TCSS 372: Machine Organization Master Syllabus
Version: April 2011
(Approved: 27 May 2011)

Catalog Description
Covers the micro architecture level of machine design and advanced architecture features for performance enhancement. Subjects include bus, memory and CPU design, hardware support for operating systems, CISC/RISC architectures, and parallelism. Prerequisite: a minimum grade of 2.0 in TCSS 371.

Preconditions
Prior to taking the class, student must be able to:

- explain the function of basic digital logic circuits
- translate between assembly instructions and machine code
- write short programs in assembly language including function calls
- write small to medium programs in C
- explain the instruction execution cycle

Course Objectives
The objectives of this course are to teach students:

- Array processing and recursion
- CPU implementation, i.e. control signals in single and multi-cycle machines
- Pipeline architectures including forwarding, stalling, exception handling
- Branch prediction
- Cache organization and performance
- Instruction level parallelism
- Microprogramming
- Address translation in virtual memory systems

Student Learning Outcomes
Upon successful completion of the course, students should be able to:

- Write array processing programs with multiple functions in assembly language
- Explain the implementation of arithmetic algorithms (e.g. multiplication) in hardware
- Trace the flow of data and control signals through a CPU
- Explain the function of pipeline techniques such as forwarding and stalling
- Identify strategies for branch prediction
- Understand the possible benefits and limitations of parallel instruction execution
- Analyze and modify microcode
- Explain how a cache can improve program execution time

Relationship of course to CSS student learning outcomes: This course supports and assesses the achievement of the following elements of the program objectives:

- An ability to apply knowledge of computing and mathematics appropriate to the discipline

- An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
- An ability to use current techniques, skills, and tools necessary for computing practice.

Relationship to UWT student learning goals (to be added to syllabus handed out to students)

- *Inquiry and Critical Thinking*: Students will acquire skills and familiarity with modes of inquiry and examination from diverse disciplinary perspectives, enabling them to access, interpret, analyze, quantitatively reason, and synthesize information critically.
- *Communication/Self-Expression*: Students will gain experience with oral, written, symbolic and artistic forms of communication and the ability to communicate with diverse audiences. They will also have the opportunity to increase their understanding of communication through collaboration with others to solve problems or advance knowledge.

Topics Covered

- MIPS Assembly Language
- Computer Arithmetic
- Single Cycle Machine Implementation
- Multi-cycle Machine Implementation
- Pipeline Architecture
- Cache Memory
- Microprogramming
- Instruction Level Parallelism
- Virtual Memory