

SAULT COLLEGE OF APPLIED ARTS & TECHNOLOGY

SAULT STE. MARIE, ONTARIO

COURSE OUTLINE

**COURSE OUTLINE: INTRODUCTION TO
MICROPROCESSOR PROGRAMMING**

CODE NO.: CET 228-4

PROGRAM: ELECTRICAL/ELECTRONIC TECHNICIAN

SEMESTER: FOUR

DATE: JANUARY 1997

**PREVIOUS
OUTLINE DATED: SEPTEMBER 1995**

AUTHOR: ENO LUDAVICIUS

NEW: ___ REV.: X ___

APPROVED: *Bill Armstrong*
COORDINATOR

DATE

L. DeSousa
DEAN

Mar 24/98

DATE

INTRODUCTION TO
MICROPROCESSOR PROGRAMMING
COURSE NAME

CET228-4
CODE NUMBER

TOTAL CREDIT HOURS: 64

PREREQUISITE(S): ELN 230 - 3

PHILOSOPHY/GOALS:

THE COURSE INTRODUCES STUDENTS TO 80x86 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 80x86 MICRO. ASSEMBLY LANGUAGE PROGRAMMING SKILLS WILL BE DEVELOPED TO ENHANCE COMPUTER SYSTEM HARDWARE AND SOFTWARE CONCEPTS.

STUDENT PERFORMANCE OBJECTIVES:

UPON SUCCESSFUL COMPLETION OF THIS COURSE, THE STUDENT WILL BE ABLE TO:

- 1) MANIPULATE AND CONVERT NUMBERS IN BINARY, HEXADECIMAL, DECIMAL AND OCTAL NUMBER SYSTEMS.
- 2) DESCRIBE THE ARCHITECTURE OF THE 80x86 MICROPROCESSOR INCLUDING ITS REGISTERS, MEMORY ORGANIZATION, THE STACK AND INTERNAL ORGANIZATION.
- 3) OPERATE WITH THE 80x86 ADDRESSING MODES EFFECTIVELY.
- 4) DEMONSTRATE THE USE OF DEBUG TO ASSEMBLE, LOAD, SAVE, UNASSEMBLED, SINGLE STEP, EXAMINE AND TROUBLESHOOT SIMPLE INSTRUCTIONS.
- 5) WRITE ASSEMBLY LANGUAGE PROGRAMS UTILIZING THE SUBSET OF COMMONLY-USED 80x86 INSTRUCTIONS TO SOLVE SIMPLE PROBLEMS.
- 6) DEMONSTRATE PROFICIENCY IN DEVELOPING AND TROUBLESHOOTING ASSEMBLY LANGUAGE PROGRAMS UTILIZING MICROSOFT ASSEMBLER.

**INTRODUCTION TO
MICROPROCESSOR PROGRAMMING
COURSE NAME**

**CET228-4
CODE NUMBER**

TOPICS TO BE COVERED:

- 1) NUMBERS SYSTEMS: BINARY, OCTAL, HEXADECIMAL AND DECIMAL.
- 2) HARDWARE CONCEPTS: 80x86 MICROPROCESSOR ORGANIZATION AND REGISTERS INCLUDING DIFFERENCES BETWEEN 8088,8086,80286, 80386, 80486, 80586 AND PENTIUM PROCESSORS.
- 3) SOFTWARE CONCEPTS: 80x86 ADDRESSING MODES
- 4) THE USE OF DEBUG TO ASSEMBLE AND TROUBLESHOOT SIMPLE PROGRAMS.
- 5) INSTRUCTION SET OF THE 80x86 MICROPROCESSOR.
- 6) WRITING ASSEMBLY LANGUAGE PROGRAMS TO SOLVE PROBLEMS.
- 7) USING MICROSOFT ASSEMBLER AS A PROGRAM DEVELOPMENT ENVIRONMENT.

