 ferh = ferh + 32.0;
 return ferh;
}
```

- (c) How many instruction formats are there in MIPS? Write down the name of each instruction format along with the length of each field in it.

(10)

BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY, DHAKA

L-3/T-1 B. Sc. Engineering Examinations 2016-2017

Sub : CSE 305 (Computer Architecture)

Full Marks : 210

Time : 3 Hours

The figures in the margin indicate full marks.

USE SEPARATE SCRIPTS FOR EACH SECTION

**SECTION – A**There are **FOUR** questions in this Section. Answer any **THREE**.

1. Consider a CPU which has PC, MAR, MDR, IR, ALU and Instruction Decoder. There are six general purpose registers R0, R1, R2, R3, R4, R5 and a temporary register TEMP inside the CPU. The CPU gets one input directly from the bus and the other input through a register Y. The output of the ALU is sent to the bus through a register Z. The ALU can perform ten arithmetic-logic operations and has a special input for Carry-in.

(a) Draw a block diagram of single bus datapaths inside the CPU following the description above. (10)

(b) Show step-wise control signals for executing the instruction

**Add (R3), R1**

which adds the contents of a memory location pointed to by Register **R3** to Register **R1**. Assume that the instruction is stored in memory and PC contains the address of the instruction. (10)

(c) Design a control word for the CPU above such that the number of bits in the control word is significantly smaller than the number of control signals. (10)

(d) What are the relative merits of horizontal and vertical microinstruction formats? (5)

2. Answer the following questions based on MIPS architecture shown in Figure 1.

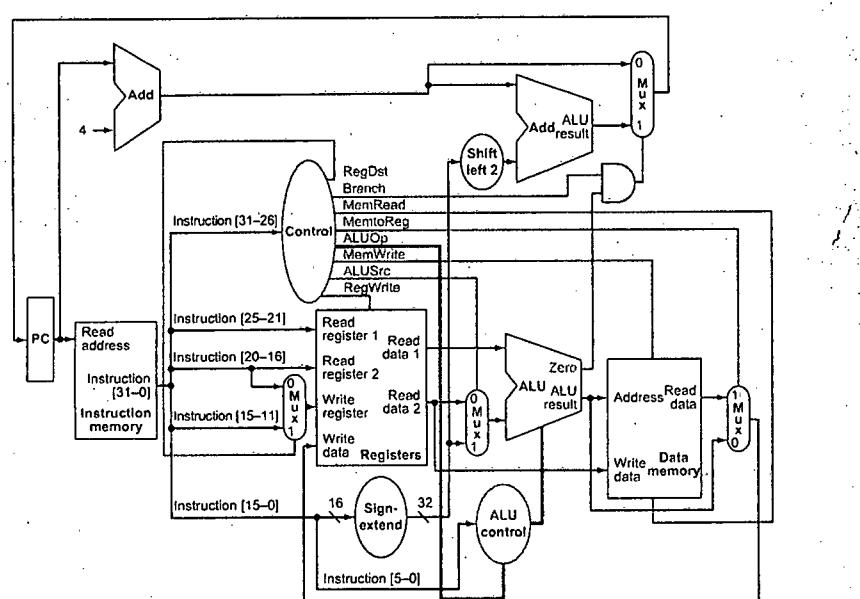


Figure 1: Figure for Question 2.

## CSE 305

### Contd ... Q. No. 2

- (a) Write the functions of the control signals ALUSrc, ALUOp, RegDst, MemWrite and RegWrite. (5)
- (b) Explain how the ALU control bits are set depending on the ALUOp control bits and the function codes for the R-type instruction. (10)
- (c) Describe the datapaths for R-type instructions. (10)
- (d) How is the content of the PC updated for the branch (beq) instruction? (5)
- (e) What are the two inputs of the MUX at the input of the ALU? (5)
3. (a) What do you mean by the principle of locality of reference? Discuss the role of the principle of locality of reference in designing memory hierarchy. (5)
- (b) Answer the following questions for a memory system that uses 32-bit address at the byte level, and a cache that uses a 64-byte line size. (15)
- (i) Assume a direct cache with a tag field in the address of 20 bits. Show the address format and determine the number of addressable units, the number of blocks in main memory, and the number of lines in the cache.
- (ii) Assume an associative cache. Show the address format and determine the number of addressable units, the number of blocks of main memory, the number of lines in the cache and the size of the tag.
- (iii) Assume a four-way set-associative cache with a tag field in the address of 9 bits. Show the address format and determine the number of addressable units, the number of blocks in main memory, the number of lines in a set, the number of sets in the cache, and the number of lines in the cache.
- (c) The access time of the cache  $M_1$  of a single cache memory system is 6ns/bit during hit and that of the main memory  $M_2$  is 900ns/bit during miss. Calculate the block transfer time and the access efficiency of the memory system at the hit ratio of 0.8. (5)
- (d) Describe SISD, SIMD and MISD architectures with block diagrams. (10)
4. (a) What are pipeline hazards? Explain a structural pipeline hazard with a time-space diagram. What type of pipeline hazards occurs in the following code segment? How can it be overcome? (10)

```
 lw $t1, 0($t0)
 lw $t2, 4($t0)
 add $t3, $t1,$t2
 sw $t3, 12($t0)
 lw $t4, 8($t0)
 add $t5, $t1,$t4
 sw $t5, 16($t0)
```

CSE 305

**Contd ... Q. No. 4**

- (b) Perform the necessary modification of the single-cycle data path in Figure 2 for pipeline implementation by inserting Pipeline Registers, Forwarding Unit and Hazard Detection Unit. Describe the modification step by step and draw the final modified datapath with all necessary control signals. (15)

(c) Compare the characteristics of typical CISC and RISC architectures. (6)

(d) Consider a bus-based shared memory multiprocessor system. It is constructed using processors with speed of  $10^6$  fetches/sec. and a bus with a peak bandwidth of  $10^5$  fetches/sec. The caches are designed to support a hit rate of 90%. (4)

  - (i) What is the maximum number of processors that can be supported by this system?
  - (ii) What hit rate is needed to support a 20-processor system?

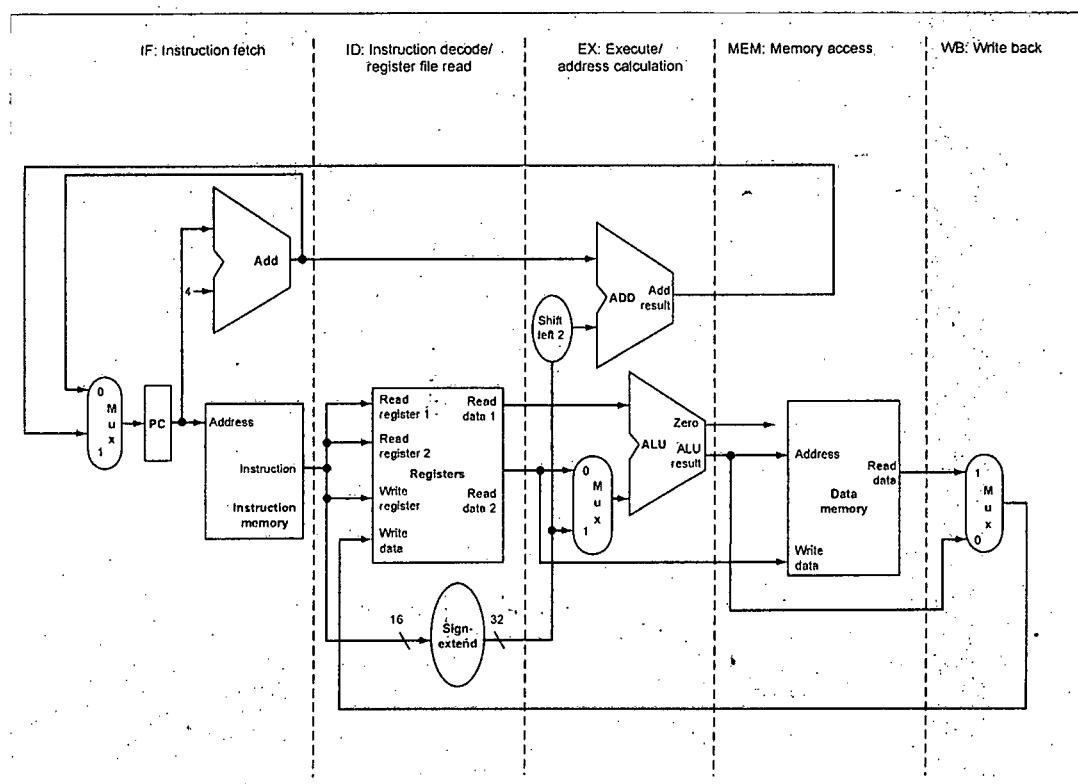


Figure 2: Figure for Question 4(b)

## CSE 305

### SECTION – B

There are **FOUR** questions in this Section. Answer any **THREE** questions.

5. (a) “Computer architectures have invented several great ideas in the last 60 years of computer design. These ideas are so powerful that they have lasted long after the first computer that used them, with newer architects demonstrating their admiration by imitating their predecessors.” These ideas include: (10)

- (i) Design for Moore’s Law
- (ii) Use abstraction to simplify design
- (iii) Make the common case faster
- (iv) Performance via parallelism
- (v) Performance via pipelining
- (vi) Performance via prediction
- (vii) Hierarchy of memories
- (viii) Dependability via redundancy

- You are given several design features of MIPS and other modern computing systems in the following. Now, please **identify** that which of the above great ideas are behind these design decisions.

- (i) While implementing instructions at hardware level, it is possible to implement all the instructions as single cycle instructions. However, modern designers do not use this single-cycle design as it is inefficient. (hint: for single cycle implementation, the clock cycle must have the same length for every instruction and the longest instruction determine the clock cycle length.)
  - (ii) Modern computers use a technique called prefetching. “In prefetching, a block of data is brought into the cache before it is actually referenced. Many microprocessors use hardware prefetching to access the block needed in the future that may be difficult for software to notice”.
