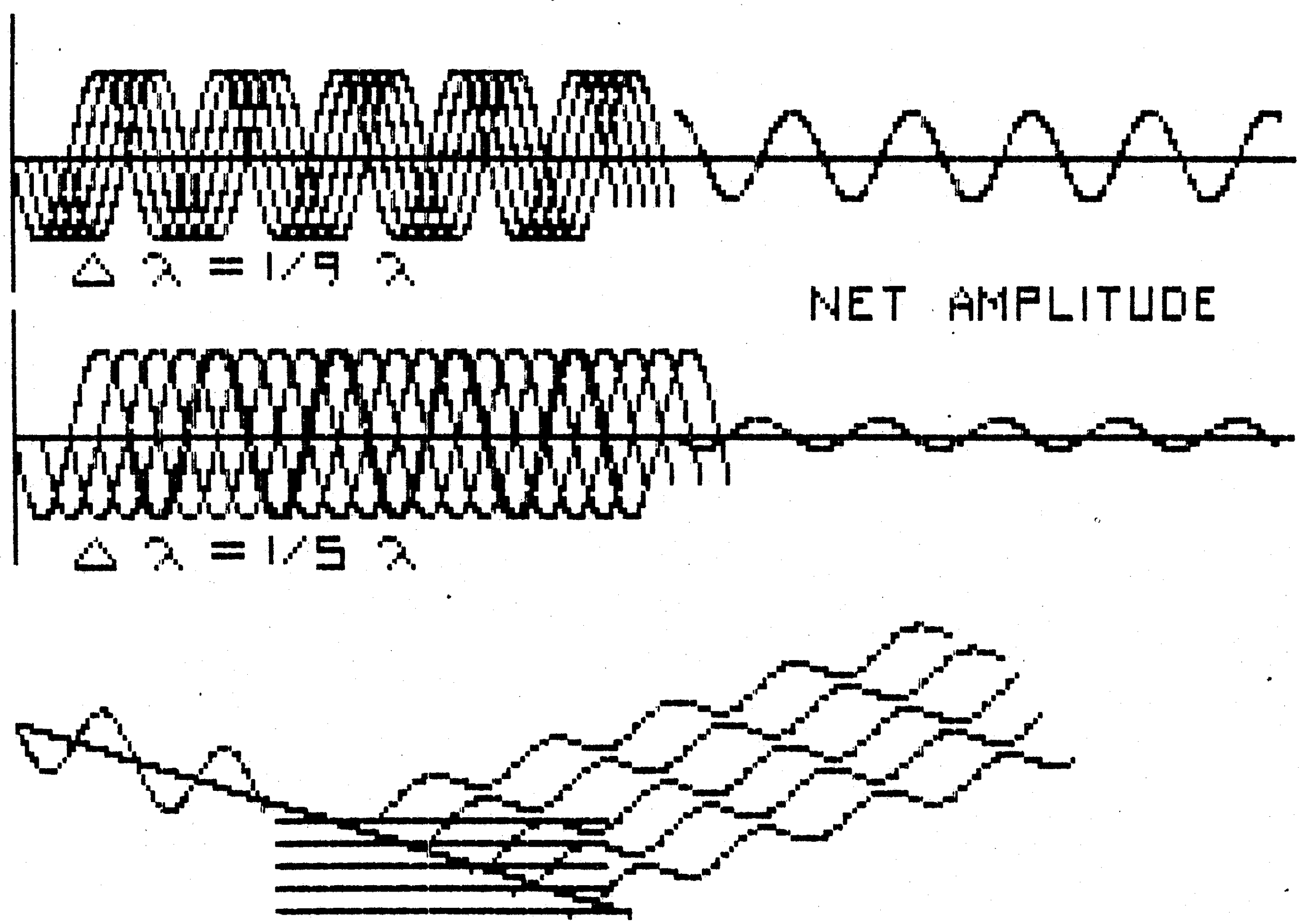
Computers in Chemical Education Newsletter

