# United International University
## Department of Computer Science and Engineering

CSE-3313: Computer Architecture

Final Examination: Spring 2025

Total Marks: 40 Time: 2 hours

Any examinee found adopting unfair means will be expelled from the trimester / program as per UIU disciplinary rules.

---

*Answer all questions. Numbers to the right of the questions denote their marks.*

1. (a) You have to modify the block diagram for the single-cycle data path and write the control unit values for this instruction: 'powerload $s0, $s1, $s2'. The job of the given instruction is to read the value of the source registers (rs, rt) and calculate twice the value of each of them. Assuming variable a and b are in register rs, and rt, the task is to calculate the values of a multiplied by 2 and b multiplied by 2. After that, the calculated values will be added first to get an address. We need to read the value of that address and write it to the destination register rd. The machine code format for the given instruction is given below in table 1, where rs, rt, and rd will contain the index numbers of $s1, $s2, and $s0 registers, respectively. [5]

| op | rs | rt | rd | shamt | func |
|---|---|---|---|---|---|
| 6 bits | 5 bits | 5 bits | 5 bits | 5 bits | 6 bits |

Table 1: Machine code format of instruction powerload

(b) Modify the block diagram for a single-cycle data path so that it can execute the following j-type instruction 'aj 1024'. The operation of this instruction is to jump to the given address (1024) only if the value stored at that memory address is greater than the address of the next instruction. Note that the Program Counter (PC) contains the address of the current instruction. Each instruction is of 4 bytes. Write down the control unit values as well. [5]

2. (a) Consider a processor that goes through the following six stages while executing an instruction:

| | |
|---|---|
| Instruction Fetch (IF) | 120 ns |
| Instruction Decode (ID) | 150 ns |
| Execution (EXE) | 180 ns |
| Memory Read (MEMR) | 190 ns |
| Memory Write (MEMW) | 200 ns |
| Write Back (WB) | 170 ns |

Table 2: Six stages pipeline

i. Using the pipeline method, calculate the total execution time for the following code. Provide a timing diagram, and reduce the number of cycles wherever possible. [5]

```
addi $t2, $zero, 100
lw $t1, 32($s0)
add $t2, $t2, $t1
sw $t1, 16($s0)
ori $t4, $t2, 10
```

ii. Now for the code snippet given in 2i, please find out an optimal solution (By re-ordering the code) where stalls will be minimum. Find out the total time. No need to show a timing diagram. [2]

iii. Calculate the total execution time for the given code below using the single-cycle method. Consider the pipeline stages and compute the time required for each instruction. Also, find the final value of $s3. [5]

```
addi $s3, $zero, 0
addi $s0, $zero, 0
addi $s1, $zero, 12
addi $s2, $zero, 4
LOOP:
sll $t0, $s0, 2
lw $t1, 0($t0)
```

```
beq $t0, $s1, END
add $t1, $t1, $s2
add $s3, $s3, $t0
addi $s1, $s1, 2
addi $s0, $s0, 2
j LOOP
END:
sw $s3, 40($s1)
```

iv. What is Control Hazard in pipelined processors? What problems does it cause, and what techniques can be used to resolve it? Explain with diagram. [3]

3. Given a direct-mapped cache with one word per block in Table-3. Where one word is 2 Bytes.

| Index | V | D | Tag | Data |
|---|---|---|---|---|
| 0 | 0 | 0 | ? | - |
| 1 | 1 | 1 | 100 | $D_b$ |
| 2 | 0 | 1 | ? | - |
| 3 | 0 | 0 | ? | - |
| 4 | 0 | 0 | ? | - |
| 5 | 1 | 0 | 011 | $D_c$ |
| 6 | 0 | 0 | ? | - |
| 7 | 1 | 0 | 000 | $D_y$ |
| 8 | 0 | 0 | ? | - |
| 9 | 0 | 0 | ? | - |
| 10 | 0 | 0 | ? | - |
| 11 | 0 | 0 | ? | - |
| 12 | 1 | 1 | 000 | $D_x$ |
| 13 | 1 | 0 | 111 | $D_a$ |
| 14 | 0 | 0 | ? | - |
| 15 | 0 | 0 | ? | - |

Table 3: Direct mapped table

(a) Calculate the actual size of the direct mapped cache. [ 3 ]

(b) Calculate the physical memory size. [ 2 ]

(c) If the CPU generates physical address requests sequentially as follows: 240, 27, 135, 241, 25. Determine the cache miss ratio. Memory is byte addressable. [ 5 ]

(d) Find the physical address if CPU wants to access the following data: $D_a$,$D_b$,$D_c$,$D_x$,$D_y$ [ 5 ]