- (b) Lenovo x3650 M5 server containing a 32 cores (Intel Xeon E5-2699v4) obtained following results in SPEC test. Find the spec ratio for the computer. (10)

| Program Name             | Execution Time | Reference Time | Spec Ratios |
|--------------------------|----------------|----------------|-------------|
| perlbench                | 236            | 9770           |             |
| bzip2                    | 375            | 9650           |             |
| gcc                      | 208            | 8050           |             |
| mcf                      | 133            | 9120           |             |
| gobmk                    | 337            | 10490          |             |
| hmmer                    | 105            | 9330           |             |
| sjeng                    | 340            | 12100          |             |
| libquantum               | 2              | 20720          |             |
| h264ref                  | 379            | 22130          |             |
| omnetpp                  | 119            | 6250           |             |
| astar                    | 195            | 7020           |             |
| xalancbmk                | 86             | 6900           |             |
| Spec ratui (SpecInt2006) | -              | -              |             |

## CSE 305

### Contd ... Q. No. 5

- (c) Assume a processor with 3 GHz clock rate is executing a program that requires following instructions, (15)

| Instructions →                      | FP | INT | L/S | BRANCH |
|-------------------------------------|----|-----|-----|--------|
| Instruction Count ( $\times 10^6$ ) | 40 | 80  | 90  | 15     |
| CPI                                 | 1  | 1   | 3   | 2      |

- (i) Calculate the execution time of the program.  
(ii) By how much must we improve the CPI of FP instructions if we want the program to run two times faster?  
(iii) By how much is the execution time program changed (improved or degraded) if the CPI of INT and FP instruction is increased by 20% and the CPI of L/S and Branch is reduced by 10%?  
6. (a) A simple MIPS assembly code snippet and its corresponding MIPS machine code is given below. Note that, the former one is not complete. Please complete it with appropriate MIPS assembly code. (10)

|      |      |    |       |       |       |       |         |
|------|------|----|-------|-------|-------|-------|---------|
| sll  | 1000 | 0  | 00000 | 10000 | 01111 | 00010 | 0000000 |
| add  | 1004 | 0  | 01111 | 10001 | 01111 | 00000 | 1000000 |
| lw   | 1008 | 35 | 01111 | 01000 | 00000 | 00000 | 0000000 |
| bne  | 1012 | 5  | 01000 | 10111 | 00000 | 00000 | 0000001 |
| addi | 1016 | 8  | 10000 | 10000 | 00000 | 00000 | 0000001 |
| j    | 1020 | 2  | 00000 | 00000 | 00000 | 00011 | 111010  |

- (b) What support does MIPS instruction set architecture provide for synchronization? Explain with an example. (10)  
(c) The following four design principles have been guiding the instruction-set designers to find a balance between the number of instructions needed to execute a program, the number of clock cycles needed by an instruction, and the speed of the clock. (15)
- (i) Simplicity favors regularity
  - (ii) Smaller is faster
  - (iii) Good design demands good compromise
  - (iv) Make the common case faster
- You are given several design features of MIPS instruction-set architecture in the following. Now, please identify that which of the above principles are behind these design decisions.

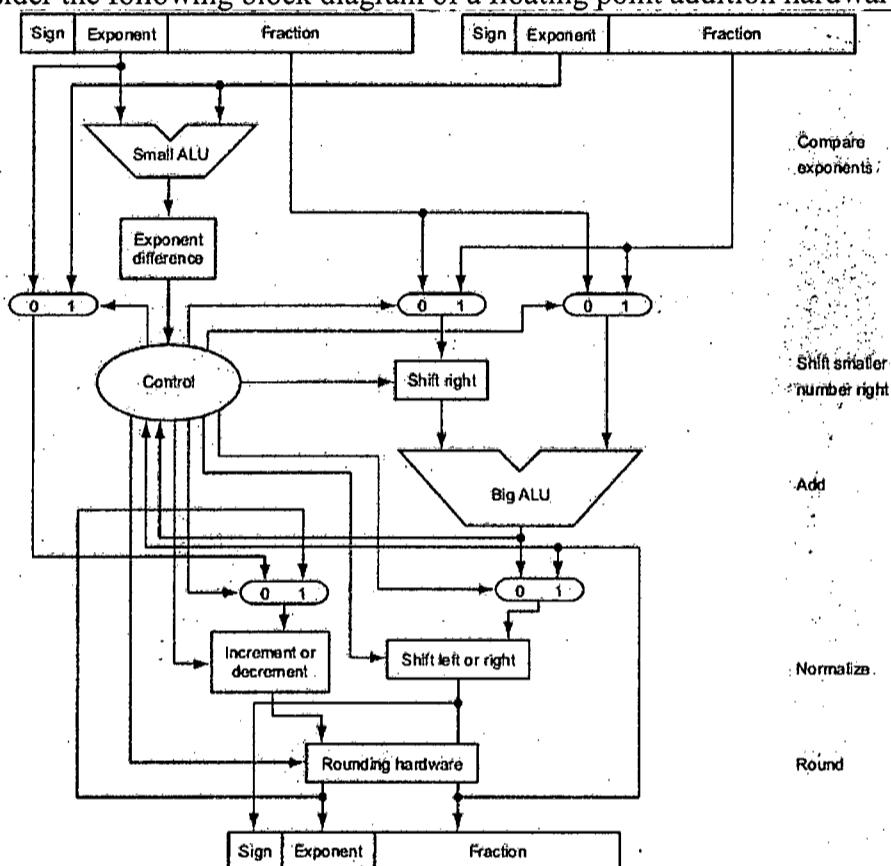
## CSE 305

### Contd ... Q. No. 6(c)

- (i) keeping instruction immediate value in a range of 16-bits
- (ii) always requiring three register operands in arithmetic instructions
- (iii) using PC-relative addressing for conditional branching
- (iv) keeping 32 registers rather than many more
- (v) keeping the register fields in the same place in each instruction format

7. (a) Please draw the illustrations of the five MIPS addressing modes. Also give an example instruction for each of the five addressing modes. (10+10)

(b) Consider the following block diagram of a floating point addition hardware. (15)



-Using the given hardware, simulate the floating addition algorithm by filling up the following tables. The two numbers for addition are,  $-(0.25)_{10}$  and  $(0.625)_{10}$

|                             |                |                |  |
|-----------------------------|----------------|----------------|--|
|                             | $-(0.25)_{10}$ | $(0.625)_{10}$ |  |
| Binary                      |                |                |  |
| Step1: Exponent Matching    |                |                |  |
| Step2: Significand Addition |                |                |  |
| Step3: Normalization        |                | =              |  |
| Step4: Rounding             |                |                |  |

## CSE 305

8. (a) The following table illustrates the various combinations of two operands for binary addition-subtraction. Please identify the overflow conditions in that table. Then, formulate the overflow boolean variable as a logical function of various input variables (You don't need to simplify the function). (5+5)

| $O$ | $S_A$ | $S_B$ | $S_R$ | $V$ |
|-----|-------|-------|-------|-----|
| 0   | 0     | 0     |       |     |
| 0   | 0     | 1     |       |     |
| 0   | 1     | 0     |       |     |
| 0   | 1     | 1     |       |     |
| 1   | 0     | 0     |       |     |
| 1   | 0     | 1     |       |     |
| 1   | 1     | 0     |       |     |
| 1   | 1     | 1     |       |     |

Here,

$$\text{Operation, } O = \begin{cases} 0, & A + B \\ 1, & A - B \end{cases}$$

$$\text{Sign bit of Operand A, } S_A = \begin{cases} 0, & A \geq 0 \\ 1, & A < 0 \end{cases}$$

$$\text{Sign bit of Operand B, } S_B = \begin{cases} 0, & B \geq 0 \\ 1, & B < 0 \end{cases}$$

$$\text{Sign bit of Result R, } S_R = \begin{cases} 0, & R \geq 0 \\ 1, & R < 0 \end{cases}$$

---


$$\text{Overflow, } V = \begin{cases} 1, & \text{Overflow occurred} \\ 0, & \text{No overflow possible} \end{cases}$$

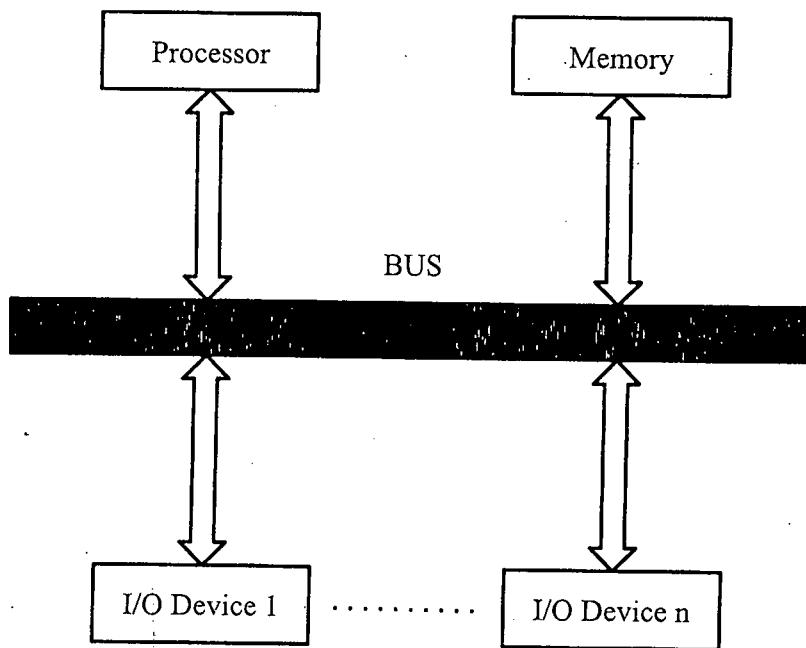

---

## CSE 305

### Contd ... Q. No. 8

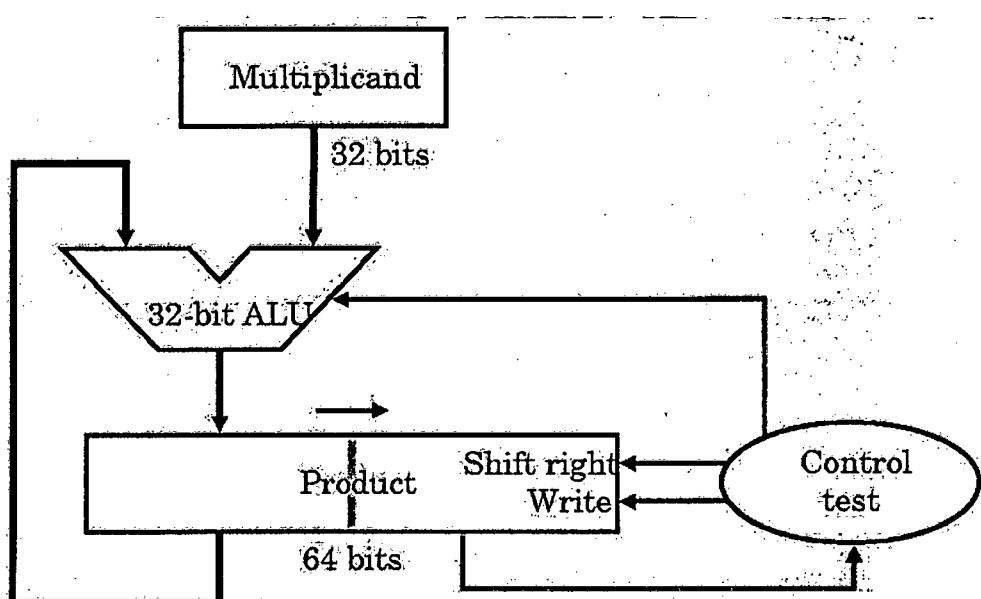
(b) In the following figure, a single bus computer architecture was drawn where, various devices like processor, memory, and I/O devices are connected with each other via a single BUS. Please note that, these devices do not operate at a similar rate. Then, how can you redesign this single bus computer architecture so that these devices can interact with each other smoothly using this single bus. Just draw your redesigned single bus computer architecture.