Published by the ACS Division of Chemical Education
Committee on Computers in Chemical Education
Paul Cauchon, Chairman/Canterbury School, New Milford, CT 06776
Donald Rosenthal, Editor/Department of Chemistry, Clarkson University, Potsdam, NY 13676

Vol. VIII Number 3 **September 1985**



ON THE COVER

The cover on this issue of the Newsletter is entitled "Bragg Diffraction from Parallel Planes". It won honorable mention in the Computer Graphics Contest at the Eight Biennial Conference on Chemical Education. The figure was submitted by Professor S. Z. Lewin of New York University. The figure was produced by a 90 line Basic program written for an Apple IIe with 64 Kbytes of core and was printed using an Epson FX-BO printer.

Professor Lewin explains that the program permits the choice of the difference in path length between successive diffracting planes (as in a crystal or grating), as well as a choice of the total number of diffracting planes or lines. He has found this useful in demonstrating (1) the way in which the intensity of a diffraction line falls off with departure from the Bragg angle (2) the way line broadening accompanies a reduction in particle size of a powder and (3) the relationship between grating size and spacing, and spectral resolution in a spectrometer.

MESSAGE FROM THE CHAIRMAN

Last April a survey concerning the use of computers in general/freshman chemistry courses was conducted via a questionnaire mailed to approximately 800 chemistry department chairs at two and four year colleges. The primary purpose was to gather information to be used in planning the schedule and structure of the national CCCE workshops which have been conducted on a biennial basis for the past 12 years. Almost 300 questionnaires were returned with the largest portion of the respondees (over 50%) indicating a need for orientation in techniques of interfacing microcomputers to laboratory instruments. As a result of this survey, plus feedback from the highly successful '85 workshops at Potsdam and Reno, we are planning three-day intensive workshops on Interfacing and CAI at Montana State University immediately after the 9th BCCE next summer. Details will appear in this newsletter and future mailings from Bozeman.

The survey also provided interesting figures concerning how extensively computers have found their way into chemistry courses at the freshman level. In short, not very far! Five types of interactive, i.e. student-oriented applications were identified: drill and practice, tutorial CAI, stimulations, interfacing and data reduction. Of these, drill and practice was reported as being used "regularly" at only 25% of the institutions, with other categories showing significantly lower "frequent" usage. 46% indicated no use of any sort by anyone in the department. On the other hand, there were generally very positive responses to a question about the effectiveness of computer-based learning. Therefore, the reasons for the relatively low usage are apparently more logistical than pedagogital.

The situation at the high school level seems to be at least as bad, if not worse - worse, that is, if you agree that computers SHOULD be used throughout all levels and activities in chemical education (As John Moore frequently says: "If your chemistry students are not learning with and about computers they are being shortchanged."). In the November issue of School Science and Mathematics, Jeffrey Lehman reports on a survey of computer usage in high school science courses conducted during April of 1984. He found 77% of the teachers surveyed make no use of computers and only 6% use them on a regular basis. On an institutional basis, in 41% of the schools not a single teacher used microcomputers in class. Dr. Lehman suggests as possible causes for the low usage: inaccessibility of hardware, lack of training and relatively high quality software.

Although the logistical problems cannot be denied, significant progress has been made in the past couple of years in terms of software development and teacher training. The CCCE/SERAPHIM workshops reached approximately 750 teachers in the past year alone, and the software packages traveling with these workshops continue to improve in both quality and variety. There is another factor, however, that must be considered when trying to account for the relatively slow rate at which computer usage is really taking hold. Glenn Crosby points to it in his thought-provoking article in the September issue of the Journal of Chemical Education. In Glenn's words, "The promise of computers for improving educational delivery is clear and undeniable." He then points out the challenge to us - the chemical educators - to learn how to use computers in ways that are not only pedagogically equal or superior, but also labor-saving for the individual instructor. And this, I believe, is the key, for as Glenn observes: "When an educational innovation, no matter how effective, requires substantially more effort by the instructor than the standard chalkboard or overhead projector, it is doomed to eventual extinction." I think that at the present time, many of our colleagues who do not use a computer in their courses may very well be convinced of its intrinsic value as a pedagogical tool, but they also see it as one which will demand considerable extra time and effort, if it is to be used on a regular, practical basis. Our real challenge is to dispel this attitude. Otherwise, the impact of computers on chemical education may never progress beyond that relatively small percentage of dedicated teachers in whose classes computerbased activities and assignments are as commonplace as the text, the laboratory, and the chalkboard.

