COURSE SYLLABUS: CPSC5155U - COMPUTER ARCHITECTURE - FALL 2014

INSTRUCTOR: Angkul Kongmunvattana, Ph.D. (Associate Professor of Computer Science)
EMAIL: kongmunvattana_angkul@columbusstate.edu
HOMEPAGE: http://csc.columbusstate.edu/kongmunvattana
PHONE: (706) 507-8170
OFFICE HOURS AND LOCATION: 2pm to 5pm on Tuesday, Wednesday, and Thursday; 11am to 12pm on Wednesday; and also by appointment / CCT 426
MEETING TIME AND PLACE: Tuesday and Thursday 9:30am to 10:45am / CCT 406 (Class attendance is mandatory)

COURSE INFORMATION

COURSE CRN NUMBER/TITLE: CRN 82886 / CPSC5155U – Computer Architecture
CREDIT HOURS/PREREQUISITES: 3 credits / CPSC3121 with a grade of “C” or higher

COURSE DESCRIPTION: Review of combinational and sequential logic. Description of computer memory. Study of the instruction set architecture and addressing modes of a simple computer. Design on the computer control unit including both hard-wired and microprogrammed CPU's. Design of Input/Output including program-controlled I/O, interrupted-driven I/O, direct memory access (DMA), and I/O channels.

REQUIRED TEXTBOOK AND MATERIALS: NONE

SUPPLEMENTARY BOOKS AND MATERIALS (OPTIONAL):

- Computer Architecture: A Quantitative Approach by John Hennessy and David Patterson
- Selected Papers from ISCA, MICRO, ASPLOS, HPCA, ICS, ICCD, and CGO
- Intel 64 and IA-32 Architectures Software Developer Manuals

LEARNING OUTCOMES

Course Objective:
The aim of this course is to introduce students to the contemporary design and implementation of microprocessors. In particular, students will demonstrate an understanding of microarchitecture concepts and features employed by contemporary microprocessors, such as pipelining, data forwarding, scoreboard, Tomasulo, superscalar, VLIW, SMT, CMP, branch predictors, trace cache, binary translation, in-order execution, out-of-order execution, re-order buffer, data prefetching, and cache coherence protocols. Additionally, students will also demonstrate a competence of basic skills in optimizing assembly programs.

Course Outcomes:

- Students will demonstrate knowledge of digital logic analysis and design.
  - Strategies and Actions used to produce the outcome:
    - Review of digital logic components.
    - Review of digital logic circuit analysis, design, and optimization.
    - Review of digital logic circuit design and simulation tools.
  - ABET Criteria covered: A, B, C, I, and J.
  - Program Objectives covered: 2 and 3.
  - Assessment Methods: Assignments and Exams.
- Students will demonstrate knowledge of instruction execution techniques.
  - Strategies and Actions used to produce the outcome:
    - Study of microprocessor execution pipelines.
    - Study of superscalar concepts and implementation.
- Study of SMT concepts and implementation.
- Study of VLIW concepts and implementation.
- Study of in-order and out-of-order instruction executions.
- Study of CMP concepts and implementation.
  - ABET Criteria covered: A, B, C, and J.
  - Program Objectives covered: 2 and 3.
  - Assessment Methods: Assignments and Exams.

- Students will demonstrate knowledge of microarchitecture features.
  - Strategies and Actions used to produce the outcome:
    - Study of the data forwarding technique.
    - Study of the scoreboard technique.
    - Study of the Tomasulo’s technique.
    - Study of the branch prediction techniques.
  - ABET Criteria covered: A, B, C, and J.
  - Program Objectives covered: 2 and 3.
  - Assessment Methods: Assignments and Exams.

- Students will demonstrate knowledge of microarchitecture optimization techniques.
  - Strategies and Actions used to produce the outcome:
    - Study of the binary translation technique.
    - Study of trace cache and its derivative.
    - Study of data prefetching techniques.
    - Study of re-order buffer design and implementation.
  - ABET Criteria covered: A, B, C, and J.
  - Program Objectives covered: 2 and 3.
  - Assessment Methods: Assignments and Exams.

