SAULT COLLEGE OF APPLIED ARTS & TECHNOLOGY

SAULT STE. MARIE, ONTARIO

COURSE OUTLINE

Course Title: COMPUTER SYSTEMS

Code No.: CET 200 - 6

Program: COMPUTER ENGINEERING TECHNOLOGY

Semester: THREE

Date: JULY, 1985

Author: PETER SAVICH

New: _____ Revision: X

APPROVED: A P Chrzuth 85/07/07

Chairperson Date

Peter Savich

CET 200

COMPUTER SYSTEMS 1

OBJECTIVES

General Objectives

The objectives of this course are to provide the student with a knowledge of the organization of the PDP 11 family of computers and skill in the use of the machine and assembly languages. The student will become capable of using both the RT11 and RSX11 operating systems, and will be capable of programming a variety of peripherals including the terminal, A/D and D/A converters, real-time clock, and digital I-O. This course is designed to be a prerequisite for Computer Systems 2, Data Communications, and Interfacing.

TEXT

PROGRAMMING IN ASSEMBLY LANGUAGE: MACRO-11
Edward F. Sowell

References

RSX-11M Guide to Program Development
Digital Equipment Corporation

PDP-11 MACRO-11
Language Reference Manual

IAS/RSX-11
I/O Operations Reference Manual

IAS/RSX-11
ODT Reference Manual

BLOCK 1. Machine Language Programming 3 weeks

At the end of this block the student shall be able to:

1. Discuss the organization of the PDP-11 family of computers.
2. Describe the format of the various machine language instructions, and convert programs from mnemonics to octal.
3. Describe the operation and use of the different addressing modes available on the PDP11.
4. Write programs and use the RSX11 operating system to solve a variety of problems of complexity similar to those in chapters 1 through 7 of the text.
5. Use the ODT debugging program to debug programs using the following tecniques:
   a. Link ODT with the program to be tested.
   b. Examine and modify memory and registers.
   c. Set breakpoints.
   d. Utilize the relocation registers.
   e. Single-step.

2. Use the FCS macros to write programs utilizing data transfer between files, the terminal, and the program.
3. Write MACRO programs using the FCS that demonstrate the following capabilities:
    1. Read and process each record of a file, using the end of file error IE.EOF to recognize the end of file.
    2. Separate the values of different variables within a record when the individual values are separated by a specified delimiter.
    3. Move strings of ASCII character from one area in memory to another
    4. Convert ASCII strings representing OCTAL or DECIMAL numbers to binary.
    5. Convert binary numbers to ASCII strings representing their OCTAL or DECIMAL values.
    6. Form composite output records for display according to a specified form.

BLOCK 4 RT11 PROGRAMMING 3 weeks

At the end of this block the student shall be able to:

1. Describe the use of CONTROL and STATUS registers to perform I/O operations on the PDP11.
2. Describe the principles of serial data transmission in computer systems.
3. Describe the operation of the ACIA in M6800 systems.
4. Use standard programming techniques to program the PDP11 console and printer in non-interrupt mode.
5. Describe the process of preparing a diskette for use with the RT11 operating system.
6. Discuss the function of, and use in typical problems, the following RT11 macros:
    1. .TTIN
    2. .TTOUT
    3. .PRINT
    4. .GTLIN
7. Write and run programs under RT11 of complexity similar to those written for RSX11.

BLOCK 5. INPUT-OUTPUT OPERATIONS 3 weeks

At the end of this block the student will be able to:

1. Describe the organization of the following I/O devices:
    the terminal
    the A/D converter
    the D/A converter
    the real-time clock
    the digital I/O unit
4. Write MACRO-11 programs to use the above I/O devices with RT11 in the non-interrupt mode.

ASSESSMENT

The student will be assessed in this course through a series of written tests, practical assessments, quizzes, and

Peter Savich

CET 200

COMPUTER SYSTEMS I

PHILOSOPHY/ GOALS

The objectives of this course are to provide an introductory level instructional package for the student into computer systems. The student will gain both theoretical knowledge of how computers work and the practical experience with computers in the laborotory. The knowledge and experience learned in this course will be applied to the second level computer systems course were more complex computer configurations and concepts are introduced. Subsequent advanced courses will provide the student with even more knowledge and more experience with other computer systems.