COMMENTS FROM THE EDITOR

The two Seventh C.C.C.E National Computer Workshops held last summer were a great success. In this issue of the Newsletter George Sheats reports on the Workshops held in Potsdam. The overall evaluation by participants was very favorable. George helped to organize and ran the software exhibit and exchange. Harry Pence organized a computer book exhibit. In handling the registrations it became very obvious that there is considerable need for affordable workshops on computer interfacing. At an early stage, almost half the registrants opted for the one on interfacing. By April we had to close registration for the workshop at 24. I believe we could have obtained 50 registrants for this workshop even in a place as remote as Potsdam, NY. The Eighth C.C.C.E. National Computer Workshops will be held in Bozeman, Montana this summer and will include a microcomputer interfacing component. There is a genuine need for many more workshops of this type. We would be pleased to publicize any such workshops in the Newsletter. Just send the information to me.

The spring ACS Meeting in New York City (April 13 to 18) will contain many papers of interest to high school, college and university teachers. The preliminary program will be published in Chemical and Engineering News in mid-January and the final program will be published in February.

Professor Raymond Dessy (a member of the C.C.C.E.) will receive the ACS Award for Computers in Chemistry at the meeting. A one day symposium in his honor will be sponsored by the Division of Computers in Chemistry tentatively on Wednesday, April 16. On Thursday a symposium on Applications Software for Lecture and Laboratory Courses is planned for the Division of Chemical Education. This all day symposium will consider eight areas in which substantial commercially available applications software exist - word processing (D. Rosenthal - Clarkson University), software for computer interfaced instruments (R. E. Dessy - VPI & SU), data base management software and data bases (R. C. Graham - West Point), spread sheets (P. C. Flath - Paul Smith's College, and D. M. Whishnant - Wofford College), numerical methods (A. L. Smith - Drexel University), statistical methods (S. N. Deming - University of Houston), graphics and plotting (V. I. Bendall - Eastern Kentucky University) and electronic bulletin boards and networks (T. Russo - Bayonne High School and J. W. Moore - Eastern Michigan University). Each speaker will give some idea of the kind of software which is available. What the software can do and what it can not do. How it is being used and can be used by teachers and students in elementary and advanced courses. What future developments are likely to occur. Those attending who have used some applications software in innovative ways in teaching are invited to make brief comments during the 5 minutes available at the end of each topic. A form is included on the last page of this Newsletter requesting information concerning the extent to which you and your students use commercially available applications software. We would like to present some of this information either in the symposium or in the Newsletter.

Other sessions sponsored by the Division of Chemical Education include a symposium on Chemometrics, and general papers on computing (Wednesday afternoon and Friday morning). A symposium on Personal Computers in Analytical Chemistry is being sponsored by the Division of Analytical Chemistry.

CONDUCTOMETRIC TITRATIONS USING A GENRAD 1658 DIGIBRIDGE AND A DEC MINC-11/03 COMPUTER

by (Gil)bert F. Pollnow*

Some of the important features and applications of the DEC MINC-11 have been described in the June 1982 issue of the Newsletter by the author, and in a chapter of the book "Computer Education of Chemists" edited by Peter Lykos (Wiley-Interscience 1984). The MINC is still an exemplary laboratory computer, but by present standards is over priced and DEC discontinued its production last July. However, DEC's announced policy is to continue to support it with service for ten years. Some complete new systems are still available for approximately one third ($8000) of the original price from American Diversified Computer Products and Services, Inc., Haverhill, MA. In the seven years of use at UWO, it has proven to be extremely reliable and trouble-free.