- Students will demonstrate knowledge of cache coherence protocols.
  - Strategies and Actions used to produce the outcome:
    - Study of the memory consistency models.
    - Study of the snoopy and directory-based cache coherence protocols.
  - ABET Criteria covered: A, B, C, and J.
  - Program Objectives covered: 2 and 3.
  - Assessment Methods: Assignments and Exams.

- Students will demonstrate knowledge of assembly programming optimization.
  - Strategies and Actions used to produce the outcome:
    - Study of software developer manuals from the microprocessor manufacturers.
    - Study of assembly programming optimization techniques.
  - ABET Criteria covered: A, B, C, I, and J.
  - Program Objectives covered: 2 and 3.
  - Assessment Methods: Assignments and Exams.

**COURSE ASSESSMENT**

**LEARNING ACTIVITIES**

1. The class will meet twice a week for one hour and fifteen minutes of lecture each.
2. Each student is expected to attend all class lectures, to complete the non-graded quizzes, and to take good notes. Students will be expected to submit their graded assignments at the beginning of class on the due dates.
3. Students must have access to computers and/or computer labs for doing assignments.
4. The ACM recommends the following: “As a general guideline, the amount of out-of-class work is approximately three times the in-class time. Thus, a unit that is listed as requiring 3 hours typically entails a total of 12 hours (3 in class and 9 outside).” Students will be expected to spend this time outside class studying their lecture notes and handouts as well as thinking and completing assignments.
COURSE EVALUATION

<table>
<thead>
<tr>
<th>GRADED LEARNING ACTIVITIES</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignments</td>
<td>30%</td>
<td></td>
</tr>
<tr>
<td>Midterm Exam</td>
<td>30%</td>
<td></td>
</tr>
<tr>
<td>Final Exam</td>
<td>30%</td>
<td></td>
</tr>
<tr>
<td>Class Attendance (based on the records of non-graded quizzes)</td>
<td>10%</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>100%</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Percentage Range</th>
<th>Final Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>90-100%</td>
<td>A</td>
</tr>
<tr>
<td>80-89%</td>
<td>B</td>
</tr>
<tr>
<td>65-79%</td>
<td>C</td>
</tr>
<tr>
<td>55-64%</td>
<td>D</td>
</tr>
<tr>
<td>54% and below</td>
<td>F</td>
</tr>
</tbody>
</table>

ADMINISTRATIVE POLICIES AND ACADEMIC RESOURCES

CSU DISABILITY POLICY
If you have a documented disability as described by the Americans with Disabilities Act (ADA) and the Rehabilitation Act of 1973, Section 504, you may be eligible to receive accommodations to assist in programmatic and/or physical accessibility. We recommend that you contact the Office of Disability Services located in Schuster Student Success Center, Room 221, 706-507-8755 as soon as possible. Students taking online courses can contact the Office of Disability services at http://disability.columbusstate.edu/. The Office of Disability Services can assist you in formulating a reasonable accommodation plan and in providing support. Course requirements will not be waived but accommodations may be able to assist you to meet the requirements. Technical support may also be available to meet your specific need.

ACADEMIC INTEGRITY
All students are expected to recognize and uphold standards of intellectual and academic integrity. As a basic and minimum standard of conduct in academic matters that students be honest and that they submit for credit only the products of their own efforts. Both the ideals of scholarship and the need for fairness require that all dishonest work be rejected as a basis for academic credit. They also require that students refrain from any and all forms of dishonorable or unethical conduct related to their academic work.
Students are expected to comply with the provisions of Section III, "Student Responsibilities," of the Columbus State University Student Handbook. This specifically includes the sections on "Academic Irregularity," and "Conduct Irregularity." In particular, the Columbus State University Student Handbook states:

“No student shall give or receive assistance in the preparation of any assignment, essay, laboratory report, or examination to be submitted as a requirement for any academic course in such a way that the submitted work can no longer be considered the personal effort of the student submitting the work.”

**Examples of Academic Dishonesty include but are not limited to:** Plagiarism (see definition below), giving or receiving unauthorized assistance on exams, quizzes, class assignments or projects, unauthorized collaboration, multiple submissions (in whole or part) of work that has been previously submitted for credit.

