# King Fahd University of Petroleum &Minerals

Information and Computer Science Department
ICS 233: Computer Architecture and Assembly Language
2006-2007 (062)

**Instructor:** Muhamed Mudawar
**Class/Laboratory Schedule:** 3 50-minute lectures and 1 3-hour lab per week. (3-3-4)
**Designation**: Required Course

**Catalog Description**

Machine organization; Assembly language: addressing, stacks, argument passing, arithmetic operations, decisions, modularization; Input/Output Operations and Interrupts; Memory Hierarchy and Cache memory; Pipeline Design Techniques; Super-scalar architecture; Parallel Architectures.

**Prerequisites(s)**

COE 202, ICS 201

1. Fundamental programming constructs
2. Digital logic and digital systems
3. Machine level representation

**Textbook(s) and or other required material**

1. Computer Organization and Design: The Hardware / Software Interface. Third Edition. David A. Patterson and John L. Hennessy. Morgan Kaufmann, 2005.
2. MIPS Assembly Language Programming. Robert L. Britton. Pearson Prentice Hall, 2004.

**Course objectives:**

- Expose students to the trade-off analysis in designing various aspects of Computer Architecture, which includes Instruction Set Design, Control Unit Design, Instruction Level Pipeline, Cache and Memory Hierarchy.
- Provide students with intermediate level experience in assembly language programming.

**Relationship of Course to program outcomes**

After completion of this course, the student shall be able to:

Outcome 1: Analyze, write, and test MIPS assembly language programs. *[Program Outcome 3]*

Outcome 2: Describe the organization and operation of integer and floating-point arithmetic units. *[Program Outcome 1]*

Outcome 3: Determine CPU and cache memory performance and compute the speedup related to given enhancements. *[Program Outcome ?]*

Outcome 4: Design the datapath and control of a processor. *[Program Outcome 3]*

Outcome 5: Use simulator tools in the lab and in projects. *[Program Outcome ?]*

**Topics Covered**

- Data Representation
- Instruction Set Architecture
- MIPS Assembly Language Programming
- Procedures and the Runtime Stack
- Interrupts
- Integer and Floating-point Arithmetic and ALU design
- MIPS floating-point coprocessor and instructions
- CPU Performance
- Single-Cycle Datapath and Control Design
- Pipelined Datapath and Control
- Memory System Design and Cache Memory

**Laboratory Projects**

| Topic | Number of Weeks |
|---|---|
| Introduction and Orientation. | 1 |
| Introduction to assembly language programming | 1 |
| Algorithm development in pseudo-code | 1 |
| Arrays and addressing modes, flow control instructions | 1 |
| Input/output and the SYSCALL instruction | 1 |
| Functions, parameters, and the runtime stack | 1 |
| Integer multiplication/division, floating-point instructions | 1 |
| Introduction to Logisim, designing and simulating a register file | 1 |
| Datapath design and simulation with Logisim | 1 |
| Control unit implementation and simulation | 1 |
| Pipelined datapath design and simulation | 1 |
| Pipeline hazards | 1 |
| Cache memory | 1 |

**Contribution of course to Meeting the professional component**

The students will have basic and fundamental exposure to computer hardware and assembly programming that enhances their programming skills, especially directly with devices and hardware.

Student projects emphasize the design, implementation, simulation, and testing the datapath and control of a simple MIPS-like CPU.

## Estimated Curriculum Category Content (Semester hours)

| Area | Core | Advanced | Area | Core | Advanced |
|---|---|---|---|---|---|
| **Algorithms** | | | **Data Structures** | | |
| **Software Design** | | | **Prog. Languages** | | |
| **Comp. Arch.** | 4 | | | | |

## Oral and Written Communications

Students are required to submit written reports for their projects.

## Social and Ethical Issues

Not applicable

## Theoretical Content

Not applicable

## Problem Analysis

Not applicable

## Solution Design

Not applicable