(10)



(c) Consider the following multiplication hardware.

(15)



-Using the given hardware, simulate the unsigned multiplication algorithm by filling up the following tables for Multiplicand, 0110 and Multiplier, 0110.

**CSE 305**

Contd ... Q. No. 8(c)

|   |  |  |  |  |  |  |  |  |
|---|--|--|--|--|--|--|--|--|
|   |  |  |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |
|   |  |  |  |  |  |  |  |  |
|   |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |
|   |  |  |  |  |  |  |  |  |
|   |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |
|   |  |  |  |  |  |  |  |  |
|   |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |
|   |  |  |  |  |  |  |  |  |
|   |  |  |  |  |  |  |  |  |

**SECTION – A**

There are **FOUR** questions in this section. Answer any **THREE**.

1. (a) Consider three different processors P1, P2 and P3 executing the same instruction set. P1 has a 3 GHz clock rate and a CPI of 1.5. P2 has a 2.5 GHz clock rate and a CPI of 1.0. P3 has a 4.0 GHz clock rate and has a CPI of 2.2. (10)
- (i) Which processor has the highest performance expressed in instructions per second?
  - (ii) If the processors each execute a program in 10 seconds, find the number of cycles and the number of instructions.
  - (iii) We are trying to reduce the execution time by 30% but this leads to an increase of 20% in the CPI. What clock rate should we have to get this time reduction?
- (b) The following table shows the numbers of instructions for a program. (10)

| Arithmetic | Store | Load | Branch |
|------------|-------|------|--------|
| 500        | 50    | 100  | 50     |

The following table shows the number of cycles taken by each instruction.

| Arithmetic | Store | Load | Branch |
|------------|-------|------|--------|
| 1          | 5     | 5    | 2      |

- (i) What is the execution time of the program in a 4 GHz processor?
  - (ii) What is the CPI of the program?
  - (iii) If the number of load instructions can be reduced by half, what is the speed-up and CPI?
- (c) Assume for arithmetic, load/store, and branch instructions, a processor has CPIs of 1, 12, and 5, respectively. Also, assume that on a single processor a program requires the execution of 2.56 billion arithmetic instructions, 1.28 billion load/store instructions, and 256 million branch instructions. Assume that each processor has a 2 GHz clock frequency. Assume that, as the program is parallelized to run over multiple cores, the number of arithmetic and load/store instructions per processor is divided by  $0.7 \times p$  (where  $p$  is the number of processors) but the number of branch instructions per processor remains the same. ( $1 \text{ billion} = 10^9$ ,  $1 \text{ million} = 10^6$ ). (15)
- (i) Find the total execution time for this program on 1, 2, and 4 processors, and show the relative speedup of the 2, and 4 processors compared to the single processor.
  - (ii) If the CPI of the arithmetic instructions was doubled, what would the impact be on the execution time of the program on 1, 2, and 4 processors?
  - (iii) How much should the CPI of load/store instructions be reduced for a single processor to match the performance of 4 processors using the original CPI values?

## CSE 305

2. (a) Consider the following information for two different computers.

(5)

| Computer | Number of Instructions | Clock Rate | CPI |
|----------|------------------------|------------|-----|
| A        | 10 Billion             | 4 GHz      | 1   |
| B        | 8 Billion              | 4 GHz      | 1.1 |

(i) Which has the highest MIPS rating?

(ii) Which is faster?

- (b) Consider the following information for load word (lw) instruction

(5)

| Instr. | Instruction Fetch | Register read | ALU op | Memory access | Register write |
|--------|-------------------|---------------|--------|---------------|----------------|
| lw     | 200 ps            | 100 ps        | 200 ps | 200 ps        | 100 ps         |

Consider the following program:

lw \$1, 100(\$0)

lw \$2, 200(\$0)

lw \$3, 300(\$0)

What is the speedup for the pipelined datapath over the single cycle datapath for this program? What will be the speedup if we add 2,000,000 more load word (lw) instructions?

(c) Draw the complete MIPS pipelined datapath and control for load (lw) instruction with pipelined registers.

(10)

(d) Consider the following C code

(10)

$$A = B + C; \quad D = A + E;$$

All the above are memory instructions. The base address is in register t0 and the offset are A(0), B(4), C(8), D(12), E(16). How many cycles are required to execute the above? Can you reduce the number of cycles using code scheduling? If yes, what will be the resulting number of cycles? Show each step.

(e) "Even with a perfect branch predictor, 1-cycle penalty for a taken branch". Why is that? How to solve this?

(5)

3. (a) What do you mean by double data hazard? Write down the equations to detect forwarding in MIPS considering double data hazard? How MIPS detect hazard due to the use of load instruction?

(10)

(b) Briefly explain write-through and write-back scheme for cache? How does write-through and write-back schemes handle write-miss?

(10)

(c) Briefly explain how page table works with respect to virtual address, physical address, main memory and disk.

(5)

(d) Though both TLB and data cache are cache memory, how are they different from each other? Explain with flow chart the interaction between TLB and data cache for both read and write operation.

(10)

## CSE 305

4. (a) Assume the miss rate of an instruction cache is 3% and the miss rate of the data cache is 5%. If a processor has a CPI of 2 without any memory stalls and the miss penalty is 100 cycles for all misses, determine how much faster a processor would run with a perfect cache that never missed. Assume the frequency of all loads and stores is 40%. What is the amount of execution time spent on memory stalls? (5)

- (b) For a direct-mapped cache design with a 32-bit address, the following bits of the address are used to access the cache. (10)

| Tag   | Index | Offset |
|-------|-------|--------|
| 31-10 | 9-5   | 4-0    |

What is the cache block size (in words) and how many entries does the cache have? Starting from power on, the following byte-addressed cache references are recorded.

0, 4, 16, 132, 232, 160, 1024, 30, 140, 3100, 180, 2180

How many blocks are replaced and what is the hit ratio?

- (c) Assume there are three small caches, each consisting of four one-word blocks and uses LRU replacement policy. One cache is fully associative, a second is two-way set-associative, and the third is direct-mapped. Find the number of misses for each cache organization given the following two independent sequences of block addresses: (15)

0, 2, 4, 8, 10, 12, 14, 16, 0

1, 3, 5, 1, 3, 1, 3, 5, 3

- (d) Suppose we have a processor with a base CPI of 1.0, assuming all references hit in the primary cache, and a clock rate of 4 GHz. Assume a main memory access time of 100 ns, including all the miss handling. Suppose the miss rate per instruction at the primary cache is 3%. How much faster will the processor be if we add a secondary cache that has a 5 ns access time for either a hit or a miss and is large enough to reduce the miss rate to main memory to 1%. (5)

**www.prokoushol.com**

### SECTION – B

There are **FOUR** questions in this section. Answer any **THREE**.

5. (a) Produce the compiled MIPS code for the following C function. Assume that addresses of dest, source in \$a0, and \$a1. Use a minimal number of MIPS assembly instructions. (25)

```
void rstrcpy (char *dest, char *source)
{
 if (*source)
 {
 *dest++=*source++;
 rcopy (dest, source);
 }
 else
 *dest='\0';
}
```

## CSE 305

### Contd ... Q. No. 5

- (b) What is the MIPS assembly code to load this 32-bit constant into register \$s0? (6)
- 0001 0000 0011 1101 0000 1001 0000 0000
- (c) What do you understand by *pseudoinstructions*? Give two examples of *pseudoinstructions* that MIPS assembler accepts. (4)
6. (a) What are the steps of floating point addition? Describe with an example. (12)
- (b) What are the different addressing modes of MIPS? Describe and illustrate the calculation process for each addressing mode. (15)
- (c) What is data race? What instructions are available in MIPS to address this issue? How these instructions work? (8)
7. (a) Assume that you have four different threads. Show how the four threads could be combined to execute on the processor more efficiently using following three multithreading options: (15)

- (i) A superscalar with coarse-grained multithreading
- (ii) A superscalar with fine-grained multithreading
- (iii) A superscalar with simultaneous multithreading

How the four threads would execute independently on a superscalar without any multithreading support is given below.

