CMPEN 331
Computer Organization and Design
Required Course for CMPEN and CMPSC

Catalog Data: Computer Organization and Design (3)
Introduction to major components of a computer system, how they function together in executing a program, how they are designed. Prerequisite: CMPEN 271; CMPSC 121 or CMPSC 201.


Course Objectives: Ultimately a computer is simply a set of gates and transistors connected in such a way as to enable the execution of a program stored in memory. A computer architecture encompasses the definition of the machine’s instruction set architecture, its use in creating a program, and its implementation in hardware. This course addresses the bridge between gate logic and executable software, and includes programming both in assembly language (representing software) and VHDL (representing hardware).

Primary Course Outcomes: Upon completion of the course, students should posses the following knowledge and skills:

- An understanding of a machine’s instruction set architecture (ISA) including basic instruction fetch and execute cycles, instruction formats, control flow, and operand addressing modes.
- The ability to create, assemble, execute, and debug assembly language programs along with a basic understanding of the assembly, linker, and loader processes.
- An understanding of a hardware description language (e.g., either VHDL or verilog) including their uses, structural, and behavioral descriptions.
- The ability to create, simulate, and debug a VHDL or verilog program.
- An understanding of the design and functioning of a machine’s central processing unit (CPU) including the datapath components (ALU, register file) and the control unit.
- An understanding of basic input/output functioning including program controlled I/O and interrupt I/O.
- An understanding of organization of memory hierarchies including the basics of cache design and DRAM architectures.

Relationship to Undergraduate Program Outcomes: CMPEN 331 is the first in a sequence of computer organization and architecture courses (the second being CMPEN 431) which collectively support the following program outcomes:

- Design of the electronic/logic circuits that form the basic building blocks of a computer system.
- Design of the organization and architecture of the basic components of a computer system.
- Analyze the performance of hardware systems using simulation methods.
- Demonstrate independent learning by using unfamiliar computer systems and software tools to solve technical problems.
- The ability to discuss major trends in industry and current research activities within computer architecture design.
Required Topics: (38 hrs total)
- Course introduction, basics of a computer system, introduction to SPIM (~2 hrs)
- Introduction to assembly language programming: add, load, store (~3 hrs)
- Assembly language logic and control flow instructions (~3 hrs)
- Supporting procedure calls and returns; addressing modes (~3 hrs)
- Introduction to the design and use of hardware description languages (~2 hrs)
- Describing basic hardware components in behavioral/structural VHDL (~1.5 hrs)
- Review of machine number representations; basic arithmetic operations (~1.5 hrs)
- Arithmetic Logic Unit (ALU) design (~2 hrs)
- Introduction to machine datapath design (~2 hrs)
- Designing a single cycle MIPS datapath (~3 hrs)
- Designing a multicycle MIPS datapath and its control unit (~3 hrs)
- Microprogramming (~1.5 hrs)
- Introduction to datapath pipelining (~1.5 hrs)
- Machine input and output; dealing with exceptions and interrupts; bus design (~3 hrs)
- Memory hierarchies, basic DRAM architectures, the basics of cache design (~3 hrs)
- Review and exams (~3 hrs)

Class Format: Two lectures per week; each lecture is 75 minutes. Students work in an unsupervised, open lab to complete assignments/projects.

Professional Component: CMPEN 331 provides the computer organization principles needed to understand a computer’s operation and prepares the student for future courses such as CMPEN 431. This course covers the topics that allow the student to understand the bridge between the hardware and the software in a computer system.

Evaluation:
- ~70% proctored assessments (exams)
- ~25% unproctered assignments (homeworks, programming projects)
- ~5% unproctered quizzes (via ANGEL)

Suggested breakdown based of 500 pts as follows:

<table>
<thead>
<tr>
<th>Component</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exam 1</td>
<td>100</td>
</tr>
<tr>
<td>Exam 2</td>
<td>100</td>
</tr>
<tr>
<td>Final</td>
<td>150</td>
</tr>
<tr>
<td>Homworks and programming projects</td>
<td>125</td>
</tr>
<tr>
<td>Online quizzes</td>
<td>25</td>
</tr>
</tbody>
</table>

Programming Assignment: Tentative Schedule

Note that the first four and last assignments use PCSpim – the MIPS assembler, simulator system. The fifth, sixth, and seventh assignments use Mentor Graphics ModelSim – a VHDL (or verilog) simulation system. Both tools are available on either windows or linux (Solaris) boxes and can be downloaded (for free) by the students onto their laptops.

<table>
<thead>
<tr>
<th>Topics</th>
<th>Assigned</th>
<th>Due</th>
<th>Worth</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction to SPIM</td>
<td>L2</td>
<td>L4</td>
<td>5</td>
</tr>
<tr>
<td>Writing a simple (add, load, store) MIPS program</td>
<td>L4</td>
<td>L6</td>
<td>10</td>
</tr>
<tr>
<td>Writing a more advanced (control flow) MIPS program</td>
<td>L6</td>
<td>L9</td>
<td>15</td>
</tr>
<tr>
<td>Writing a recursive MIPS program</td>
<td>L9</td>
<td>L12</td>
<td>20</td>
</tr>
<tr>
<td>Introduction to VHDL simulation tools</td>
<td>L14</td>
<td>L16</td>
<td>5</td>
</tr>
<tr>
<td>Designing a basic MIPS ALU in VHDL</td>
<td>L16</td>
<td>L20</td>
<td>25</td>
</tr>
<tr>
<td>Designing a single or multi-cycle MIPS datapath in VHDL</td>
<td>L20</td>
<td>L25</td>
<td>25</td>
</tr>
<tr>
<td>Interrupt I/O using SPIM</td>
<td>L25</td>
<td>L28</td>
<td>20</td>
</tr>
</tbody>
</table>

Author: Mary Jane Irwin, Yuan Xie
Last Revised: July 1, 2007