COURSE INFORMATION & SYLLABUS

Term: Fall 1997

Course No. and Title: COMP 362: Computer Architecture

Instructor: Ronald I. Greenberg
Department of Mathematical and Computer Sciences
Loyola University
6525 North Sheridan Road
Chicago, Illinois 60626-5385

Phone: (312)508-3997   **Electronic Mail:** rig@math.luc.edu (This is a good way to reach me.)

Lectures: Mon. and Wed. 4:00–5:15 pm in Damen 342.
If you have to miss a class, get notes from another student and get copies of any handouts. Another student's notes will probably be easier to follow than mine, since mine are typically pieced together from more than one place with a lot of metacomments.

Office Hours: In Damen 330F: Mon. and Wed. 2:00–4:00 pm, and Tues. 12:00-2:00 pm.
These are the guaranteed times to find me except as announced in advance. You should also be able to find me at lots of other times; feel free to look for me or make appointments.

Course Objectives: Provides a basic understanding of digital logic, data representation in computers, computer hardware organization, and the hardware/software interface.

Prerequisites: As specified in the catalog: COMP 211 (Discrete Structures) and 275 (Computer Organization and Assembly Language Programming). In actuality, these requirements are being relaxed in the process of reorganizing undergraduate course offerings in this area and reducing the prerequisites for architecture. While it is anticipated that students will have some background in boolean algebra and Karnaugh maps, there will also be some review of this material.

Required Text: M. Morris Mano. *Computer System Architecture.* Prentice-Hall, third edition, 1993.

Course Requirements: There will be several homework assignments, two midterm exams, and a final. The weightings within the semester grade will be: Homework 20%, Exam I 20%, Exam II 25%, and Final exam 35%.

Homework: Only homework turned in by the due date is guaranteed to be graded. Any special circumstances that cause difficulty in meeting the deadlines should be brought to the attention of the instructor in advance. Homework must be handed in at the beginning of class, since solutions may be handed out in the same class on occasion. *Collaboration* on homework is acceptable, but *copying* is not! You may discuss solution techniques with other students, but you must write up your solutions independently.

Exams: The midterm exams, tentatively scheduled for October 1 and November 5, are 75 minutes long. The final exam is 2 hours long and is scheduled for Tuesday, December 9 at 3:00 pm.

Handouts: Copies of handouts given out in class will be made available on the course web site http://www.math.luc.edu/~rig/courses/c362f97/ in case you have to miss a class.

Tentative Course Outline and Approximate Schedule:

Corresponding sections from the text are shown for each lecture.

1. (8/25) Sections 1.1-1.3: Review of boolean algebra, truth functions, boolean expressions, logic networks, canonical forms.

2. (8/27) 1.4-1.5: Simplification of boolean expressions using Karnaugh-maps, combinational design principles, combinational circuits.

3. (9/3) Section 1.6: Flip-flops, excitation tables.

4. (9/8) Section 1.7: Sequential circuits, finite-state machines.

5. (9/10) Sections 2.1-2.4: Decoders and multiplexers, registers.

6. (9/15) Sections 2.5-2.7: Shift registers and counters, memory organization.

7. (9/17) Sections 3.1-3.3: Base conversions, signed number representations, fixed point representation.

8. (9/22) Sections 3.3-3.6: Fixed point representations continued, floating point representation, codes.

9. (9/24) Sections 4.1-4.3: Register transfer logic, register and bus transfers.

10. (9/29) Sections 4.4-4.7: ALU design.

11. (10/1) Exam 1. Chapters 1–3.

12. (10/6) Sections 5.1-5.2: Instruction codes, computer registers.

13. (10/8) Sections 5.3-5.4: Computer instructions, control unit design.

14. (10/15) Section 5.5: Instruction cycle.

15. (10/20) Section 5.6: Register-reference instructions, memory-reference instructions.

16. (10/22) Section 5.7: I/O, interrupts.

17. (10/27) Sections 5.8-5.9: Complete computer design, control of the basic computer, accumulator design.

18. (10/29) Sections 6.1-6.3: Assembly language programming.

19. (11/3) Section 6.4: Assembler construction.

20. (11/5) Exam 2. Chapters 4–5.

21. (11/10) Sections 6.5-6.7: Loops, simple programming, subroutines.

22. (11/12) Section 6.8: I/O programming.

23. (11/17) Sections 7.1-7.2: Control memory, address sequencing.

24. (11/19) Sections 7.3-7.4: Microprogrammed control unit.

25. (11/24,11/26,12/1) Catchup or additional material.