The University of Texas at Tyler
Department of Electrical Engineering

EENG 3307: Microprocessors (required)

Syllabus

Catalog Description:

Microprocessor architecture, programming and interfacing. Introduction to assembly language programming; Microcomputers, microcontrollers, instruction set, chip interfacing, addressing modes, interrupts, input/output, communication. Three hours of lecture per week with integrated laboratory sessions. **Prerequisites:** EENG 3302 and COSC 1336/1436

Prerequisites:	EENG 3302 - Digital Systems, COSC 1336/1436 – Programming

Credits:	(	2	hours lecture,	3	hours laboratory per week)

Text(s):	Ronald J. Tocci and Frank J. Ambrosio, **Microprocessors and Microcomputers: Hardware and Software, 6th ed**. Prentice Hall, 2003 ISBN: 0-13-060904-8, ISBN-13: 9780130609045
Additional Material:	Motorola 68HC11 Development Board. Laboratory projects are integrated to provide students with hands-on experience.
Course Coordinator:	Melvin Robinson, Assistant Professor

Topics Covered: (paragraph of topics separated by semicolons)

Microcomputer Fundamentals: number systems, codes, digital circuits, memory devices, and introduction to computers; Microprocessors: elements, structure, operation, memory, bus architecture, and instruction set; Microcomputer Programming: assembly language, arithmetic operations, decisions, loops, tables, lists, subroutines, and interrupts; Microcomputer Interfacing: input/output modes, serial and parallel interfaces, synchronous and asynchronous communication.

Evaluation Methods: (only items in dark print apply):

1. Examinations/ Quizzes
2. Homework
3. Report/ Paper
4. Computer Programming
5. Project/ Model
6. Presentation
7. Course Participation
8. Peer Evaluation

Course Learning Outcomes[1]: By the end of this course students will be able to:

1. Solve problems involving conversions between decimal, binary, octal and hexadecimal number systems, signed numbers, arithmetic operations, floating point numbers and representation standards [1]
2. Understand the operation of basic digital systems in the context of microcontroller design including parallel/serial transmission, tri-state logic, clocking, flip-flops and registers, data bus operation [1]
3. Demonstrate knowledge of memory systems including architecture, operation, types, read/write cycles, timing diagrams, applications and techniques to expand word size and capacity [1]
4. Explain the basic operational principles of microprocessors and microcontrollers including architecture, instruction formats, machine language, program and data sections, firmware, step-wise program execution detail and the fetch-decode-execute cycle [1]
5. Design complete and partial address decoding schemes for the microcontroller using memory modules, memory maps, read/write timing and logic components like decoders and tri-state buffers [1]
6. Identify and explain the microcontroller operation from functional block diagrams including: register section, ALU, timing and control, multiplexed buses, pinout, modes of operation and signals [3]
7. Analyze the various types of microcontroller assembly language instructions including addressing modes, processor condition codes, speed of operation and analysis of programs or code segments [3]
8. Outline the operation of an assembler and implement the entire process of writing, compiling, loading and running an assembly language program [3]
9. Illustrate the following concepts and their implementation on the microcontroller: stack operation, interrupt service routines, reset vectors, memory maps, time delay routines [1]

10. Formulate microcontroller input-output solutions utilizing general purpose I/O, interrupts and the timer subsystem [3]
11. List input/output interfacing solutions for issues like voltage mismatch, implementation technology mismatch, power requirements, isolation from electrical loads, and parallel/serial interfacing [1]
12. Implement microcontroller applications using peripherals like the serial interface and the analog-to-digital convertor (ADC) subsystem [3]
13. Incorporate information gained by independent learning from microcontroller technical reference manuals and other sources to implement projects and enhance reports [3]
14. Utilize modern software and hardware tools and techniques to design, debug and test microcontroller based projects using assembly language programming [4]
15. Perform laboratory experiments utilizing microcontroller systems demonstrating combined hardware-software interaction, co-design and debugging [3]
16. Write laboratory reports with experimental results demonstrating visual and written communication skills [3]

[1]Numbers in brackets refer to method(s) used to evaluate the course objective.

Relationship to Program Outcomes (Student Learning Outcomes) [2]: This course supports the following Electrical Engineering Program Outcomes, which state that our students will:

1. have the ability to apply knowledge of the fundamentals of mathematics, science, and engineering; [1, 2, 11]
2. have the ability to use modern engineering tools and techniques in the practice of electrical engineering; [8, 14]
3. have the ability to analyze electrical circuits, devices, and systems; [3, 4, 6, 7, 9]
4. have the ability to design electrical circuits, devices, and systems to meet application requirements; [5, 10]
5. have the ability to design and conduct experiments, and analyze and interpret experimental results; [12, 15]
6. have the ability to identify, formulate, and solve problems in the practice of electrical engineering using appropriate theoretical and experimental methods; [10]
7. have effective written, visual, and oral communication skills; [16]
8. possess an educational background to understand the global context in which engineering is practiced, including:
   a. knowledge of contemporary issues related to science and engineering;
   b. the impact of engineering on society;
   c. the role of ethics in the practice of engineering;
9. have the ability to contribute effectively as members of multi-disciplinary engineering teams;
10. have a recognition of the need for and ability to pursue continued learning throughout their professional careers. [13]

[2]Numbers in brackets refer to course learning outcome(s) that address the Program Outcome.

Contribution to Meeting Professional Component: (in semester hours)

Mathematics and Basic Sciences:		hours
Engineering Sciences and Design:	3	hours
General Education Component:		hours

Grade Replacement:

If you are repeating this course for a grade replacement, you must file an intent to receive grade forgiveness with the registrar by the 12th day of class. Failure to file an intent to use grade forgiveness will result in both the original and repeated grade being used to calculate your overall grade point average. A student will receive grade forgiveness (grade replacement) for only three (undergraduate student) or two (graduate student) course repeats during his/her career at UT Tyler. (2006-08 Catalog, p. 35)

Prepared By:	Mukul V. Shirvaikar	Date:	August 8, 2003
Modified By:	Mukul V. Shirvaikar		August 25, 2004 August 20, 2005 January 11, 2010 January 11, 2012 January 4, 2013 December 29, 2014 January 9, 2017