The GenRad 1658 Digibridge ($3350 + $650 for GPIB) is a smart digital impedance meter and limit comparator which incorporates the latest microprocessor technology. It can be used to measure capacitance, inductance, or resistance at either 100 or 1000 Hz. Measurement results are clearly shown with decimal points and units, which are automatically presented to assure correctness. Display resolution is 5 full digits for C, L, and R (4 for D or Q). The overall basic accuracy is 0.1 percent. A frequency programmable bridge (Model 1689, 12 Hz to 100 kHz) with a basic accuracy of 0.02 percent, and the GPIB currently sells for $5850.

The general-purpose interface bus (GPIB) option provides full "talker/listener" and "talker only" capabilities consistent with the IEEE-488 Bus standard. Incidentally, the IEEE-488, ANSI MC1.1, and IEC 625-1 standards are all synonymous with the GPIB standard which was first introduced by Hewlett-Packard in 1965. A separate connector also interfaces with component handling and sorting equipment. System configurations with the GPIB bus may be either in the form of a star, a daisy chain, or a combination of the two. The unique connector allows connections to be made piggyback since each has both a male and female part. Data transmission rates are in the 500 kbyte to 1 Mbyte/sec range giving the GPIB a throughput of 4 to 8 Mbps. Some 4000 instruments from 500 different manufacturers are estimated to be currently utilizing the GPIB bus.

By virtue of a patented new measurement technique in which a microprocessor computes the desired impedance parameters from a series of 5, 8, or 16 voltage measurements, no user calibration adjustments of the 165B bridge are ever required. Even the sine wave test signal used to measure the impedances starts with a digital signal 256 times the selected test frequency, F, (100 or 1000 Hz). Binary dividers count down from 256 F, providing 128 F, 64 F, 32 F, ...2 F, F. This set of signals is used to address a ROM which contains a 256 step approximation to a sine function. The ROM output (as an eight-bit binary number) is converted by a D/A converter to a sine wave, which is then smoothed by filtering before going to the conductance cell.

Communication between the MINC and the Digibridge is by means of character strings, as illustrated in the brief program below, and as described in more detail in the second reference above.

```
100 REM PROGRAM  [RLC4] TO READ GEN RAD 1658 DIGIBRIDGE VIA IEEE-488 BUS
110 DISPLAY_CLEAR \ IEEE_BUS_CLEAR \ ALL_INSTR_CLEAR \ SET_TERMINATORS(10)
120 C=3 \ N=2
130 SEND("D2S0C0F1L1R4N2X4E1",C)
140 FOR I=0 TO N
150 TRIGGER_INSTR(C)
160 INSTR_TIME_LIMIT(0)
170 RECEIVE(C$,,C)
180 R=VAL(SEG$(C$,10,15))
190 E$=SEG$(C$,5,6)
200 IF E$=' O' THEN  \ F1=1 \ GO TO 230
210 IF E$='kO' THEN  \ F1=1000 \ GO TO 230
220 IF E$='MO' THEN  \ F1=1.00000E+06 \ GO TO 230
230 PRINT R,E$,F1*R
240 IF I=N THEN 260 \ PAUSE(10)
250 NEXT I
260 END


10.024      O        10.024
1.0006      kO       1000.6
.03259      MO       32590
```

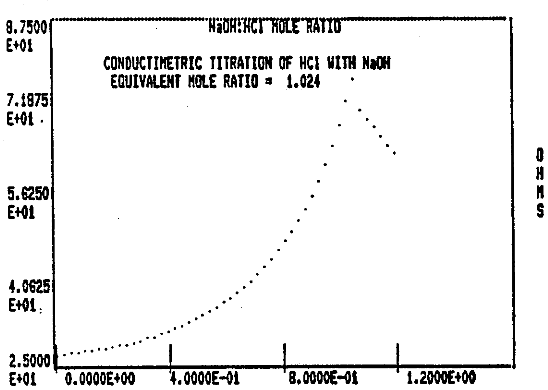
The MINC programmed instruction: SEND ("charaCter string", address) sets the instrument functions. This is followed in the program by the instruction: TRIGGER_INSTRUMENT (address) which actuates a measurement. The results of the measurement are returned to the computer by the programmed instruction: RECEIVE (character string,,address). The 17 character string which is received, is searched for the information of interest using the SEG$ (string label, first position, last position) operator. If numeric values are required, the VAL operator converts the string segment to its decimal equivalent. Other portions of the string are searched for labels which characterize the auto gain and measurement mode and must be utilized to multiply, by the correct multiple of ten, the previously generated number.