|        |     | Issue Slots → |          |          |          |
|--------|-----|---------------|----------|----------|----------|
|        |     | Thread A      | Thread B | Thread C | Thread D |
| Time ↓ | 1   | A A           | B B B    | C C C    | D        |
|        | 2   | A A A         | B        |          |          |
|        | 3   | A A           | B        |          |          |
|        | 4   | A A A A       | B        |          |          |
|        | 5   |               |          |          |          |
|        | 6   |               |          |          |          |
|        | 7   |               |          |          |          |
|        | 8   |               |          |          |          |
|        | 9   |               |          |          |          |
|        | 10  |               |          |          |          |
|        | 11  |               |          |          |          |
|        | 12  |               |          |          |          |
|        | 13  |               |          |          |          |
|        | 14  |               |          |          |          |
|        | 15  |               |          |          |          |
|        | 16  |               |          |          |          |
|        | 17  |               |          |          |          |
|        | 18  |               |          |          |          |
|        | 19  |               |          |          |          |
|        | 20  |               |          |          |          |
|        | 21  |               |          |          |          |
|        | 22  |               |          |          |          |
|        | 23  |               |          |          |          |
|        | 24  |               |          |          |          |
|        | 25  |               |          |          |          |
|        | 26  |               |          |          |          |
|        | 27  |               |          |          |          |
|        | 28  |               |          |          |          |
|        | 29  |               |          |          |          |
|        | 30  |               |          |          |          |
|        | 31  |               |          |          |          |
|        | 32  |               |          |          |          |
|        | 33  |               |          |          |          |
|        | 34  |               |          |          |          |
|        | 35  |               |          |          |          |
|        | 36  |               |          |          |          |
|        | 37  |               |          |          |          |
|        | 38  |               |          |          |          |
|        | 39  |               |          |          |          |
|        | 40  |               |          |          |          |
|        | 41  |               |          |          |          |
|        | 42  |               |          |          |          |
|        | 43  |               |          |          |          |
|        | 44  |               |          |          |          |
|        | 45  |               |          |          |          |
|        | 46  |               |          |          |          |
|        | 47  |               |          |          |          |
|        | 48  |               |          |          |          |
|        | 49  |               |          |          |          |
|        | 50  |               |          |          |          |
|        | 51  |               |          |          |          |
|        | 52  |               |          |          |          |
|        | 53  |               |          |          |          |
|        | 54  |               |          |          |          |
|        | 55  |               |          |          |          |
|        | 56  |               |          |          |          |
|        | 57  |               |          |          |          |
|        | 58  |               |          |          |          |
|        | 59  |               |          |          |          |
|        | 60  |               |          |          |          |
|        | 61  |               |          |          |          |
|        | 62  |               |          |          |          |
|        | 63  |               |          |          |          |
|        | 64  |               |          |          |          |
|        | 65  |               |          |          |          |
|        | 66  |               |          |          |          |
|        | 67  |               |          |          |          |
|        | 68  |               |          |          |          |
|        | 69  |               |          |          |          |
|        | 70  |               |          |          |          |
|        | 71  |               |          |          |          |
|        | 72  |               |          |          |          |
|        | 73  |               |          |          |          |
|        | 74  |               |          |          |          |
|        | 75  |               |          |          |          |
|        | 76  |               |          |          |          |
|        | 77  |               |          |          |          |
|        | 78  |               |          |          |          |
|        | 79  |               |          |          |          |
|        | 80  |               |          |          |          |
|        | 81  |               |          |          |          |
|        | 82  |               |          |          |          |
|        | 83  |               |          |          |          |
|        | 84  |               |          |          |          |
|        | 85  |               |          |          |          |
|        | 86  |               |          |          |          |
|        | 87  |               |          |          |          |
|        | 88  |               |          |          |          |
|        | 89  |               |          |          |          |
|        | 90  |               |          |          |          |
|        | 91  |               |          |          |          |
|        | 92  |               |          |          |          |
|        | 93  |               |          |          |          |
|        | 94  |               |          |          |          |
|        | 95  |               |          |          |          |
|        | 96  |               |          |          |          |
|        | 97  |               |          |          |          |
|        | 98  |               |          |          |          |
|        | 99  |               |          |          |          |
|        | 100 |               |          |          |          |

- (b) Find the maximum number of processors each having clock rate  $i$  with cache memories that a bus with Bandwidth B can support. (10)
- (c) What is the use of *Bias* in floating point representation? Give the encoding of floating point infinity. (10)

## CSE 305

8. (a) What are the different methods to maintain coherence among all caches and global memory in a shared memory system? Describe briefly. (10)
- (b) Suppose we want to sum 64,000 numbers on a shared memory multiprocessor computer with uniform memory access time. Let's assume we have 64 (i) UMA processors, (ii) message passing processors. Write the corresponding C code. (20)
- (a) for each processor to sum their subset of numbers
- (b) to add the partial sums
- (c) Draw the organization of a message passing multiprocessor. (5)
- 



[www.prokoushol.com](http://www.prokoushol.com)

**SECTION – A**There are **FOUR** questions in this section. Answer any **THREE**.

1. (a) Describe SISD, SIMD, MIMD and MISD architectures with block diagrams. (10)  
(b) Consider a bus-based shared memory multiprocessor system. It is constructed using processors with speed of  $10^6$  fetches/sec, and a bus with a peak bandwidth of  $10^5$  fetches/sec. The caches are designed to support a hit rate of 90%. (4)
  - (i) What is the maximum number of processors that can be supported by this system?
  - (ii) What hit rate is needed to support a 20-processor system?  
(c) Compare the characteristics of typical CISC and RISC architectures. (8)  
(d) Show the time-space diagram for pipelining in a superscalar processor. Briefly describe the architecture of a typical superscalar RISC processor with necessary block diagrams. Mention the differences between a superscalar processor and a VLIW processor. (3+8+2)
  
2. (a) List out the key characteristics of computer systems. What do you mean by the principle of locality of reference? Discuss the role of the principle of locality of reference in designing memory hierarchy. (6+3+3)  
(b) Write short notes on (i) flash storage and (ii) holographic data storage. (4+4)  
(c) Answer the following questions for a memory system that uses 32-bit address at the byte level, and a cache that uses a 64-byte line size. (15)
  - (i) Assume a direct cache with a tag field in the address of 20 bits. Show the address format and determine the number of addressable units, the number of blocks in main memory, and the number of lines in the cache.
  - (ii) Assume an associative cache. Show the address format and determine the number of addressable units, the number of blocks of main memory, the number of lines in the cache and the size of the tag.
  - (iii) Assume a four-way set-associative cache with a tag field in the address of 9 bits. Show the address format and determine the number of addressable units, the number of blocks in main memory, the number of lines in a set, the number of sets in the cache, and the number of lines in the cache.

## CSE 305

3. (a) What are pipeline hazards? Explain a structural pipeline hazard with a time-space diagram. What type of pipeline hazards occurs in the following code segment? How can it be overcome? (10)

```

lw $t1, 0($t0)
lw $t2, 4($t0)
add $t3, $t1, $t2
sw $t3, 12($t0)
lw $t4, 8($t0)
add $t5, $t1, $t4
sw $t5, 16($t0)

```

- (b) Briefly describe different dynamic branch prediction techniques for dealing with control hazards. (10)
- (c) Perform the necessary modification of the single-cycle data path in Figure 1 for pipeline implementation by inserting pipeline registers and a forwarding unit. Describe the modification step by step and draw the final modified datapath with all necessary control signals. (15)

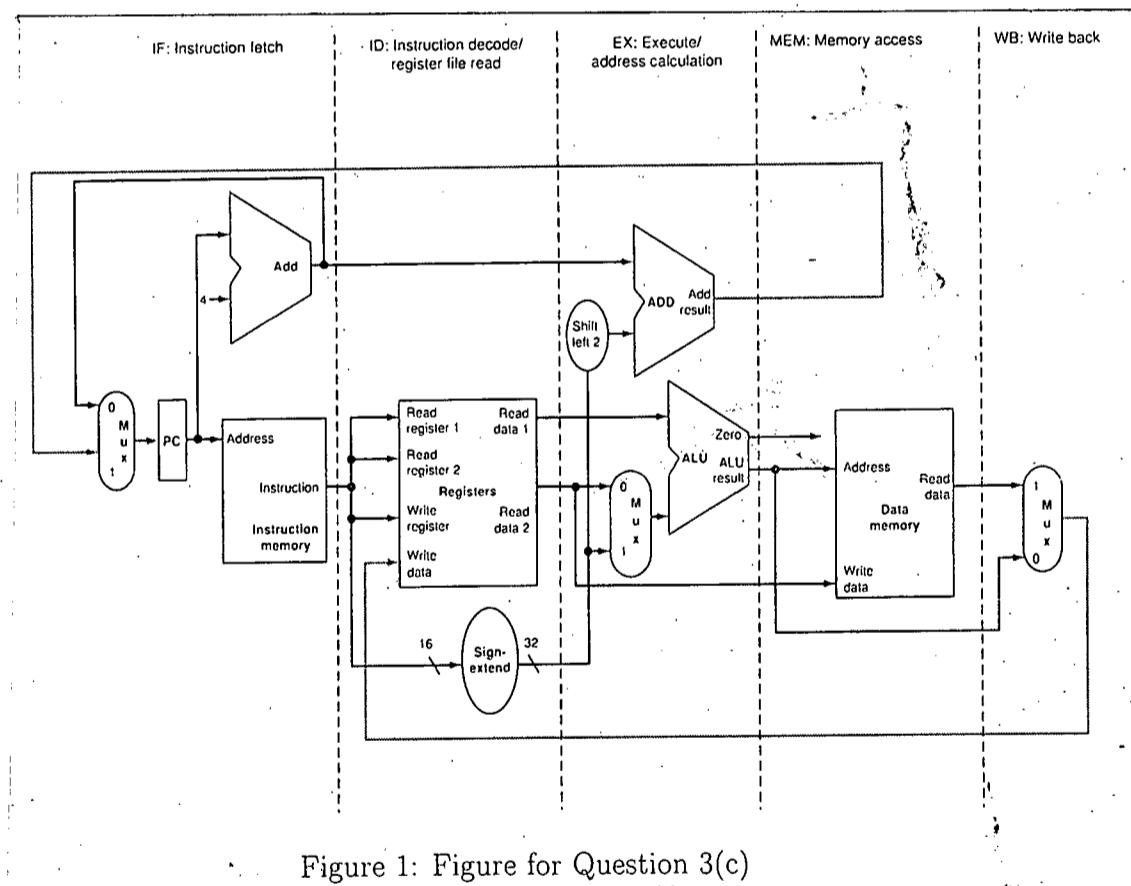


Figure 1: Figure for Question 3(c)

4. (a) Show the components of an I/O module using a block diagram and describe the functions of each component. (8)
- (b) Describe the operation of a vector interrupt structure with daisy chain interface for detecting and managing interrupting I/O devices. (10)

## CSE 305

### Contd ... Q. No. 4

- (c) Mention the advantages of DMA transfer over programmed I/O and interrupt-driven I/O. Briefly describe a DMA transfer structure with necessary diagrams. (4+8)
- (d) The access time of the cache  $M_1$  of a single cache memory system is 6ns/bit during hit and that of the main memory  $M_2$  is 900ns/bit during miss. Calculate the block transfer time and the access efficiency of the memory system at the hit ratio 0.8. (5)

### SECTION – B

There are **FOUR** questions in this section. Answer any **THREE**.