Plagiarism is any attempt to represent the work or ideas of someone else as your own. This includes purchasing or obtaining papers from any person and turning them in as your own. It also includes the use of paraphrases or quotes from a published source without properly citing the source. All written assignments may be submitted for textual similarity review to Turnitin.com for the detection of plagiarism.

Any work turned in for individual credit must be entirely the work of the student submitting the work. **All work must be your own.** You may share ideas but submitting identical assignments (for example) will be considered cheating. **You may discuss the material in the course and help one another with debugging, however, I expect any work you hand in for a grade to be your own.** A simple way to avoid inadvertent plagiarism is to talk about the assignments, but don't read each other's work or write solutions together. Keep scratch paper and old versions of assignments until after the assignment has been graded and returned to you. **If you have any questions about this, please see me immediately.**

For assignments, access to notes, textbook, books and other publications is allowed. Stealing, giving or receiving any code, diagrams, drawings, text or designs from another person (CSU or non-CSU) is not allowed. Having access to another person’s work on the system or giving access to your work to another person is not allowed. It is your responsibility to keep your work confidential.

No cheating in any form will be tolerated. Please be aware that anyone caught cheating or plagiarizing in this class will receive a “0” for the assignment/exam and may receive an “F” for the course.

**STUDENT COMPLAINT PROCESS**

Information and resources for student complaints and academic appeals are located at the following link on the Columbus State University website [http://aa.columbusstate.edu/appeals/](http://aa.columbusstate.edu/appeals/).

**COURSE ATTENDANCE POLICY**

Class attendance is the responsibility of the student, and it is the student's responsibility to independently cover any materials missed. At my discretion, I may drop you from the course when you accumulate more than **four (4)** absences. Missing an exam or a quiz is considered an absence. Missed classes caused by participation in documented, formal, University-sponsored events will not count as absences provided you notify me of such anticipated absences in advance and as soon as possible. **You are responsible for all class work missed, regardless of the reason for the absence(s).** Late assignments will **not be accepted**, so if you are absent on the day an assignment is due, it is your responsibility to make alternate arrangements. No makeup exams or quizzes will be given, so please make sure you are present for all exams/quizzes.

Refer to the CSU Catalog [http://ace.columbusstate.edu/advising/a.php#AttendancePolicy](http://ace.columbusstate.edu/advising/a.php#AttendancePolicy) for more information on class attendance and withdrawal.
Getting help
Student assistants in the public Computer Center labs / Library can help you with basic computer-related problems such as logging on to the network, saving your work, etc., but they are not obligated to help you with your assignments. There are a few tutors in the Computer Science lab (CCT450) who can help you with the assignments. Their schedule is posted in the Computer Science School. You can always contact me during my posted office hours or by appointment.

Student Responsibilities
- manage your time and maintain the discipline required to meet the course requirements,
- come to class prepared to ask questions to maximize your understanding of the materials taught,
- complete all assignments, quizzes and exams

“I didn’t know” is NOT an acceptable excuse for failing to meet the course requirements. If you fail to meet your responsibilities, you do so at your own risk.