LEARNING OUTCOMES

REQUIRED RESOURCES

1.0) <u>INTRODUCTION TO ASSEMBLY LANGUAGE</u>	TEXTBOOK: THE 80x86 IBM PC & COMPATIBLE COMPUTERS
1.1) DESCRIBE WHAT ASSEMBLY LANGUAGE IS AND WHY LEARNING ASSEMBLY IS IMPORTANT.	CHAPTER #1 - SECTION 1.3
1.2) BE ABLE TO MANIPULATE NUMBERS IN BINARY, HEXADECIMAL, OCTAL AND DECIMAL NUMBER SYSTEMS AND PERFORM CONVERSIONS BETWEEN THEM.	CHAPTER #0 - SECTION 0.1
1.3) BE ABLE TO USE DEBUG TO RUN SAMPLE PROGRAMS.	CHAPTER #1 - SECTION 1.3 APPENDIX A - DEBUG
1.4) UNDERSTAND BASIC PROGRAM SYNTAX AND STRUCTURE.	CHAPTER #1 - SECTION 1.4 - SECTION 1.5
1.5) BE ABLE TO EXECUTE FIRST ASSEMBLY PROGRAM CALLED HELLO	CHAPTER #2 - SECTION 2.1 - SECTION 2.2
2.0) <u>HARDWARE AND SOFTWARE ARCHITECTURE</u>	
2.1) INTRODUCTION TO MICROCOMPUTER HARDWARE COMPONENTS AND INTEL FAMILY OVERVIEW.	CHAPTER #0 - SECTION 0.2 CHAPTER #1 - SECTION 1.1 CHAPTER #9 - SECTION 9.1
2.2) BE ABLE TO DESCRIBE SYSTEM ARCHITECTURE.	CHAPTER #1 - SECTION 1.1 - SECTION 1.2
2.3) DISCUSS SYSTEM SOFTWARE AND MEMORY.	CHAPTER #1 - SECTION 1.4 CHAPTER #11- SECTION 11.1
2.4) UTILIZING DEBUG TO RUN SIMPLE PROGRAMS TO VIEW SYSTEM ARCH.	CHAPTER #2 - SECTION 2.3 APPENDIX A - DEBUG

INTRODUCTION TO
MICROPROCESSOR PROGRAMMING
COURSE NAME

CET228-4
CODE NUMBER

<u>3.0) ASSEMBLY LANGUAGE FUNDAMENTALS</u>	<u>REQUIRED RESOURCES</u>
3.1) DESCRIBE DATA DEFINITION DIRECTIVES AND APPLY THEM TO DATA TRANSFER INSTRUCTIONS.	CHAPTER #1 - SECTION 1.4 CHAPTER #2 - SECTION 2.5
3.2) INTRODUCTION TO BASIC INTEGER ARITHMETIC INSTRUCTIONS.	CHAPTER #3 - SECTION 3.1 - SECTIONS 3.2 & 3.3
3.3) OVERVIEW OF THE FIVE TYPES OF ADDRESSING MODES USED TO LOCATE MEMORY LOCATIONS.	CHAPTER #1 - SECTION 1.6
3.4) DISCUSS PROGRAM STRUCTURE INVOLVING STANDARD MEMORY MODEL.	CHAPTER #2 - SECTION 2.1 - SECTION 2.2 CHAPTER #11- SECTION 11.2
<u>4.0) THE MACRO ASSEMBLER: THE ASSEMBLY PROCESS</u>	
4.1) DISCUSS THE ASSEMBLE-LINK-EXECUTE CYCLE.	CHAPTER #2 - SECTION 2.2
4.2) UNDERSTANDING THE RELATED FILES THAT ARE CREATED DURING THE ASSEMBLY PROCESS.	
4.3) DISCUSS ASSEMBLY SYMBOLIC NAMES, OPERATORS, EXPRESSIONS AND TRANSFER OF CONTROL.	CHAPTER #2 - SECTION 2.4
4.4) DISCUSS THE UTILIZATION OF THE 80386/80486 PROCESSORS.	CHAPTER #8 - SECTION 8.1 CHAPTER #8 - SECTION 8.2
4.5) TROUBLESHOOTING PROGRAMS WITH DEBUG.	APPENDIX A - DEBUG PROGRAMMING
<u>5.0) INPUT AND OUTPUT: SERVICES</u>	
5.1) DISCUSS THE USE ASSEMBLY SUBROUTINES FOR I/O SERVICES.	CHAPTER #2 - SECTION 2.4 CHAPTER #12- SECTION 12.1
5.2) DEFINE AND DISCUSS HARDWARE AND SOFTWARE INTERRUPTS.	CHAPTER #4 - SECTION 4.1
5.3) DISCUSS DOS FUNCTION CALLS FOR CONSOLE AND KEYBOARD INPUT AND OUTPUT.	CHAPTER #14- SECTION 14.1
5.4) DISCUSS THE UTILIZATION OF BIOS-LEVEL VIDEO CONTROL.	CHAPTER #4 - SECTION 4.2
5.5) TROUBLESHOOTING INPUT/OUTPUT PROGRAMS WITH DEBUG.	CHAPTER #18- SECTION 18.2 CHAPTER #4 - SECTION 4.1 CHAPTER #16- SECTION 16.3 APPENDIX A - DEBUG PROGRAMMING