5. (a) "Computer architects have invented several great ideas in the last 60 years of computer design. These ideas are so powerful that they have lasted long after the first computer that used them, with newer architects demonstrating their admiration by imitating their predecessors". These ideas include: (20)
1. Design for Moore's Law
  2. Use abstraction to simplify design
  3. Make the common case faster
  4. Performance via parallelism
  5. Performance via pipelining
  6. Performance via prediction
  7. Hierarchy of memories
  8. Dependability via redundancy

You are given several design features of MIPS and other modern computing systems in the following. Now, please **identify** that which of the above great ideas are behind these design decisions.

1. In MIPS datapath, there is an adder which computes the value of **(PC+4)**, where PC means Program Counter. For branch addressing the MIPS address is actually relative to this **(PC+4)** rather than **(PC)**. This form of branch addressing is called PC-relative addressing.
2. In the past, most PCs (personal computers) and server systems used separate SRAM (static random access memory) chips for either their primary, secondary, or even tertiary caches. Today, all levels of caches are integrated onto the processor chip, so the market for separate SRAM chips has nearly evaporated.
3. A fast multiplication hardware, "unrolls the loop" by using 31 adders rather than using a single 32-bit adder 31 times.
4. Procedures or Functions allow programmers to concentrate on just one portion of the task at a time, parameters act as an interface between the procedure and the rest of the program and data, since they can pass values and return results.
5. Modern computers use a technique called prefetching. In prefetching, a block of data is brought into the cache before it is actually referenced. Many microprocessors use hardware prefetching to access the block needed in the future that may be difficult for software to notice.

## CSE 305

### Contd ... Q. No. 5

- (b) Assume a processor with 2 GHz clock rate is executing a program that requires the following instructions.

(15)

| Instructions                        | FP  | INT | L/S | BRANCH |
|-------------------------------------|-----|-----|-----|--------|
| Instruction Count ( $\times 10^6$ ) | 160 | 110 | 10  | 16     |
| CPI                                 | 2   | 1   | 5   | 2      |

- (i) Calculate the execution time of the program.
- (ii) By how much must we improve the CPI of FP instructions if we want the program to run two times faster?
- (iii) By how much must we improve the CPI of L/S instructions if we want the program to run two times faster?

6. (a) A simple recursive C function and its corresponding MIPS assembly code is given below. Note that, the later one is not complete. Please complete it with appropriate expressions.

(15)

```

reurseSum:
 $t0, $a0, 1
 beq $t0, $zero, Recuse
 add $v0, $zero, $zero
 jr []
Recuse:
 addi $sp, $sp, []
 sw $ra, []
 sw [], 0($sp)
 addi $a0, $a0, -1
 jal recurseSum
 lw [], 0($sp)
 lw $ra, []
 []
 add [], $a0, $v0
 jr []
}

int recurseSum(int n){
 if(n<1) return 0;
 else return n+recurseSum(n-1);
}

```

- (b) What support does MIPS instruction set architecture provide for synchronization?

(5)

- (c) A simple MIPS assembly code snippet and its corresponding MIPS machine code are given below. Note that, the later one is not complete. Please complete it with appropriate decimal values.

(15)

|       |                      |      |    |    |    |   |   |    |
|-------|----------------------|------|----|----|----|---|---|----|
| Loop: | sll \$t1, \$s2, 2    | 6000 | 0  | 0  | 18 | 9 |   | 0  |
|       | add \$t1, \$t1, \$s6 | 6004 | 0  |    |    |   | 0 | 32 |
|       | lw \$t0, 0(\$t1)     | 6008 | 35 |    |    |   | 0 |    |
|       | beq \$t0, \$s5, L1   | 6012 | 4  | 8  | 21 |   |   |    |
|       | j Exit               | 6016 |    |    |    |   |   |    |
| L1:   | addi \$s2, \$s2, 1   | 6020 | 8  | 18 | 18 |   | 1 |    |
|       | j Loop               | 6024 | 2  |    |    |   |   |    |
| Exit: |                      | 6028 |    |    |    |   |   |    |

## CSE 305

7. (a) Please draw the illustrations at the five MIPS addressing modes. (15)

(b) MIPS instruction set architecture supports *beq* and *bne* instructions, however, does not support other branching instructions like *blt*, *bge*, etc. What is the reason behind this design decision? How could you program these instructions (*blt*, *bge*, etc.) in MIPS instructions? (5+5)

(c) What is the problem in the following code snippet? How can you solve this problem? (5+5)

| Address | Instruction            |
|---------|------------------------|
| 005000  | beq, \$s0, \$s1, 30500 |

8. (a) The following table illustrates the various combinations of two operands for binary addition-subtraction. Please identify the overflow conditions in that table. Then, formulate the overflow boolean variable as a logical function of various input variables (You don't need to simplify the function). (8+4)

| O | S <sub>A</sub> | S <sub>B</sub> | S <sub>R</sub> | V |
|---|----------------|----------------|----------------|---|
| 0 | 0              | 0              |                |   |
| 0 | 0              | 1              |                |   |
| 0 | 1              | 0              |                |   |
| 0 | 1              | 1              |                |   |
| 1 | 0              | 0              |                |   |
| 1 | 0              | 1              |                |   |
| 1 | 1              | 0              |                |   |
| 1 | 1              | 1              |                |   |

Here,

$$\text{Operation, } O = \begin{cases} 0, & A + B \\ 1, & A - B \end{cases}$$

$$\text{Sign bit if Operand A, } S_A = \begin{cases} 0, & A \geq 0 \\ 1, & A < 0 \end{cases}$$

$$\text{Sign bit of Operand B, } S_B = \begin{cases} 0, & B \geq 0 \\ 1, & B < 0 \end{cases}$$

$$\text{Sign bit of Result R, } S_R = \begin{cases} 0, & R \geq 0 \\ 1, & R < 0 \end{cases}$$

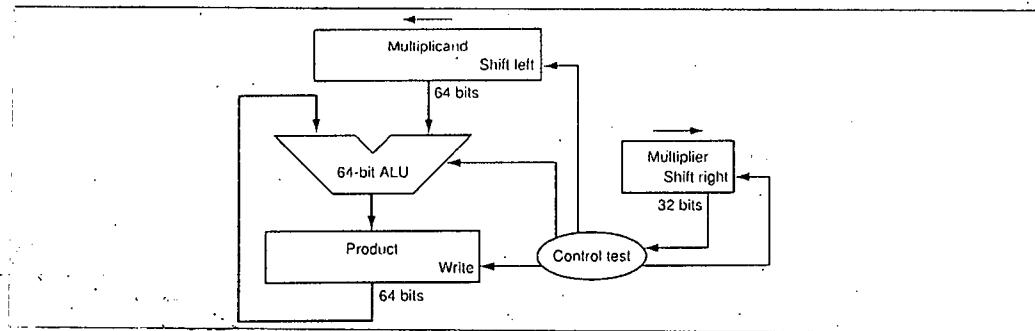
$$\text{Overflow, } V = \begin{cases} 0, & \text{Overflow occurred} \\ 1, & \text{No overflow possible} \end{cases}$$

## CSE 305

### Contd ... Q. No. 8

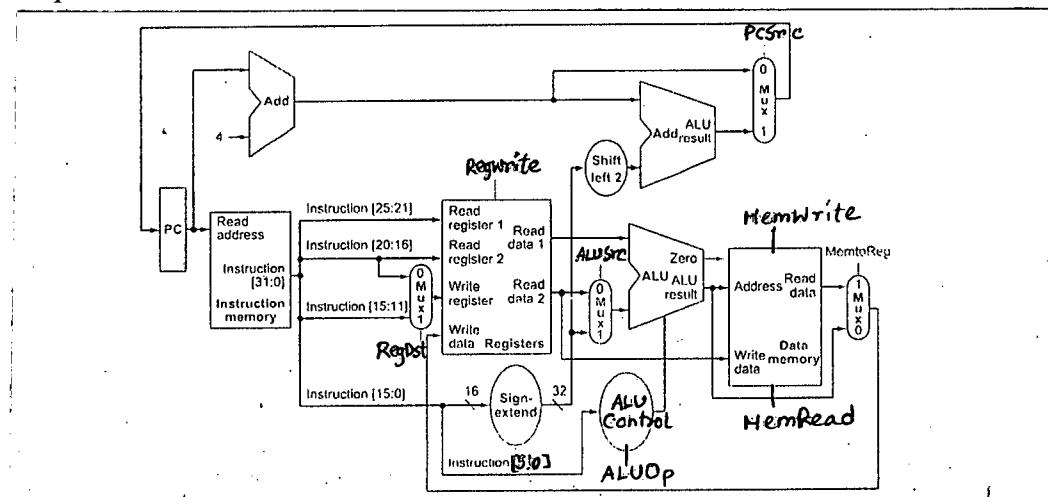
(b) In the following figure, a multiplication hardware is given. Now draw the flowchart of a the sequential version of multiplication algorithm for this hardware. Please note that, this simple hardware is not register-optimized, therefore, draw a register-optimized refined version of this hardware.

