CS130 Course Syllabus Updated July 7, 2017

Course Number and Title: CS130 **Computer Organization**
Number of Credits: 3
Instructor Name: David Davidian
Email Address: ddavidian@aua.am
Telephone Number: xxx-xx-xx-xx
Office Location: Room xxx
Office Hours: by appointment
Term/Year: Fall 2017
Class Schedule: MWF, xx:xx
Pre-/co-requisites: CS110, CS111, CS120, CS121
Teaching Assistant: TBD

Course Description: The topics include functional organization and operation of digital computers.

The assembly language module covers – data representation, instruction formats, addressing modes, stacks / frames, argument passing, arithmetic/logical/decision making/ input-output operations, modularization, compilers, linkers, debuggers, device drivers, system calls, computer data path and instruction level parallelism.

The **logic design module** covers - combinational logic, Karnaugh maps, circuit analyze timing diagrams, circuit synthesis, basic and complex logic devices (gates, latches, decoders, multiplexors, flip-flops, registers, counters)

Students are required to demonstrate intermediate knowledge of software and hardware systems related to computational sciences. Students are required to complete individual projects related to assembly level drivers. Three hours of instructor-led class time per week including discussions and problem sets.

Note: This syllabus is subject to change to accommodate students' need.

Required text and material: (supplied by instructor)
(1) x86 Assembly Language Reference Manual (many available, suggested pdf will be supplied)
(2) Access to an x86 based laptop or PC running DOSBOX and Debug (installation tutorial available), assembler/linker/loader will be provided.
(3) Class presentations notes generated by instructor.

Course Schedule

Week	Date	Topics
1	Aug 28, 30, Sept 1	Introductory class. Overview of Computer Systems: CPU, Memory (RAM, ROM, Flash), I/O, Storage (DAS, SAN, NAS).
2	Sept 4, 6, 8	Architecture CISC, RISC, Harvard, Von Neumann, x86 micro architecture, data flow through a CPU and memory, MMUs, Caching, Machine level representation of data: 4-64 bits. Virtual cloud-based systems. Number formats (Int and FP), introduction to x86 Assembly Language: bit, byte, word logical and shift operations, math functions, I/O operations.

3	Sept 11, 13, 15	Introduction to MASM-type assemblers/linkers. Introduction to operating systems. Embedded system: digital thermostats and Baxi systems In-class assembly debug sessions. Simple device drivers in assembly language
4	Sept 18, 20, 22	Introduction to operating systems, file systems, their function and architecture.
5	Sept 25, 27, 29	Moving from tiny systems to larger systems. Introduction to execution environments, from remote controls, cell phone, to large scale systems.
6	Oct 2, 4, 6	**Mid Term Exam I (Architecture)**
7	Oct 9, 11, 13	Boolean algebra review, combinational logic, gates, minimization, Karnaugh maps, truth tables
8	Oct 16, 18, 20	Introduction to hardware design
	Oct 23, 25, 27	System interfaces: circuits, signals, timing, race conditions, propagation delays. Simple micro-controller in-class design
9	Oct 30, Nov 1, 3	**Mid Term Exam II (Assembly Language)**
10	Nov 6, 8, 10	Sequential circuits, multiplexers, decoders, adders, simple ALU, latches flip-flops, counters
11	Nov 13, 15, 17	Software stacks (ex: network, system, SOA)
12	Nov 20, 22, ~~24~~	System benchmarking and real-life execution environments (system architecture)
13	Nov 27, 29, Dec 1	**Final Group Projects (assembly language hardware simulator)**
14	Dec 11, 13	**Final Group Projects (assembly language hardware simulator)**
	Dec 15…	~~**Final Exams**~~

Learning Objectives & Outcomes:

CS130 Course Syllabus Updated July 7, 2017

Program Goals	Program Student Learning Outcomes	Course Learning Outcomes
To provide knowledge and skills in designing, developing and managing efficient computing systems that provide the basis for solving real-life industrial and academic problems.	Students completing this course are expected to possess the following skills and abilities: 1. Understand the basics of CPU architecture recognize and describe the hardware components in modern compute systems. **I** 2. Learn and program in new assembly languages **B** 3. Utilize machine level debugging information when programming low level operations. **I** 4. Understand the basic components of operating systems. **B** 5. Understand and be familiar with computer system resources and their management. **I** 6. Attain skills enabling understanding multi-threaded code and device drivers. **B** 7. Design basic combinatorial logic blocks. **B**	1. Introduce the Von Neumann computer organization and machine level representation of data. 2. Introduce the x86 assembly language instructions by functional groups and operations. 3. Teach and operationally demonstrate assembly language concepts such as subroutines, interrupts, I/O operations. 4. Introduce the essential elements of operating systems and computer resources. 5. Introduce and exercise the organization of modern operating systems and programming concepts. 6. Teach students the basic concepts of hardware design associated with simple microprocessors and items that must be managed and controlled in real-life applications. 7. Demonstrate and exercise classic software debugging techniques at the hardware level. 8. Teach students how to create and simulate logical hardware circuits.

B = Beginner

I = Intermediate

A = Advanced

Course Structure: Instructor-led 50 minute classes, three per week, with homework in synergy with the syllabus.

Method of Evaluation:

Student learning will be evaluated on the basis of the following weighted components:

- Attendance and class participation (Bonus)
- Homework/Reports (20%)
- Midterm examinations and quizzes (50%)
- Final Exam/Project (30%)

Library and Media/Technology Use
Students are encouraged to use supplemental online and reference materials available at the library to enhance their overall learning in the course. Students are encouraged to use audio-visual aids and presentation software as appropriate. If students have any questions or need additional support in using library resources or technology, they should confer with library staff, ICT, or the instructor.

Late Policy
A half grade will be deducted from an assignment each day that it late (e.g. an assignment graded as an A will become a B+ if it is submitted two days late). The instructor might not penalize the student if the student submits convincing evidence of a medical or other emergency that made completing the assignment at the scheduled time impossible.

Make-up Procedures
Make-up assignment, exam, and quiz will be given at the instructor's discretion. Students must submit convincing evidence of a medical or other emergency that makes completing an assignment or taking an exam or quiz at the scheduled time impossible.

Policy on Grade Appeal
Students are entitled to appeal grades in line with the university's grade appeal policy which is available online at http://policies.aua.am/policy/11.

Standards for Academic Integrity
Students are required to conduct themselves in an academically responsible and ethical manner in line with the Student Code of Ethics. Acts of academic dishonesty impair the academic integrity of AUA and create an unfair academic advantage for the student involved and other member(s) of the academic community. These acts are subject to disciplinary measures as prescribed in the AUA Code of Student Ethics (http://policies.aua.am/policy/10).

AUA Student Code of Conduct
Students are required to conduct themselves in a responsible manner in line with the AUA Student Code of Conduct. Acts that violate the AUA Student Code of Conduct are subject to disciplinary measures as prescribed in the AUA Student Code of Conduct: http://policies.aua.am/policy/101.

Special Needs
Students requiring special accommodations for learning should contact the Student Services Coordinator by the end of the Drop/Add period with such requests.
(http://studentsuccess.aua.am/files/2013/12/Disability-Self-Identification-Form-Fillable.pdf)

"Note: This syllabus is subject to change to accommodate students' need."