INTRODUCTION TO
MICROPROCESSOR PROGRAMMING
COURSE NAME

CET228-4
CODE NUMBER

<u>6.0) CONDITIONAL PROCESSING</u> <u>(BRANCHING WITH CONDITIONS)</u>	<u>REQUIRED RESOURCES</u>
6.1) DESCRIBE USE OF BOOLEAN AND COMPARISON INSTRUCTIONS TO MANIPULATE DATA BITWISE.	CHAPTER #3 - SECTION 3.3
6.2) INTRODUCTION TO CONDITIONAL JUMPS IN ASSEMBLY STRUCTURES.	
6.3) INTRODUCTION TO CONDITIONAL LOOPS IN ASSEMBLY STRUCTURES.	
6.4) DISCUSS HIGH-LEVEL LOGIC STRUCTURES IN ASSEMBLY LANGUAGE.	CHAPTER #7 - SECTION 1.1
<u>7.0) ARITHMETIC IN ASSEMBLY</u> <u>LANGUAGE</u>	
7.1) DISCUSS MULTIPLICATION AND DIVISION BY SHIFTING AND ROTATING BITS.	CHAPTER #3 - SECTION 3.5
7.2) UNDERSTANDING SAMPLE EXAMPLES USING SHIFTS AND ROTATES. ie. DISPLAY A NUMBER IN ASCII	
7.3) DISCUSS THE DIFFERENCES BETWEEN SIGNED AND UNSIGNED ARITHMETIC.	CHAPTER #6 - SECTION 6.1
7.4) DISCUSS THE UTILIZATION OF STACK PARAMETERS.	CHAPTER #2 - SECTION 2.1 CHAPTER #2 - SECTION 2.3
<u>8.0) 825/54 TIMER AND MUSIC</u>	
8.1) DISCUSS THE TIMER DESCRIPTION AND INITIALIZATION	CHAPTER #13 - SECTION 13.1
8.2) CREATING MUSIC WITH TIMING DELAYS.	CHAPTER #13 - SECTION 13.2 - SECTION 13.3
<u>9.0) BASICS OF SERIAL COMMUNICATION</u>	
9.1) SERIAL VS. PARALLEL DATA TRANSFER.	CHAPTER #17 SECTION 17.1
9.2) RS232 AND OTHER SERIAL I/O STANDARDS.	CHAPTER #17 SECTION 17.1

INTRODUCTION TO
MICROPROCESSOR PROGRAMMING
COURSE NAME

CET228-4
CODE NUMBER

REQUIRED STUDENT RESOURCES

THE 80x86 IBM PC & COMPATIBLE COMPUTERS, VOLUME I&II
ASSEMBLY LANGUAGE, DESIGN AND INTERFACING
By: M.A. MAZIDI, J.GILLISPIE MAZIDI

ADDITIONAL RESOURCE MATERIALS

- 1) A.SINGH & A.TRIBEL, IBM PC/8088 ASSEMBLY LANGUAGE PROGRAMMING
PRENTICE HALL
- 2) J.GODFREY, IBM MICROCOMPUTER ASSEMBLY LANGUAGE
BEGINNING TO ADVANCED
PRENCTICE HALL, 1989
- 3) J. PEROZZO, ASSEMBLING AND TROUBLESHOOTING MICROCOMPUTERS
DELMAR, 1991

METHOD(S) OF EVALUATION

THE FINAL GRADE OF THIS COURSE WILL BE DIVIDED BETWEEN
THE THEORY (60%), AND LABWORK(40%).
EACH UNIT OF THE COURSE WILL BE INDEPENDENTLY ASSESSED,
AND EACH MUST BE SUCCESSFULLY COMPLETED TO COMPLETE THE
COURSE.

THE FINAL GRADE FOR WILL BE DERIVED FROM THE RESULTS OF
TWO INSTRUCTOR ASSIGNED TESTS, ONE PRACTICAL TEST
AND BEST EIGHT OUT OF TEN LAB ASSIGNMENTS.

T1 - FEB 26/97
WED. 11:30 → 12:30

TWO TESTS	50%	(25% PER TEST)
PRACTICAL	10%	
ASSIGNMENTS	40%	(5% PER ASSIGNMENT)

TOTAL 100%

THE GRADING SYSTEM USED WILL BE AS FOLLOWS:

A+	>= 90%	CONSISTENTLY OUTSTANDING ACHIEVEMENT
A	80-89%	EXCELLENT ACHIEVEMENT
B	70-79%	ABOVE AVERAGE ACHIEVEMENT
C	55-69%	SATISFACTORY ACHIEVEMENT
R		REPEAT
X		INCOMPLETE

NOTE: THERE ARE NO REWRITES IN THIS COURSE!