CS 370 Computer Architecture  
SDSU Fall 2016 Syllabus

Instructor: Guy Leonard  
Office & Phone: GMCS 540  
Email Address: gleonard@mail.sdsu.edu  
Lecture Time/Place: MW 1730 - 1845 in WC-201  
Office Hours: MW 2030 or by appointment, email anytime

Required Textbook

Course Description
Logic gates, combinational circuits, sequential circuits, memory and bus system, control unit, CPU, exception processing, traps and interrupts, input-output and communication, reduced instruction set computers, use of simulators for analysis and design of computer circuits, and traps/interrupts

Pre-requisites
CS 237 (Machine Organization) or equivalent

Course Objectives/Overview:
1. To learn fundamentals of digital systems analysis and design.
2. To become familiar with components and functional units used in digital computers.
3. To obtain a rudimentary understanding of computer organization and architecture.
4. To acquire some knowledge of the PC hardware and associated peripherals.

Class Format
The class format will be primarily lecture with questions-answers based on the text/readings and slide content. Students will work on software projects and may form small groups to discuss work on the projects.

Attendance
Lecture attendance is strongly recommended but will not be graded on attendance. NOTE - exam material will come from material presented/discussed during class lectures in addition to the textbook and materials posted on Blackboard; the student is responsible for all material presented during lectures.

Blackboard
The syllabus and slides for each lecture will be posted on Blackboard as will additional reading/study/reference material. The scores for your exams and assignments will also be posted on Blackboard. Important class announcements will be posted occasionally on Blackboard as needed.

Evaluation:
You will take three exams and turn-in three (software) projects. The exams will be 60% of your grade; the projects will be 40% of your grade.

1. Exams (3): There will be three (3), 75 minute exams, each one worth 50 points. The
exams will be closed book exams and the dates will be posted on Blackboard. Exams will consist of a mixture of multiple-choice, fill in the blank, and prose response styles of questions.

2. **Projects (3):** You will be assigned three (3) lab (software) projects. You will be assigned to work on and turn-in three projects, each worth 33 points.

3. **Homework/exercises:** Several questions will be assigned at the end of each chapter. These exercises will not be graded; they are intended as a study guide to prepare you for the exams to help you understand the material. Solutions to these exercises will be posted on our class web page and you are highly recommended to self-evaluate your weekly exercises and ask questions if you are stuck on any of them.

**Final Grade**
Your final grade will be calculated by adding all points you earned during the course as follows (I do not use a curve):

- 228 - 249 = A
- 223 - 227 = A-
- 218 - 222 = B+
- 203 - 217 = B
- 198 - 202 = B-
- 193 - 197 = C+
- 179 - 192 = C
- 169 - 178 = C-
- 149 - 168 = D
- 000 - 148 = F

**Make-Up Exam Policy**
Make-up exams may be allowed at the sole discretion of the instructor. Make-up exams will likely be approved for serious, unforeseen events, such as an unexpected illness/injury, legal, or military obligation happens. In these cases, written documentation from the health services provider, court, clergy, or commanding officer will be needed. Requests for make-up exams for other reasons will likely not be approved.

**Plagiarism, Cheating, and Academic Integrity**
All incidents of plagiarism/cheating will be handled according to University Policy:

http://www.sa.sdsu.edu/srr/cheating-plagiarism.html

**Students with Disabilities**
If you are a student with a disability and believe you will need accommodations for this class, it is your responsibility to contact Student Disability Services at (619) 594-6473. To avoid any delay in the receipt of your accommodations, you should contact Student Disability Services as soon as possible. Please note that accommodations are not retroactive, and that I cannot provide accommodations based upon disability until I have received an accommodation letter from Student Disability Services. Your cooperation is appreciated.

**Course Topics / Learning Objectives:**
1. **Introduction (1 lecture)**
   
   - Course & topic overview, Levels of abstraction of digital systems (transistor, gate, register, processor). Design and analysis of digital systems. CAD tools

2. **Data types and representation (2 lectures)**
   
   - Binary coding and conversions between different number systems. Ones, two's complement, sign magnitude representation. Adding, subtracting and multiplication of binary numbers. Fixed and floating-point representation. Error detection and correction (cube representation of binary strings, Hamming code)
3. Boolean algebra and logic design (4 lectures)
Aximatic definition of Boolean algebra. Boolean operators and duality principle Theorems of
Boolean algebra. Boolean functions and expression equivalence Minterms and maxterms
(stanndard forms of Boolean functions). Sixteen binary logic operations . Digital logic gates and
example of full-adder . Design with NAND/NOR gates, complex gates. Custom design (gate
arrays, FPGA)

4. Simplification of Boolean functions (3 lectures)
Karnuagh maps (1,2,3, 4 and 5–variable maps). Selection of prime implicants. Don't care
conditions. Technology mapping for gate arrays (standard to NANMD/NOR schemes, standard
to gate arrays. Timing issues. Static and dynamic hazard.

5. Combinational circuit analysis and design (4 lectures)
Ripple-carry serial adder. Carry-look-ahead generator. CLA adder. Twos' complement adder
comparators. Shifters (including barrel shifters). ROM. Programmable logic arrays. Full adder
with ROM, PLA

6. Sequential circuit analysis and design (4 lectures)
SR-latch (NOR, NAND implementation). Gated SR-latch, Gated D-latch. Flip-flops. Master-
slave FF, Edge triggered FF. FF types (SR, JK, D, T). Design of sequential circuits (analysis

7. Memory organization (2 lectures)
Simple RAM (coincident decoding). Array of RAM chips (extending the address space and the
word size). Push-down stack. FIFO queue. Memory timing . Implementation of error detection
and correction. ROM.

8. Processor (4 lectures)
Register transfer micro operations . Register transfer language. Arithmetic micro operations.
Logic micro operations. ALU and shifter unit. Datapaths. Bus transfer Processor unit . Control
unit (hardwired and micro programmed). Micro programs and micro routines. Instruction set
and addressing modes. Complex instruction set vs. RISC. Reduced instruction set – a 32-bit
example. Data-forwarding and branch prediction.

9. System bus (4 lectures)
Synchronous bus control. Asynchronous bus control. Connecting memory to CPU
Implementation of interrupts. Vectored interrupts and auto-vectored interrupts. Programming
with traps and interrupts