CSE381: Computer Architecture

Syllabus: Fall 2012

1. Teaching personnel

Instructor: Qinru Qiu, Department of Electrical Engineering and Computer Science
Phone: (315) 443-1836 Email: qiqiu@syr.edu
Instructor's office hours: CST 4-281, 11:30AM ~ 3:30PM Tuesday or by appointment

2. Class schedule

Lecture
Day: Tuesday & Thursday
Location: CST 4-216
Time: 9:30AM ~ 10:50AM

Help Session
Day: Wednesday
Location: Link Hall 103
Time: 11:40am-12:35pm

3. Text Book

Required: David A. Patterson, John L. Hennessy, Computer Organization and Design, 4th Edition, Morgan Kaufmann, 2008

4. Course Web Page

All instruction materials, including lecture notes, homework and lab assignments will be posted on BlackBoard@SU. Look for Blackboard announcement for important messages.

5. Description

Our goal in this course is to study the fundamentals of computer architecture and learn how to analyze and compare computer performance from processor or system point of view.

The course starts with a basic review of digital design, assembly language and Boolean arithmetic. we continue then with quantitative analysis of processors. Later, we will introduce a basic single instruction per cycle CPU architecture and extend the basic component of which to develop a pipeline processor. We also study the effect of memory and memory organizations on overall performance of a computer. We wrap up with Virtual Memory and special topics on branch prediction, storage elements and/or superscalar processor.

6. Course Objectives

After taking the course, the students should have the ability to:

- describe and understand the principle characteristics of a computer architectures and digital logic layer defining the technology with respect to computer system and systematic.
- translate between various formal representations for design and analysis, (e.g., translate instruction set architecture to data path and control path) and interpret the meaning of the various formal representations.
- analyze and enhance the computation power by evaluating the processor performance with respect to a given abstract description of a component or a system implementation
- understand the bus structures existing in computer architectures and the layered memory system including cache and main memory.
- when given an abstract description of computing system, you should be able to create a concrete implementation (such as data path and its controller) that is a faithful refinement of the specification

7. Topics covered

1. Introduction. ISA. Performance analysis. (Homework #1)
2. Technology, design, & performance analysis. (Homework #2)
3. Arithmetic & Floating point units. (Homework #3)
4. Single-cycle Datapath. & Controller. (Homework #4)

5.	Midterm Exam Review.	**(Midterm Exam)**
6.	Multi-cycle Datapath.	(Homework #5)
7.	Pipeline Control. Advanced pipelining.	
8.	Memory systems & Caches.	(Homework #6)
9.	Branch Prediction Techniques	(Homework #7)
10.	Exceptions. Pipelining. Pipelining Hazards.	(Homework #8)
11.	Virtual Memory.	(Special Topics)
12.	Superscalar architectures. ILP.	(Special Topics)
13.	Busses, I/O systems.	(Time-Permit)
14.	Interconnection Networks.	(Time-Permit)
15.	Multiprocessors. Future Trends. Sample CPUs.	(Time-Permit)

Note: The course material might change during the semester depending on the progress of the class. All departmental, college and university regulation regarding class attendance, course drop, etc will be followed.

8. Grading policy

Final Exam: 35%
Mid Term Exams: 30%
Home Work Assignments: 30%
Pop Quizzes/Attendance: 5%

Grading Scale:
90 ~ 100 = As
80 ~ 89 = Bs
70 ~ 79 = Cs
60 ~ 69 = D
Below 60 = F

9. Homework Policy

Homework assignments are to be submitted through Blackboard website or hand to the instructor on the assignment due date. Assignments submitted after the due date will be deducted 10 points for each day late.

10. Exams

All exams must be taken at the scheduled time unless a previous arrangement (with a good reason) has been made with the instructor

11. Attendance

You are expected to attend each class punctually and remain for the entire class period. You need to inform the instructor in advance if you expect to miss a class or leave the course before the end of the semester. If you miss class your absence will be excused by the instructor only if a doctor's certificate or other evidence is submitted. You remain to be responsible for the work associated with the class you missed, even if your absence has a valid reason. There will be a number of unannounced popup quizzes during the semester.

12. Academic Honesty

Cheating in any form is not tolerated, nor is assisting another person to cheat. The submission of any work by a student is taken as a guarantee that the thoughts and expressions in it are the students own except when properly credited to another.
Violations of this principle include giving or receiving aid in an exam or where otherwise prohibited, fraud, plagiarism, the falsification or forgery of any record, and any other deceptive act in connection with academic work. Plagiarism is the representation of another's words, ideas, programs, formulae, options or other products of work as one's own work from others, since it is often not possible to determine who the originator or the copier was. Such offense will result in a failing grade "F" and a letter of reprimand in your department student file.