TENTATIVE Course Schedule

<table>
<thead>
<tr>
<th>Days</th>
<th>Subjects</th>
<th>Days</th>
<th>Subjects</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (August 19)</td>
<td>Class Administration</td>
<td>17 (October 14)</td>
<td>Fall Break</td>
</tr>
<tr>
<td>2 (August 21)</td>
<td>Combinational Circuit Analysis, Design, and Optimization</td>
<td>18 (October 16)</td>
<td>Binary Translation</td>
</tr>
<tr>
<td>3 (August 26)</td>
<td>Combinational Circuit Implementation and Simulation</td>
<td>19 (October 21)</td>
<td>Trace Cache</td>
</tr>
<tr>
<td>4 (August 28)</td>
<td>Sequential Circuit Analysis, Design, and Optimization</td>
<td>20 (October 23)</td>
<td>Data Prefetching</td>
</tr>
<tr>
<td>5 (September 2)</td>
<td>Sequential Circuit Implementation and Simulation</td>
<td>21 (October 28)</td>
<td>ROB</td>
</tr>
<tr>
<td>6 (September 4)</td>
<td>Contemporary CPU Designs</td>
<td>22 (October 30)</td>
<td>Memory Consistency Models</td>
</tr>
<tr>
<td>7 (September 9)</td>
<td>Microprocessor Designs</td>
<td>23 (November 4)</td>
<td>Cache Coherence Protocols-I</td>
</tr>
<tr>
<td>8 (September 11)</td>
<td>Instruction Execution Pipelines</td>
<td>24 (November 6)</td>
<td>Cache Coherence Protocols-II</td>
</tr>
<tr>
<td>9 (September 16)</td>
<td>Superscalar, VLIW, SMT, and CMP</td>
<td>25 (November 11)</td>
<td>Code Optimization-I</td>
</tr>
<tr>
<td>10 (September 18)</td>
<td>In-order and Out-of-order Executions</td>
<td>26 (November 13)</td>
<td>Code Optimization-II</td>
</tr>
<tr>
<td>11 (September 23)</td>
<td>Data Forwarding</td>
<td>27 (November 18)</td>
<td>Code Optimization-III</td>
</tr>
<tr>
<td>12 (September 25)</td>
<td>Scoreboard</td>
<td>28 (November 20)</td>
<td>Code Optimization-IV</td>
</tr>
<tr>
<td>13 (September 30)</td>
<td>Tomasulo</td>
<td>29 (November 25)</td>
<td>Contemporary Issues</td>
</tr>
<tr>
<td>14 (October 2)</td>
<td>Branch Predictors</td>
<td>30 (November 27)</td>
<td>Thanksgiving Break</td>
</tr>
<tr>
<td>15 (October 7)</td>
<td>Review Session</td>
<td>31 (December 2)</td>
<td>Contemporary Issues</td>
</tr>
<tr>
<td>16 (October 9)</td>
<td>Midterm Exam</td>
<td>32 (December 4)</td>
<td>Review Session</td>
</tr>
</tbody>
</table>
ABET Criteria:
Students in CS/IT will have a(n)
A. ability to apply knowledge of computing and mathematics appropriate to the discipline;
B. ability to analyze a problem, and identify and define the computing requirements appropriate to its solution;
C. ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs;
D. ability to function effectively on teams to accomplish a common goal;
E. understanding of professional, ethical, legal, security, and social issues and responsibilities;
F. ability to communicate effectively with a range of audiences;
G. ability to analyze the local and global impact of computing on individuals, organizations and society;
H. recognition of the need for, and an ability to engage in, continuing professional development;
I. ability to use current techniques, skills, and tools necessary for computing practice.
J. 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. ability to apply design and development principles in the construction of software systems of varying complexity.

CS Program Objectives:
Our graduates will have achieved:
1) A broad general education assuring an adequate foundation in science and mathematics relevant to computing.
2) A solid understanding of concepts fundamental to the discipline of computer science.
3) Good analytic, design, and implementation skills required to formulate and solve computing problems.
4) The ability to function and communicate effectively as ethically and social responsible computer science professionals.

By signing below, you (________________________________________________________) are hereby acknowledged that you have carefully read and agreed to abide by the rules and policies stated in this course syllabus.

______________________________________________________________
(Signature) (mm/dd/yyyy)
ACM Code of Ethics and Professional Conduct

THE CODE represents ACM’s commitment to promoting the highest professional and ethical standards, and makes it incumbent on all ACM Members to:

- Contribute to society and human well-being.
- Avoid harm to others.
- Be honest and trustworthy.
- Be fair and take action not to discriminate.
- Honor property rights including copyrights and patent.
- Give proper credit for intellectual property.
- Respect the privacy of others.
- Honor confidentiality.

And as computing professionals, every ACM Member is also expected to:

- Strive to achieve the highest quality, effectiveness and dignity in both the process and products of professional work.
- Acquire and maintain professional competence.
- Know and respect existing laws pertaining to professional work.
- Accept and provide appropriate professional review.
- Give comprehensive and thorough evaluations of computer systems and their impacts, including analysis of possible risks.
- Honor contracts, agreements, and assigned responsibilities.
- Improve public understanding of computing and its consequences.
- Access computing and communication resources only when authorized to do so.

This flyer shows an abridged version of the ACM Code of Ethics. The complete version can be viewed at: www.acm.org/constitution/code