

SAULT COLLEGE OF APPLIED ARTS & TECHNOLOGY

SAULT STE. MARIE, ONTARIO

COURSE OUTLINE

Course Title: MICROPROCESSOR PROGRAMMING

Code No.: CET127-4 Semester: 2

Program: COMPUTER ENGINEERING TECHNOLOGY

Author: TYCHO BLACK

Date: JAN., 1994 Previous Outline Dated: JAN., 1993

APPROVED:

H. P. Choquet

Dean

94-01-04

Date

MICROPROCESSOR PROGRAMMING
COURSE NAME

CET127-4
CODE NO.

TOTAL CREDIT HOURS: 64

PREREQUISITES: CET105

I. PHILOSOPHY/GOALS:

This course introduces students to 8088 Assembly Language Programming. As a first course in assembly language programming it is necessary to study number systems and the fundamentals of microprocessor architecture in addition to the specific registers, addressing modes and instructions of the 8088 microprocessor. More significant assembly language programming skills will be developed in subsequent courses.

Initially the concentration is on the development of simple instruction sequences using DEBUG, an interactive debugging utility that is available as part of DOS. Later, the use of an Assembler such as Microsofts MASM or Borlands Turbo Assembler will be used to assemble source programs. In addition, the Linker, and Debugging utilities are used to link and test more complex programs.

II. STUDENT PERFORMANCE OBJECTIVES:

Upon successful completion of this course the student will:

1. Be able to manipulate and convert numbers in binary, hexadecimal, decimal and octal number systems.
2. Describe the architecture of the 8088 microprocessor including its registers, memory organization, the stack and internal organization.
3. Be able to use the 8088 addressing modes effectively.
4. Demonstrate the use of DEBUG to assemble, load, save, unassemble, single step, examine and troubleshoot simple instruction sequences.
5. Be able to write assembly language programs utilizing the subset of commonly-used 8088 instructions to solve simple problems.
6. Demonstrate proficiency in developing and troubleshooting assembly language programs utilizing an Assembler.

III. TOPICS TO BE COVERED:

1. Number Systems: Binary, Octal, Hexadecimal and Decimal.
2. 8088 Microprocessor Organization and Registers.
3. 8088 Addressing Modes.
4. The use of DEBUG to assemble and troubleshoot simple programs.
5. Instruction Set of the 8088 Microprocessor.
6. Writing Assembly Language programs to solve problems.
7. Using an Assembler as a Program Development Tool.

Note: The following Learning Activities may not be covered in the exact order specified below.

IV. LEARNING ACTIVITIES

REQUIRED RESOURCES
TEXTBOOK:

BLOCK 1: INTRODUCTION AND NUMBER SYSTEMS

1. Describe a general Microcomputer System Architecture.
2. Describe the Bus organization in typical microcomputer systems and the execution sequence of an instruction.
4. Describe Assembly language programming as to its nature, advantages and disadvantages.
5. Be able to manipulate numbers in binary, hexadecimal, octal and decimal number systems and perform conversions between them.
6. Be able to utilize two's complement numbers to represent negative numbers and perform addition and subtraction of signed binary and hexadecimal numbers.
7. Understand the nature of BCD and ASCII codes and Parity checking.

"ASSEMBLY LANGUAGE FOR THE IBM PC"

by Kip Irvine
(Macmillan)

CHAP. 1,2

BLOCK 2: SOFTWARE ARCHITECTURE OF THE 8088 MICROPROCESSOR

1. Describe the Software model of the 8088 microprocessor.
2. Describe the 8088 memory address space and understand its use of memory segmentation.
3. Describe the nature and functioning of the 8088 internal registers.
4. Be able to distinguish between physical

CHAP. 2

- and logical addresses.
5. Describe the operation of the stack in 8088 systems.
 6. Be able to describe the I/O address space in 8088 systems.
 7. Be able to describe and use the 8088 addressing modes in simple programs.

BLOCK 3: MACHINE LANGUAGE CODING AND THE USE OF "DEBUG"

1. Be aware of the nature of machine code and the method of converting assembly language instructions to machine language.
2. Demonstrate the use of DEBUG on the PC to examine and modify the contents of memory and registers, to load and save machine code programs on disk, to Assemble and Unassemble instructions, to execute and single-step through programs and other tasks required in program debugging.

