KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
COLLEGE OF COMPUTER SCIENCES & ENGINEERING

COMPUTER ENGINEERING DEPARTMENT

COE 205 Computer Organization & Assembly Language
Syllabus - Term 033

Catalog Description

Introduction to computer organization. Signed and unsigned number representation, character representation, ASCII codes. Assembly language programming, instruction format and types, memory and I/O instructions, dataflow, arithmetic, and flow control instructions, addressing modes, stack operations, and interrupts. Datapath and control unit design. RTL, microprogramming, and hardwired control. Practice of assembly language programming.
Prerequisite: COE 200 and ICS 201

Instructor Dr. Aiman H. El-Maleh. Room: 22/318 Phone: 2811
Email: aimane@ccse.kfupm.edu.sa

Office Hours SUMTW 11:30-12:30 PM (and by appointment)

Course Objectives

1. Master assembly language programming in general and for the x8086 processor family in particular. The student should be able to analyze, debug, test and understand assembly language programs and implement algorithms in assembly language.

2. Understand the basic components in a CPU design. The student should be able to design a small CPU starting from the instruction set.

Course Learning Outcomes

The student should be able to:

1. Explain how an instruction is fetched from memory and executed.

2. Explain the relationship between the representation of machine level operation at the binary level and their representation by a symbolic assembler.

3. Explain why a designer adopted a given different instruction formats, such as the number of addresses per instruction and variable length vs. fixed length formats.

4. Write small programs and fragments of assembly language code to demonstrate an understanding of machine level operations.

5. Implement some fundamental high-level programming constructs at the machine-language level.

6. Use computer simulation packages to investigate assembly language programming.

7. Compare alternative implementations of datapaths.

8. Discuss the generation of control signals using hardwired and microprogrammed implementations.

Text Books & References:

- *Introduction to Assembly Language Programming: From 8086 to Pentium Processors*, Sivarama P. Dandamudi, et al., Springer Verlag, 1998. (ISBN: 0387985301).

- *Computer Systems Design and Architecture*, Vincent Heuring, Harry F. Jordan, Miles Murdocca, Addison Wesley 1997. (ISBN 0-8053-4330-X).

- *Assembly Language Programming and Organization of the IBM PC*, Ytha Yu and Charles Marut, McGraw Hill, 1992. (ISBN: 0-07-072692-2).

Grading Policy

Laboratory	20%	
Programming Assignments	10%	
Quizzes	10%	
Exam I	15%	(July 15, 1:30PM)
Exam II	20%	
Final	25%	

- Assignments are to be submitted in class in the specified due date.
- Late assignments will be accepted but will be penalized 10% per each late day.

Course Topics

1. ***Introduction and Information Representation.*** **6 lectures**
   Introduction to computer organization. Instruction Set Architecture. Computer Components. Fetch-Execute cycle. Signed number representation ranges. Overflow.

2. ***Assembly Language Concepts.*** **6 lectures**
   Assembly language format. Directives vs. instructions. Constants and variables. I/O. INT 21H. Addressing modes.

3. ***8086 Assembly Language Programming.*** **17 lectures**
   Register set. Memory segmentation. MOV instructions. Arithmetic instructions and flags (ADD, ADC, SUB, SBB, INC, DEC, MUL, IMUL, DIV, IDIV). Compare, Jump and loop (CMP, JMP, Cond. jumps, LOOP). Logic, shift and rotate. Stack operations. Subprograms. Macros. I/O (IN, OUT). String instructions . Interrupts and interrupt processing, INT and IRET.

4. ***Memory System Design.*** **4 lectures**
   Main memory, SRAM, DRAM. External memory, magnetic and optical disks. Bus system.

5. ***CPU Design.*** **12 lectures**
   Register transfer. Data-path design. 1-bus, 2-bus and 3-bus CPU organization. Fetch and execute phases of instruction processing. Performance consideration. Control steps. CPU-Memory interface circuit. Hardwired control unit design. Microprogramming. Horizontal and Vertical microprogramming. Microprogrammed control unit design.