Premier University, Department of CSE
Fall 2024, 6th Semester, Assignment, February 2025
Course Title: Computer Organization & Architecture
Course Code: CSE 337 Course Outcome: CO4, Total Marks: 10

Problem: Efficient CPU Design with Multiple Addressing Modes

You are tasked with designing an **efficient CPU architecture** that optimizes instruction execution while effectively utilizing different **CPU organizations and addressing modes**. The goal is to maximize performance, minimize memory access latency, and improve computational efficiency while considering trade-offs in instruction format and control unit complexity.

Objectives: Design a CPU architecture that integrates efficient data processing with optimized memory access to ensure high performance. Implement a flexible instruction format that supports zero, one, and two-addressing modes, balancing performance with hardware complexity. Develop a control unit capable of managing instruction decoding, execution flow, and addressing mode selection. Analyze the trade-offs between different CPU organizations (Accumulator-based, Register-based, Stack-based) and justify the design choices to optimize overall system performance and hardware efficiency.

Investigation: Investigate CPU architecture, focusing on instruction formats, addressing modes, and control unit operations. Consider performance factors like reducing memory access and optimizing execution flow.

Evaluation: Justify your design by explaining how the architecture improves efficiency compared to traditional models, balancing instruction size with execution speed. Discuss how the control unit manages instruction decoding and addressing modes effectively.

Design: Your solution should include the following components

Choose a CPU structure (Accumulator, Register, or Stack-based) and define instruction flow through the CPU pipeline.

Design an instruction format supporting zero, one, two, and three-addressing modes while optimizing encoding to minimize fetch cycles and delays.

Integrate addressing modes (Immediate, Direct, Indirect, Indexed), justifying trade-offs in speed, memory efficiency, and complexity.

Implement a control unit to manage instruction decoding, execution flow, and address resolution.

Propose techniques to minimize execution time, reduce memory stalls, and optimize control flow, addressing pipeline bottlenecks.

Deliverables: Submit a report with diagrams of CPU organization, instruction format, addressing modes, and control unit interactions. Provide a comparative analysis of your design against traditional models, highlighting efficiency improvements and trade-offs in architecture and instruction formats.

Complex Problem-Solving Questions:

- How does your CPU design efficiently handle instruction execution across different addressing modes?
- What trade-offs did you consider when balancing **performance, instruction size, and control unit complexity**?
- How does your control unit manage **instruction decoding and execution flow** while supporting multiple addressing modes?
- What performance challenges arise in your design, and how have you addressed them?
- Does your design align with industry standards for modern CPU architecture?

Rubrics for Assignment Marking:

Task	Criteria	Good (4-5)	Moderate (2-3)	Poor (1)
i.	Problem solution	Properly or near appropriately reasoned solution	Appropriate solution for some cases	Inappropriate or no solution
ii.	Problem analysis	In-depth analysis	Shallow analysis	Incomplete analysis