APPENDIX B

BLOCK 4: ASSEMBLY LANGUAGE PROGRAM DEVELOPMENT

1. Be able to develop programs on the PC using MASM (Macro Assembler) or TASM (Borland Assembler) and be able to use proper statement syntax, program initialization and pseudo operations (pseudo-ops).
2. Be able to create assembler source files using an editor.
3. Be able to generate run modules using MASM and LINK and then execute and troubleshoot assembler programs with CODEVIEW (or alternately using TASM and Borlands Debugger).

CHAP. 3,4

BLOCK 5: 8088 MICROPROCESSOR PROGRAMMING

1. The student will demonstrate the operation of the following groups of instructions and their proper utilization in programs:
 - a) Data transfer instructions
 - b) Arithmetic instructions
 - c) Logic instructions

MICROPROCESSOR PROGRAMMING

CET127-4

COURSE NAME

CODE NO.

- d) Shift and rotate instructions
 - e) Flag-control instructions
 - f) Compare instructions
 - g) Jump and loop instructions
 - h) Subroutine handling
 - i) String instructions
2. Students will demonstrate the ability to solve assigned problems using the instructions and addressing modes most appropriate to the task.
 3. Students will demonstrate the proper use of DOS INT 21h services for basic Input Output services.

CHAP. 5, 6, 7

V. METHOD OF EVALUATION:

3 THEORY TESTS (20% each)	60%
ASSIGNMENTS	25%
QUIZZES AND PRACTICAL TESTS	15%

(The percentages shown above may have to be adjusted to accurately evaluate student skills. Students will be notified of any changes made.)

GRADING SCHEME

A+	90	-	100%
A	80	-	89%
B	70	-	79%
C	55	-	69%
I	Incomplete		
R	Repeat		

TESTS

MICROPROCESSOR PROGRAMMING
COURSE NAME

CET127-4
CODE NO.

Written tests will be announced about one week in advance. Quizzes may be conducted without advance warning. No "re-write" opportunities exist for quizzes not written.

UPGRADING OF INCOMPLETES

When a student's course work is incomplete or final grade is below 55%, there is the possibility of upgrading to a pass when a student meets the following criteria:

1. The student's attendance has been satisfactory.
2. An overall average of at least 40% has been achieved.
3. The student has not had a failing grade in all of the theory tests taken.
4. The student has made reasonable efforts to participate in class and complete assignments.

ASSIGNMENTS AND LAB ACTIVITIES:

Lab activities and assignments represent a very important component of this course in which practical 'hands-on' skills will be developed. Because of this, lab attendance is mandatory and the satisfactory completion of all assignments is required. It is the student's responsibility to discuss absences from regularly scheduled labs with the instructor so that alternate arrangements (where possible) can be made to complete the lab requirements.

A penalty for late assignments will be applied unless there are extenuating circumstances. A 10% per week penalty will be applied. After 4 weeks late assignments will not be accepted for credit.

It is acceptable that students consult with each other in relation to their assigned problems. However, it is unacceptable to copy programs written by someone else and submit them as your own work. Where plagiarism or copying is found and it is impossible to determine whose original work it is, a mark of zero will be assigned to all assignments involved.

ATTENDANCE:

Absenteeism will affect a student's ability to succeed in this course. Absences due to medical or other unavoidable circumstances should be discussed with the instructor, so that remedial activities can be scheduled. A Quiz or Test missed because of an unauthorized absence will result in a zero grade being assigned.

MICROPROCESSOR PROGRAMMING
COURSE NAME

CET127-4
CODE NO.

VI. REQUIRED STUDENT RESOURCES:

1. TEXTBOOK: "ASSEMBLY LANGUAGE FOR THE IBM-PC"
by Kip Irvine
(Macmillan)
2. 8088 INSTRUCTION SET SUMMARY.

VII. ADDITIONAL RESOURCE MATERIALS AVAILABLE:

VIII. SPECIAL NOTES:

Students with special needs (eg. physical limitations, visual or hearing impairments, or learning disabilities) are encouraged to discuss any required accommodations confidentially with the instructor.

Your instructor reserves the right to modify the course as deemed necessary to meet the needs of students or take advantage of new or different learning opportunities.