METHOD OF ACHIEVING GOALS

The engineering technology department has exclusively available for its students the PDP-11/73 computer from Digital Equipment Corporation. Up to 25 terminals (either VT52 or VT100) can be configured to this computer. The Mini-VAX computer can also be made available to students but is intended for use in the more advanced courses.

In the laborotory and in the class the primary focus of study will be on the multi-user, multi-tasking operating system supplied by Digital called RSX-11. Other operating systems and systems architectures will be discussed only in a relational context to that of the RSX-11 operating system. The computer operates on a binary based machine code for all its functions. The lowest level readily intelligible to human operators is the mneumonically based low level language called assembly. The PDP 11 family of computers has the reknown assembly language version called Macro-11. This Macro-11 language will be used in writting all the application programs the students will need for the course.

Since classroom facilities exist in the laborotory rooms, lectures will be given during scheduled lab times to facilitate the course objectives being delivered.

EVALUATION PROCEDURES AND ATTENDANCE REQUIREMENTS

The student will be assesssed through a series of written tests, practical demonstrations, quizzes, attendance, and programming assignments.

All tests and assignments will be conducted on a timely basis with one week notice.

All programming assignments submitted past the due date will be accepted but 50% of mark earned will be recorded.

All assignments will be weighted to equal value each except the File Control Services programming assignments which are double weighted.

All quizzes will be equivalent weight to that of assignments and may not be given any advance notice.

Attendance in all labs and classes is expected.
A bonus program of up to 10% of final mark will be given to any student with a perfect record of attendance. A student with 80% attendance record or less will not receive any bonus.

RECORD OF ATTENDANCE	BONUS
less than 80%	0%
81% - 85%	2%
86% - 90%	5%
91% - 95%	7%
96% - 100%	10%

Re-writes will be available for the written tests. However, regardless of mark earned on the re-write (assuming mark is above 65%) a mark of 65% will be recorded for that particular test.

The final mark will be calculated using the following formula that uses the raw marks recorded from the assessment techniques described earlier.

written tests	50%
programming assignments	30%
quizzes	10%
practical demonstrations	10%
FINAL MARK	100%
attendance BONUS	10%

The grade system to be used is as follows.

PERCENTAGE MARK	GRADE MARK
80 - 100%	A
66 - 79%	B
55 - 65%	C
below 55%	F

The final grades must be submitted by Jan 3/87. Students must have a valid reason in order to receive an "X" grade if any work remains uncompleted at this deadline. DO NOT assume you will be given the "X" grade unless you have been told specifically.

COURSE OUTLINE

TEXT

Programming In Assembly Language: Macro-11
by Edward F. Sowell

BLOCK 1. System Architecture and Machine Language Programming

At the end of this block the student should be capable of:

1. Describing the hardware compenents of a computer system

2. Discussing how a computer works and specifically the PDP 11 family of computers.

3. Understanding the differences between various operating systems and specifically the RSX-11 Op/sys.

4. Programming simple instructions in assembly language Macro-11

5. Editing, assembling, linking, running, debugging simple programs.

6. Converting from mneumonics to machine code octal instuctions.

BLOCK 2. Subroutines, Macros and Libraries for Programming

At the end of this block the student should gain some valuable experience in the following areas:

1. Using subroutines and macros to modularize more complex programs.

2. Linking and mapping of seperate files for running of programs.

3. Using the librarian to create and maintain a library of routines.

4. Using the on line debugging routine called ODT.

BLOCK 3. File Handling

At the end of this block the student should have experience with the following proprietary software:

1. Describing the principles of file control.

2. Using the FCS macros to perform Input and Output operations.

3. Using the FCS supplied macros to write programs that manipulate files that contain data that must be transformed or processed before transfer to the terminal, printer, or another file.

BLOCK 4. Floating Point Numbers

At the end of this block the student should be capable of:

1. Converting between decimal representation of fractions and integers into the floating point representation and vice versa.

2. Using the floating point instruction set in a programming environment.