
<u>Course Number</u>	<u>Course Name</u>	<u>Lecture Hrs</u>	<u>Credit</u>
ELET 2231	Microprocessor Fundamentals	3	3

Course Description:

This course covers application and design of assembly and C language programming for AVR microprocessors. Topics include system timing, bus cycles, interrupts, stacks and subroutines. Upon completion, students should be able to design, program, verify, analyze, and troubleshoot AVR assembly and C language programs.

Textbook:

Morton, John, 2002, AVR: An Introductory Course, 1st Edition, Elsevier ISBN-10: 0-7506-5635-2 (ISBN-13: 978-0-7506-5635-2)

References:

Barnett, R., O'Cull, L., Cox, S., 2007, Embedded C Programming for the Atmel AVR, Thompson-Delmar Learning, ISBN: 1-4180-3959-4

Gadre, Dhananjay V., 2001, Programming and Customizing the AVR Microcontroller, McGraw-Hill, ISBN: 0-07-134666-X

Instructor:

Stephen J. Kuyath, PhD, Associate Professor, Electrical Engineering Technology
Smith 336
sjkuyath@uncc.edu
704.687.4799

Goal:

The goal of this course is for students to be able to design, troubleshoot, simulate, and execute working AVR assembly and C language programs.

Prerequisites/Co-requisites

Prerequisite: Digital Circuits I
Co-requisite: Microprocessor Fundamentals Lab

Prerequisites by Topic:

1. A demonstrated understanding of digital electronics.
 2. A demonstrated understanding of decimal, binary, octal, and hexadecimal number systems.
-

Course Topics:

1. Introduction
2. Microcontroller Fundamentals
3. Microcontroller Architecture
4. The AVR Addressing Modes
5. The AVR Instruction Set
6. Memory Systems, Interrupts, and Timers
7. AVR Development Systems and Development Boards
8. Code Development with C
9. AVR Hardware Design Issues
10. Hardware and software interfacing issues
11. Expanding the AVR's I/O
12. Interfacing A/D and D/A Converters
13. AVR Applications

Outcomes:

1. Demonstrate the ability to design, troubleshoot, and simulate AVR assembly and C language programs.
2. Demonstrate the ability to perform logic and timing simulations using an AVR Development System.
3. Demonstrate an understanding of controlling circuits and systems interfaced to an AVR development board.

Computer Usage:

1. Students will use a development system to write, simulate, and troubleshoot AVR assembly and C language programs
2. Students will download machine code to an AVR development system to control interfaced devices and systems.

Design Component:

Students will design assembly and C language programs for AVR processors.

Examinations, Homework, Projects:

Several homework assignments and projects will be assigned throughout the semester. The instructor may assign some individual assignments, but for most homework assignments and projects, students are encouraged to work together. However, each student is responsible for understanding the course materials as well as submitting individual work. Late submissions for homework and projects will not be accepted.

A midterm exam will be given near the middle of the semester. A final exam will be given during final exam week. If for any reason a student is unable to attend an exam, arrangements with the instructor must be made **prior** to the test. If prior arrangement is not made, the student will receive a grade of zero for the exam.

Grade Calculation:

Homework	5%
Projects	20%
Exam 1	25%
Exam 2	25%
Final Exam	25%
Total	100%

Grade Assignment:

90 – 100%	A
80 – 89%	B
70 – 79%	C
60 – 69%	D
0 – 59%	F

Follow-Up Courses:

1. Digital Systems
2. Microcontroller Systems
3. Embedded Systems

Academic Integrity:

All students are required to *know, observe and enforce* the guidelines set forth by the UNC Charlotte Code of Student Academic Integrity, which may be found the web at <http://www.legal.uncc.edu/policies/ps-105.html>. It should be noted that this Code may be modified and that students are encouraged to contact the Office of the Dean of Students to ensure they consult the most recent version. This code forbids cheating, fabrication or falsification of information, multiple submissions of academic work, plagiarism, abuse of academic materials, and complicity in academic dishonesty. Students should pay

particular attention to the standards involving Complicity and Plagiarism. A complete definition of all terms and a delineation of procedures to be followed for alleged violations may be found in the complete text of the Code at the website referenced above.

For this course, students are strongly encouraged to discuss concepts and techniques; however, with the exception of designated assignments, all material submitted for a grade must be *individual work*. The submission of another student's work, whether written materials or computer simulation results, will not be tolerated under any circumstances. Instances of Complicity will result in the division of the single lowest grade received between all participants. The distribution of the single grade will be at the discretion of the instructor. If evidence of Plagiarism is obtained, the student will be given an 'F' in the course and documented in the student's records as outlined in the Code of Student Academic Integrity.