(8+5)



(c) A simple MIPS datapath and its respective control signals are illustrated below. Please design the control bits corresponding to the instruction "beq rs, rt, address" for this datapath.

(10)



| Control Signal | 0                                | 1                                                |
|----------------|----------------------------------|--------------------------------------------------|
| RegDst         | Write register address = $rt$    | Write register address = $rd$                    |
| RegWrite       | -                                | Write register                                   |
| ALUSrc         | ALU Second Operand = Read data 2 | ALU Second Operand = lower 16-bit of instruction |
| MemRead        | -                                | Read data from memory                            |
| MemWrite       | -                                | Write data into memory                           |
| MemtoReg       | Register Write Data from ALU     | Register Write Data from data memory             |
| PCSrc          | $PC=PC+4$                        | $PC=$ branch target                              |
| ALUOp1         | not dependent on function        | dependent on function                            |
| ALUOp0         | addition                         | subtraction                                      |

**SECTION - A**There are **FOUR** questions in this section. Answer any **THREE**.

1. (a) State the principle of temporal locality and the principle of spatial locality. (3+3)  
 (b) Assume a two-way set associative cache with four blocks. The current state of the cache is empty. Assuming an LRU replacement policy, how many hits does the address sequence 0, 2, 4, 0, 2, 4, 0, 2, 4 exhibit? Show the state of the cache after every address access. (9)  
 (c) State three conditions required for a cache/memory to be coherent. (9)  
 (d) What is the limitation of a processor that does not have a TLB but uses the virtual memory technique? (5)  
 (e) What is the size of the tag field for a direct-mapped cache with  $2^m$  blocks and the following configuration: 32-bit byte address and the block size:  $2^n$  words (Assume 1 word = 4 bytes). (6)
2. (a) Suppose a single shared memory processor has 20 GB of main memory. Five clustered computers each have 4 GB. The OS occupies 1 GB. How much more space is there for users with shared memory? (5)  
 (b) Determine the total network bandwidth and dissection bandwidth for ring and fully connected topologies. Assume that the number of processor is  $P$ . (8)  
 (c) Differentiate UMA shared memory multiprocessor from NUMA shared memory multiprocessor. (5)  
 (d) A virtual memory system has the following parameters: (7)  
     Virtual address (bits): 32  
     Physical DRAM installed: 4 GB  
     Page size: 8 KB  
     PTE size (byte): 4  
     How many page table entries are needed? How much physical memory is needed for storing the page table?  
 (e) Write down the procedure to handle a TLB page miss. (10)
3. (a) Write down the steps for the addition of two binary floating point numbers. (10)  
 (b) Show the IEEE 754 binary representation of the number  $-0.75_{10}$  in single precision. (10)  
 (c) Draw a block diagram of hardware organization and corresponding flowchart for the multiplication of two 32-bit binary unsigned integer numbers, where there is no separate multiplier register. (15)

## CSE 305

4. (a) Why does RAID 0 improve disk performance? (5)  
(b) Write down the steps of handling an interrupt. (12)  
(c) What happens if the interrupt enable bit of the Cause register is not set when handling an interrupt? What value could the interrupt mask value take to accomplish the same thing? (3+3)  
(d) What is "DMA stale data problem"? How can it occur in a system with caches? (12)

### SECTION – B

There are **FOUR** questions in this section. Answer any **THREE**.

5. (a) What is Moore's Law? What do you mean by response time and throughput? (6)  
(b) Consider two different implementations of the same instruction set architecture. There are four classes of instructions, A, B, C and D. The clock rate and CPI of each implementation are given in the following table: (15)

| Implementation | Clock Rate | CPI Class A | CPI Class B | CPI Class C | CPI Class D |
|----------------|------------|-------------|-------------|-------------|-------------|
| P1             | 3 GHz      | 1           | 2           | 3           | 4           |
| P2             | 2 GHz      | 2           | 2           | 2           | 2           |

Given a program with 200 instruction divided into classes as follows:

10% class A, 20% class B, 50% class C and 20% class D.

- (i) Find the clock cycles required in both cases.  
(ii) Which implementation is faster?  
(iii) What is the global CPI for each implementation?  
(c) What do you mean by Amdahl's law? Write down the equation. Suppose a sample program takes 100 seconds to complete. The division operation accounts for 80 seconds. How much improvement in division operations' performance can give 4x overall improvement? (8)  
(d) Consider the following information for two different computers. (6)

| Computer | Number of Instructions | Clock Rate | CPI |
|----------|------------------------|------------|-----|
| A        | 10 Billion             | 4 GHz      | 1   |
| B        | 8 Billion              | 4 GHz      | 1.1 |

- (i) Which one has the highest MIPS rating?  
(ii) Which one is faster?

6. (a) What are the four design principles used for MIPS instruction set design? What are the steps for starting a Java application? (8)  
(b) Write down the MIPS code and corresponding machine code for the following C code. (12)

$$A[3] = A[1] - A[2];$$

Consider the base address of A is in \$s0. The op-code for load word is 35, and for store word is 43. The (op-code, function-code) for addition is (0, 32), and for subtraction is (0, 34). Registers \$t0 to \$t7 are numbered from 8 to 15, and \$s0 to \$s7 are numbered from 16 to 23.

## CSE 305

### Contd ... Q. No. 6

(c) Write down the MIPS code for the following C procedure: (12)

```
int sum (int n)
{
 if (n <=0) return 0;
 else return n + sum (n - 1);
}
```

Consider n in \$a0 and result in \$v0.

(d) If branch target is too far to encode with 16-bit offset, how does the assembler rewrite the following code: (3)

```
beq $s0, $s1, L1
```

7. (a) Briefly describe the two instructions used for synchronization in MIPS. Write down the MIPS code to do the following atomically: (8)

A [0] = X

Consider the base address of A is in \$s1 and X is in \$s2.

(b) Describe MIPS addressing modes. (4)

(c) Consider the following set of instruction: (14)

```
add, sw, beq
```

Draw the complete single cycle datapath (not pipelined) and control that can execute the above set of instructions.

(d) What are the five states of MIPS pipeline? Consider the following information for load word (lw) instruction (9)

| Instruction | Instruction Fetch | Register read | ALU op | Memory access | Register write |
|-------------|-------------------|---------------|--------|---------------|----------------|
| lw          | 200 ps            | 100 ps        | 200 ps | 200 ps        | 100 ps         |

Consider the following program:

```
lw $1, 100($0)
lw $2, 200($0)
lw $3, 300($0)
```

What is the speedup for the pipelined datapath over the single cycle datapath for this program? What will be the speedup if we add 1,000,000 more load word (lw) instructions?

8. (a) Draw the complete MIPS pipelined datapath and control for load word (lw) instruction with pipelined registers. (15)

(b) Consider the following MIPS code: (7)

```
add $s0, $t0, $t1
sub $t2, $s0, $t3
lw $s1, 20($t1)
sub $t4, $s1, $t2
```

How many data hazards are there? Describe the techniques to eliminate data hazards in the above code.

## CSE 305

### Contd ... Q. No. 8

(c) What do you mean by data hazard for branches? Give examples with MIPS code for each of the following cases of data hazards for branches: (9)

- (i) Needs no stall to resolve
- (ii) Needs one cycle stall to resolve
- (iii) Needs two cycle stalls to resolve

(d) Consider the following code: (4)

outer:

// some code

inner:

// some code

beq \$t0, \$zero, inner

// some code

beq \$t1, \$zero, outer

What is the problem with the above code if we use 1 bit branch predictor? How you can solve that? Explain

---

CE 312  
12 + 3

L-3/T-1/CSE

Date : 10/05/2014

BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY, DHAKA

L-3/T-1 B. Sc. Engineering Examinations 2012-2013

Sub : **CSE 305** (Computer Architecture)

Full Marks : 210

Time : 3 Hours

The figures in the margin indicate full marks.

USE SEPARATE SCRIPTS FOR EACH SECTION

---

### SECTION - A

There are **FOUR** questions in this section. Answer any **THREE**.

1. (a) What is a jump address table? Discuss with an appropriate example how Case/Switch Statements can be implemented with a jump address table. (15)  
(b) How you can create a 32 bit constant in MIPS Assembly Language? (5)  
(c) Consider the following MIPS assembly language code and assume that the code fragment is loaded at location 80000 in the memory. What should be the values of the immediate fields of *bne* instruction and *j* instruction? (8)

Loop:  
s11 \$t1,\$s3,2  
add \$t1,\$t2,\$s6  
Lw \$t0, o(\$t1)  
bnr \$t0,\$s5, Exit  
addi \$s3, \$s3, 1  
j Loop

Exit:

- (d) Explain why the assembler might have problems directly implementing the branch instruction in the following code fragment. How can this problem be fixed? (7)

Here: *beq \$s0, \$s2, there*

...

there: *add \$s0, \$s0, \$s0.*

2. Consider the single cycle datapath in Figure 2 which is constructed for *add, sub, and, or, lw, sw, beq*. Now, implement the following instructions in the datapath: *bne, j, jr, jal*. You need to clearly explain your implementation and mention the values of the control lines (including any additional control lines you have introduced) for the new instructions. (35)

[You may modify Figure 2 (separate sheet) and attach the update figure with your answer script after having it signed by the respected invigilator.]

## CSE 305

3. (a) Present and explain a microprogram for the control unit of the multicycle implementation illustrated in Figure 3. (17)
- (b) Modify the implementation of Figure 3 so that it can handle the **Undefined Instruction Exception** and the **Overflow Exception**. (8)
- [You may modify Figure 3 (separate sheet) and attach the updated figure with your answer script after having it signed by the respected invigilator.]
