**EE 361 Digital Systems and Computer Design**

**Credits:**  3

**Categorization of credits:** Engineering topic

**Instructor’s or course coordinator:** Galen Sasaki, Jan. 9, 2021.

**Designation**: Required for Computer Engineering, and Elective for Electrical Engineering

**Text Book and Other Required Materials:**

* "Computer Organization and Design: The Hardware/Software Interface" by Patterson and Hennessey.
* “Logic Design and Verification Using SystemVerilog, Revised” by Donald Thomas.

**Catalog Description:** EE 361 Digital Systems and Computer Design (3) Design methodology, processor design, control design, memory organization, system organization. Pre: 160 and 260, or consent.

**Pre-and Co-requisites:** Pre-requisites: EE 160 "Programming for Engineers" and EE 260 "Introduction to Digital Design".

**Class/Lab Schedule:** 3 lecture hours per week

**Topics Covered:**

1. Principles of instruction set design, the role of compiler and optimizations, assembly language, machine level programs, how high level programs are implemented with machine instructions (10 hours)
2. Review of digital circuits, elementary design methods, and hardware description language (HDL) (3 hours)
3. Processor data path design (6 hours)
4. Processor controller design (3 hours)
5. Algorithms and circuits for integer arithmetic, and the arithmetic logic unit (ALU) (6 hours)
6. Data representations: review of representing characters and integers (1 hour)
7. Floating point representation and arithmetic (2 hours)
8. Memory system design, implementing memory cells with transistors, RAM, memory hierarchy, caches (4 hours)
9. I/O systems (2 hours)

**Course Objectives and Their Relationship to Program Objectives:**

A student should understand (i) computer organization, (ii) the principles of designing efficient computers, and (iii) the relationship between programs (software) and the hardware they run on. A student should be able to design efficient and complex digital systems, such as a RISC single-cycle, multi-cycle, and pipelined processors, computer arithmetic circuits, and control circuits. In addition, a student should master modern design methods for digital circuits, including appropriate computer-aided design (CAD) tools.

**Program Objectives this course addresses**: 1, 2, 3, 4, 5.

**Course Outcomes and Their Relationship to Program Outcomes:**

The following are the course outcomes and the subset of Program Outcomes (numbered 1-8 in square braces "[ ]") they address:

* Design assembly language programs from simple programs written in a high-level language. [1,2]
* Design assembly language functions that can be called by programs written in a high-level language. [1,2]
* Translate assembly language programs into machine language programs. [1,2]
* Design simple I/O drivers. [1,2]
* Understand the relationship between programs and the computer hardware they run on. [1]
* Understand how data is represented in computers [1,8]
* Understand algorithms for arithmetic and be able to design arithmetic circuits. [1,2,8]
* Design a data path for a computer. [1,2]
* Design a controller for a computer. [1,2]
* Understand pipelined computer architecture and dealing with data and control hazards [1,2]
* Learn to use a hardware description language (HDL) to design complex circuits at the behavioral level [1,2,7].
* Learn to use an HDL functional simulation to verify and debug designs [7]
* Implement a computer using an HDL. [1,2,8]
* Understand memory hierarchy and the algorithms. [1,8]
* Use tools for assembly language (machine level) programming such as the SPIM simulator. [1,2,7]

**Contribution of Course to Meeting the Professional Component**

Engineering Topics: 100%

**Computer Usage:**

A SPIM MIPS processor simulator is used to run and debug assembly language programs.

Linux servers are used to run high-level programming language assignments.

To simulate HDL code (e.g., SystemVerilog code), functional simulators such as *EDAPlayground.com* and *Xilinx Webpack* are used.

Approximately 30% of the assignments use computers.

**Design Credits and Features:**

EE 361 has 1 design credit. About 10% of the homework assignments are writing programs or fragments of programs in assembly or machine language. About another 20% are circuit design problems, which includes a 4-week task of implementing a single cycle processor using a hardware description language (HDL) such as SystemVerilog.