Academic integrity statement: The Syracuse University Academic Integrity Policy holds students accountable for the integrity of the work they submit. Students should be familiar with the Policy and know that it is their responsibility to learn about instructor and general academic expectations with regard to proper citation of sources in written work. The policy also governs the integrity of work submitted in exams and assignments as well as the veracity of signatures on attendance sheets and other verifications of participation in class activities. Serious sanctions can result from academic dishonesty of any sort. For more information and the complete policy, see http://academicintegrity.syr.edu.

Disability Statement: Students who are in need of disability-related academic accommodations must register with the Office of Disability Services (ODS), 304 University Avenue, Room 309, 315-443-4498. Students with authorized disability-related accommodations should provide a current Accommodation Authorization Letter from ODS to the instructor and review those accommodations with the instructor. Accommodations, such as exam administration, are not provided retroactively; therefore, planning for accommodations as early as possible is necessary. For further information, see the ODS website, Office of Disability Services.

Appendix A. Required Knowledge and Skills

Recall:

1. **You should be able to explain operations pertained to a basic electrical circuit, gate technologies, and memory systems.**
2. **You should be able to explain Boolean algebra and the design methods such as Combinational logic design and Sequential machine design.**
3. **You should be able to enumerate the basic components of a computer.**
4. **You should be able to enumerate the steps required to execute a program.**

Comprehension:

5. **When given an integer number in decimal, binary or hexadecimal representation, you should be able to translate it to any other base including two's complement.**
6. **You should be able to translate between various formal representations for design and analysis, (e.g., translate state tables into a schematic of registers and logic, translate logic schematics into logic functions, etc.) and interpret the meaning of the various formal representations.**
7. **When given an abstract description or model of a component or system, you should be able to translate the description into an alternative abstract description or model.**
8. **You should be able to recognize the basic components of a computing system as ALU, Memory, and Input/Output Systems and explain their operations.**
9. **When given a computer architecture, you should be able to explain the Instruction Execution Cycle.**
10. **You should be able to describe a microprocessor architecture, memory management and input-output system, the assembly, linking and running processes.**

Application:

11. **When given English language descriptions of behavior, you should be able to translate them into the appropriate specification such as a Boolean equation or a state transition diagram.**
12. **When given a formal model specifying the behavior of a component or system such as a finite-state machine specification, you should be able to refine it into an implementation consisting of registers and combinational logic (encoders, decoders, multiplexers, adders, ALUs, and counters).**
13. **When given a flowchart, you should be able to write a program in assembly language that includes conditional processing, and integer and logic arithmetic.**

Appendix B. Acquired Knowledge and Skills

Recall:

1. **You should be able to explain layered computer architectures and digital logic layer defining the technology.**
2. **You should be able to define the differences between various classes of systems such as CISC & RISC, micro programmable & wired decoding, SIMD & MIMD systems, and distributed computing & parallel processing.**
3. **You should be able to explain bus structures existing in computer architectures and the layered memory system including cache and main memory.**

Comprehension:

4. **You should be able to translate between various formal representations for design and analysis, (e.g., translate instruction set architecture to data path and control path) and interpret the meaning of the various formal representations.**
5. **You should be able to recognize the pipelining hazards and means to avoid them.**

Application:

6. **When given a formal model specifying the behavior of a component or system such as a data path specification, you should be able to refine it into an implementation consisting of registers and combinational logic.**
7. **You should be able to modify a system to enhance its overall throughput using various techniques such as pipelining, dual-ported memory, and multiple execution units.**

Analysis:

8. **When given an abstract description of a component or system and an implementation, you should be able to determine if the implementation is faithful to the abstract description.**
9. **When presented an enhancement in a system, you should be able to analyze its effect on the overall performance and cost.**

Synthesis:

10. **When given an abstract description of computing system, you should be able to create a concrete implementation (such as data path and its controller) that is a faithful refinement of the specification.**

Evaluation:

11. **You should be able to judge the degree of correspondence between abstract behavioral specifications and the actual behavior of implementations.**
12. **When given a specification and implementation, you should be able to evaluate whether the implementation is correct with respect to its specification and meets performance requirements.**

Appendix C. Contributions of the course towards students achieving the Computer Engineering Educational Outcomes[1]

Computer Engineering Educational Outcomes	CSE 381
a. An ability to apply knowledge of mathematics, science and engineering	S
c. An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social , ethical, health and safety, manufacturability and sustainability	M
e. An ability to identify, formulate, and solve engineering problems	S
j. A knowledge of contemporary issues	S
l. An ability to verify design correctness and evaluate performance of computing systems	S

CSE 381 – Computer Architecture

Prerequisites: CSE 261, CSE 281, MAT 397.

[1] Assessment Scale: W: Weak; M: Moderate; S: Strong.