- (c) Which stage should you detect the **overflow exception** in so as to ensure that the offending instruction does not change the content of the destination register? Justify your answer. (5)
- (d) Your friend claims that the *PCWriteCond* signal can be safely replaced by *PCSource[0]*. Do you agree with him? Justify your answer. (5)
4. (a) Discuss how you can detect data hazards and solve those with the help of a forwarding unit. (17)
- (b) Present a scenario when you cannot solve a data hazard with the help of a forwarding unit. Suggest what to do in this case. (13)
- (c) What is a branch hazard? (5)

### SECTION – B

There are **FOUR** questions in this section. Answer any **THREE**.

5. (a) What is the average time to read or write a 512-byte sector for a typical disk rotating at 15,000 RPM? The seek time is 4 ms, the transfer rate is 100 MB/sec, and the controller overhead is 0.2 ms. Assume that the disk is idle so that there is no waiting time. (10)
- (b) What is the purpose of using RAID organization? Briefly describe the concept of different RAID levels with diagrams. (2+18)
- (c) Differentiate between NAND and NOR flash memories. (5)
6. (a) Consider a computer running program with CPU times shown in the following table: (3+3+3)

| FP<br>intr. | INT<br>instr. | Load/Store<br>instr. | Branch<br>intr. | Total<br>Time |
|-------------|---------------|----------------------|-----------------|---------------|
| 50sec       | 80sec         | 50sec                | 30sec           | 210sec        |

- (i) By how much (in %) is the total time reduced if the time for FP operations is reduced by 20%?
- (ii) Assume that the time for INT operations is improved. By how much (in %) is the time for INT operations is reduced if the total time is reduced by 20%?
- (iii) Can the total time be reduced by 20% by reducing only the time for branch instructions?

## CSE 305

### Contd ... Q. No. 6

(b) What is the disadvantages of using virtually addressed cache? (6)

(c) "The TLB acts as a cache of the page table for the entries that map to physical pages only" – Do you agree or disagree? Explain. (7)

(d) A program accesses 2 cache blocks, one that begins at memory address 0x1000 and another one that begins at memory address 0x2000. Memory accesses alternate between these two blocks and each block is accessed 100 times.

If the program is run on a system with a 1 KB direct-mapped cache with 32 blocks, how many cache misses will occur? (13)

How many of these misses will be compulsory and conflict misses. Mention the reason behind classifying a miss as a compulsory or a conflict miss. Note that there will be no capacity miss as the total amount of data accessed by the program is less than the capacity of the cache.

7. (a) State the advantages of using write back cache. (6)

(b) If a cache has a capacity of 16 KB and a line length (block size) of 128 bytes, how many sets does the cache have if it is 2-way, 4-way and 8-way set associative? (9)

(c) Consider a processor with following parameters: (12)

Base CPI, no memory stalls: 2.0

Processor speed: 1 GHz

Main memory access time: 100 ns

First-level cache miss rate per instruction: 4%

Second level cache, direct-mapped speed: 10 cycles

Global miss rate with second-level cache, direct mapped: 4%

Second level cache, eight-way set associative speed: 20 cycles

Global miss rate with second-level cache, eight-way set associative: 1.6%

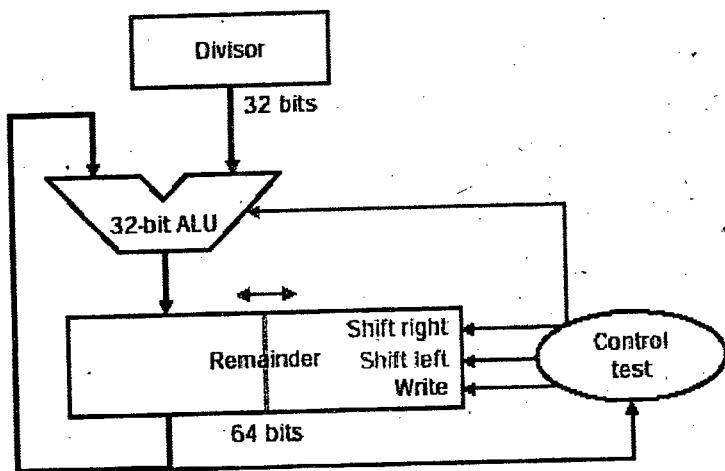
Calculate the CPI for the processor using (i) only a first-level cache, (ii) both a first-level cache and a second-level direct-mapped cache, (iii) both a first level cache and a second-level eight-way set associative cache.

(d) Briefly discuss interleaved memory organization. (4)

(e) Differentiate between fine-grain and coarse-grain multithreading. (4)

CSE 305

8. (a) What is the purpose of using biased notation for representing exponents of floating point numbers? (5)
- (b) Draw the flowchart showing the execution steps of the division operation for the division hardware organization given below: (12)



- (c) Describe the steps of DMA transfer. (8)
- (d) Discuss crossbar network and Omega network with diagrams. (10)

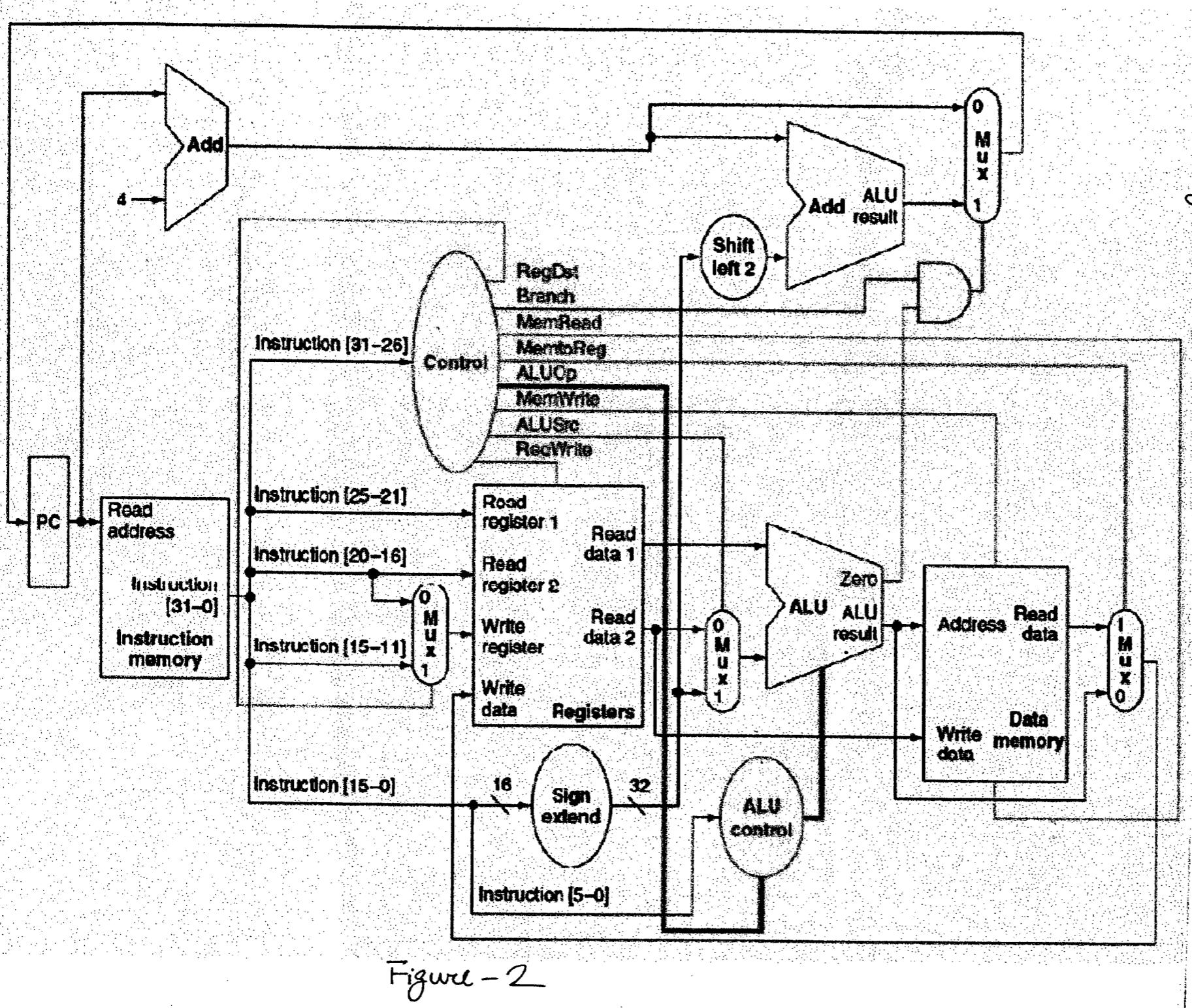


Figure - 2

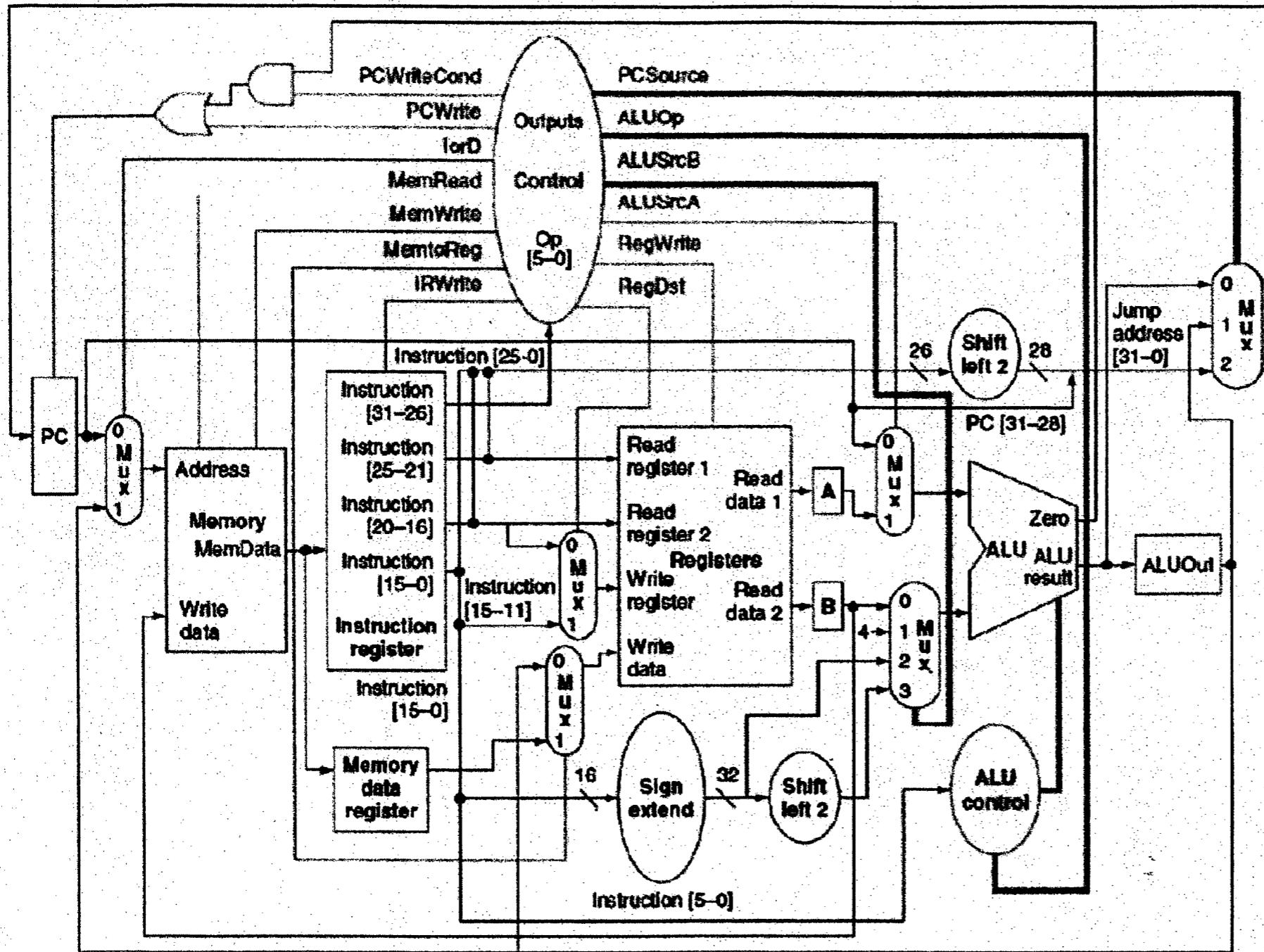


Figure - 3

**SECTION - A**

There are **FOUR** questions in this Section. Answer any **THREE**.

