King Saud University  
College of Computer and Information Sciences  
Department of Computer Science  
CSC220  Computer Organization  
(2-2-1) - Required Course

Current Instructor: Dr. Mohammed Barbar, Dr Rachid S Sammouda  
Course Coordinator: Dr. Mohammed Barbar

Catalogue Description

The course introduces basic digital logic design techniques and integrates the topics of generic assembly language programming, computer organization, and computer design. The objectives of this course are to: help students learn the fundamental elements of computer architecture from a functional, hardware perspective; foster an appreciation of organizational models and design decisions that determine the overall performance, capabilities, and limitations of a computer system; and help students understand the interdependencies among assembly languages, computer organization, and design. Topics include Introduction to basic computer organization and how the computer works; DeMorgan’s Law, simplifying circuits using Karnaugh maps, Instruction code, Computer registers, Instruction set, Timing and control; Register Transfer Language HDL (Hardware Description Language) and Micro-operations; Computer Arithmetic Logic Unit Design. -Hardwired control unit, instruction set, introduction to addressing modes; Central Processing Unit Design, Register organization, Instruction format, Addressing modes.

Pre-requisites for this course (if any)

CSC111  computer Programming I

Textbook :


Course Objectives:

The course aims to give the students:

- Design and understanding of the different basic components of a computer system;
- a theoretical and practical understanding of how the different components inside the computer system functions
- a theoretical and practical understanding of the register transfer language
- a basic understanding of the importance of control units.
- a theoretical and practical understanding of the arithmetic logic unit.
- a help to understand the interdependencies among assembly languages, computer organization, and design

Course Learning Outcomes

Outcomes will be assessed using classroom performance (lab if any), graded homework assignments (lecture and lab), quizzes (lecture and lab), graded lab exercises, course project (lab), and midterm and final examinations (lecture and lab).

Upon completing CSC220, students should have the following capabilities:

1. Describe all common components of computer systems
2. Write instructions in generic assembly language and understand the interdependencies among assembly languages, computer organization, and design
3. Design an Arithmetic and Logic Unit (ALU)
4. Understand the functionality of different CPU components.
Detailed Outcomes Assessment
Outcomes will be assessed using classroom performance (lab), graded homework assignments (lecture and lab), quizzes (lecture and lab), graded lab exercises, course project (lab), and midterm and final examinations (lecture and lab).

Expected Performance Criteria
The student is expected to implement several assigned programming tasks, and to pass three written examinations.

Topics:
- Logic gates, Boolean Algebra, Map Simplification
- Digital components: Decoders, Multiplexers, Register with parallel load, Shift Registers, Binary Counters
- Data Representation, Number Systems, Complements, Fixed Point Representation. Floating Point Representation
- Register transfer language, Bus and Memory Transfer
- Arithmetic Microoperations, Logic Microoperation, Shift Microoperations.
- ALU design
- Addressing modes, Control of flow and micro-operations instruction set.

Schedule:
14 weeks of 2 lectures, each is one-hour and one-hour tutorial.

Contribution to Professional Component:
CSC220 provides the computer organization principles needed to understand a computer’s operation and prepares the student for future courses such as CSC227. This course covers the topics that allow the student to understand the bridge between the hardware and the software in a computer system.

Engineering Topics: 80% Technical Writing: 20%

Relationship of course to ABET Criteria:
Criterion 2 - Program Educational Objectives:
This course allows the student to gain the necessary skill and experience to contribute to either a research or development project involving the design concepts of modern uni-processor computers.

Criterion 3 – Student Outcomes:
a- an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline
Students apply knowledge of computer organization to efficient hardware-dependent program design in assembly language of a simple processor.

b- an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
Students are exposed to different aspects of hardware design and will be able to propose a solution to the problem and an improvement under different conditions.

c- an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired.
Students are required to write instructions in assembly programs to better understand the inner workings of uni-processor computers.

d- an ability to function effectively on teams to accomplish a common goal

ec- an understanding of professional, ethical, legal and social issues and responsibilities

f- an ability to communicate effectively
g- an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

h- a recognition of the need for, and an ability to engage continuing professional development

i- an ability to use the current techniques, skills, and tools necessary for computing practice.

j- an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

k- an ability to apply design and development principles in the construction of software systems of varying complexity

Relationship of Course to Program Outcomes

The relationship shown in Table 1. This course meets CS/ABET’s outcomes a, b, and c.

<table>
<thead>
<tr>
<th>Course Learning Outcomes</th>
<th>Student Outcomes</th>
<th>(a)</th>
<th>(b)</th>
<th>(c)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>35%</td>
<td>25%</td>
<td>40%</td>
</tr>
<tr>
<td>1</td>
<td>x</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>x</td>
<td>x</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>x</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Criterion 4 – Professional Component:

The course involves system design as part of its assignment

Prepared by: Dr Mohamed A BERBAR

Reviewed by: Dr Rachid S Sammouda