1. (a) What are the components of a computer? Mention the task of each of these components. (7.5)
- (b) Briefly describe ISA classes with examples and block diagrams. (10)
- (c) Suppose we have two implementations of the same instruction set architecture Computer A has a clock cycle time of 250 ps and a CPI of 2.0 for some program, and computer B has a clock cycle time of 500 ps and a CPI of 1.2 for the same program. Which computer is faster for this program and by how much? (Show all steps of calculations.) (7)
- (d) Suppose a program runs in 10 seconds and the program spends 50% of its time executing floating point instructions. We improve the floating point unit by a factor of five. How big is the speedup? (6)
- (e) What are the problems with measuring performance in terms of MIPS (million instructions per second)? (4.5)
  
2. (a) What are the MIPS design principles? Mention one use of each of these design principles in MIPS ISA. (4+4)
- (b) Which instructions and registers are used in procedure call in MIPS hardware? What are the purpose of using these registers and instructions? (7.5)
- (c) If branch target is too far to encode with 16-bit offset, how does the assembler rewrite the following code: BEQ \$s0, \$s1, L1? (3.5)
- (d) Describe MIPS addressing modes. (10)
- (e) Give three examples of complexity of IA-32. (6)
  
3. (a) Consider the following set of instructions: ADD, SW, JUMP. Draw the single cycle datapath and control that can execute the above set of instructions. Minimize the number of hardware used as much as possible, i.e., there should be no redundant hardware in your design. (Note that explanation is not required.) (15)
- (b) Write the purpose of using ALU unit for R-type, load/store, and branch instructions. (6)
- (c) Briefly describe the properties of hard-wired and microprogrammed control unit design. (6)

## CSE 305

### Contd ... Q. No. 3

(d) Consider the following MIPS instructions:

- (i) LW \$6, 36(\$1)
- (ii) ADD \$5, \$5, \$5

What do these instructions perform in EX and MEM stages. Assume that the pipelined datapath has 5 stages: IF (Instruction Fetch), ID (Instruction decode and register file read), EX (Execution op address calculation), MEM (Data memory access), and WB (Write Back).

IF / ID

4. (a) What are the values stored in IF/D and ID/EX pipeline registers? (6)

(b) What is a pipeline hazard? What are the different types of pipeline hazards? (2+6)

(c) Give an example of double data hazard. (3)

(d) Consider the following sequence of instructions: (4)

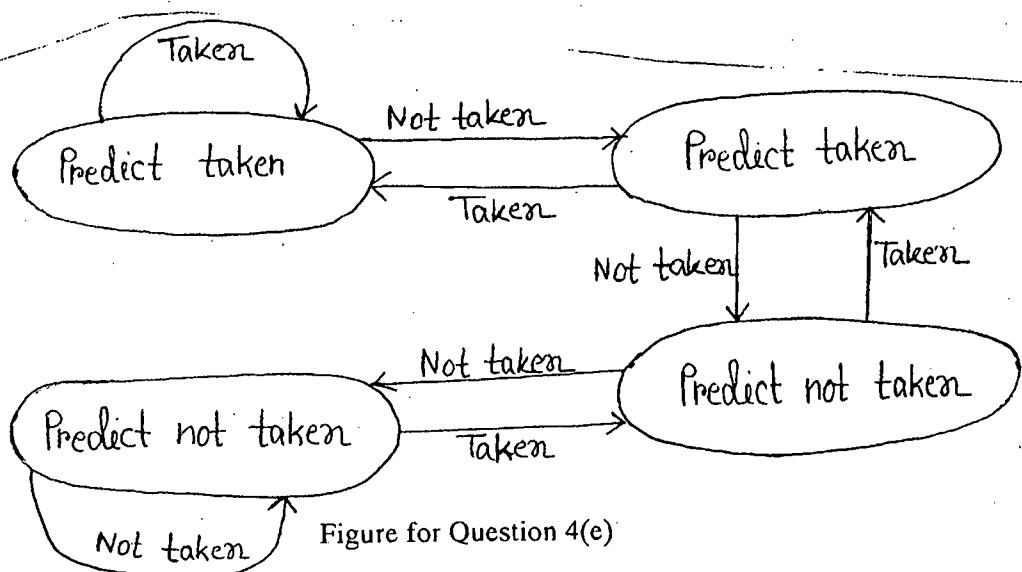
LW \$1, 40(\$6)

ADD \$6, \$2, \$2

SW \$6, 50(\$1)

Assume that there is no forwarding in this pipelined processor. Estimate data hazards with "nop" instruction.

(e) Consider the following repeating pattern of branch outcome: T, T, NT, T. What is the accuracy of the two-bit predictor for this sequence of branch outcomes? What is the accuracy of the two-bit predictor in steady state if this pattern is repeated forever? Assume that the 2-bit predictor starts off in the bottom left state (predict not taken) of the figure given below. (7)



(f) Why does the MIPS implementation use EPC and Cause registers for processing an exception? What is an imprecise exception? (4+3)

## CSE 305

### SECTION - B

There are **FOUR** questions in this Section. Answer any **THREE**.

5. (a) Draw the flow diagram of floating point addition. Also write down the condition of overflow or underflow during a floating point addition. (8+2)
- (b) What are the 'direct mapped', 'set associative', and 'fully associative' cache schemes? (6)
- (c) Assume there are three small coaches, each consisting of 4 one-word blocks. One each is fully associative, another one is 2-way set associative, and the third one is direct mapped. Find the number of misses for each cache organization given the following sequence of block addresses: 0, 4, 0, 2, 6, 8. (15)
- (d) What is a split cache? How is it different from the multilevel cache system? (2+2)  
*Split*
6. (a) Draw diagrams of the first and refined versions of the multiplication hardware. Discuss the improvement of the refined version over the first version. (5+5+5)
- (b) Assume the miss rate of an instruction cache is 2% and the miss rate of a data cache is 4%. If a processor has a CPI of 6 without any memory stalls, determine how much faster a processor would run with a perfect cache that never misses. Assume a main memory access time of 100 ns (including all the miss handling) and the processor clock rate is 4 GHz. (10)
- (c) Write down the steps to handle a cache miss. (5)
- (d) Discuss the components of disk data access time. (5)
7. (a) (i) What do you understand by the 'virtual memory' scheme? (3)  
(ii) Explain the purpose of a 'page table' and a 'page table register' in this scheme. (4+2)  
(iii) Draw the mapping from a 32-bit virtual address to a 30-bit physical address using a page table. (6)  
(iv) With a 32-bit virtual address, 4 kB pages and 4 bytes per page table entry, compute the total size of a page table. (5)
- (b) Draw the state diagram of a simple cache controller. (10)
- (c) What are the 'synchronous' and the 'asynchronous' bus system? (5)
8. (a) Assuming a cache of 16k blocks, a 4-word block size and a 32 bit address, find the total number of tag bits for a 4-way set associative cache. Also find the total size of the cache. [byte offset = 2 bits, 1 word = 32 bits, and  $1k = 2^{10}$ ]. (10)
- (b) Briefly describe the 'shared memory multiprocessor' and the 'message passing multiprocessor' systems. (6+6)

**CSE 305**

**Contd ... Q. No. 8**

(c) Suppose a single shared memory processor has 20 GB of main memory. Five clustered computers occupy 4 GB each and the OS occupies 2 GB. How much more space is there for users with shared memory? (8)

(d) What are processor memory bus and back-plane bus? (5)

---