The conductometric titrations were carried out using a specially constructed dipping type pair of platinized Pt electrodes sealed, with epoxy resin, into a common support. Measurements were programmed to be taken 5 seconds after each addition of titrant, with continuous and vigorous stirring using a magnetically coupled Teflon coated bar. Control parameters of the Digibridge were programmed as follows: medium speed, parallel equivalent circuit, 1 kHz frequency, average of ten readings, and auto-ranging mode. A cell constant of 0.450/cm was calculated from manual measurements using a 0.0200 N KCl solution. The titration cell was supported in a controlled constant temperature bath at 25.0°C for all measurements.

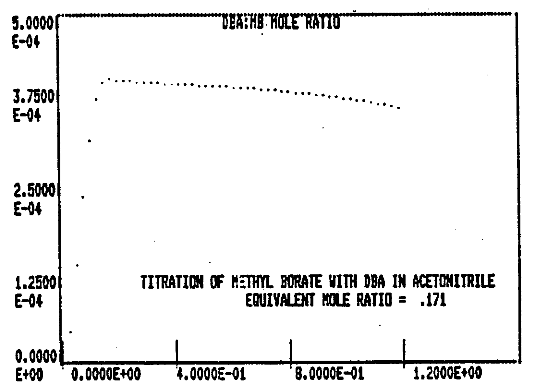
Prior to starting the titration of interest, the normality (or molarity) of both solutions (titrant and titrand), volume of titrand, calibration constant for the buret (ml/sec), and number of increments to be added for the complete curve are entered in the program.

Titrations were carried out utilizing a Sargent-Welch (Model C) constant rate buret (50 ml), with the volume of each increment of titrant being determined from an earlier calibration, under computer control via the D/A and clock modules. The mechanical volume counter on the buret was found to be consistently inaccurate relative to the computer calculated and measured volumes actually delivered, despite its apparent readout precision of 0.01 ml. Gordos solid state relays (GA5-4D25) which can be triggered by the 5 ma output of the D/A module formed the interface to the buret 115 V supply line. Temperature was monitored by means of a calibrated thermistor connected to the pre-amp module, which on the MINC is internally connected to a corresponding channel of the A/D converter.

Progress of the titration was continuously monitored on the CRT and the Okidata printer, both of which displayed the volume of titrant added, conductance in mhos, temperature in degrees C, mole ratio of the reactants, and the derivative of the conductance with respect to the mole ratio of reactants. In earlier versions, the titrations were monitored by simply plotting the resistance against the volume of titrant added. Since this program was developed for research purposes in non-aqueous solvents (acetonitrile), it was found more useful to plot the conductance in mhos vs the mole ratio of reactants. In either case, the change in sign of the slope can be used to precisely determine the equivalence point. The data are stored in sequential files for subsequent use prior to graphing. Copies of the CRT are obtained from a dedicated Tektronix 4632 video, silver-paper based, copier. Shown below are two titration curves, which are self-explanatory.



MOLARITIES: .100 NaOH, .100 HCl; Pt ELECTRODES, 1KZ, MED, PAR, AVE, 24.0 C



0.0996 M MB, 0.0991 DBA, Pt ELECTRODES, 1 KZ, MED, PAR, AVE, 25.0 C

In introducing these techniques, the student first utilizes the bare-bones program, shown above, and another one to gain familiarity with the Digibridge and the MINC by having the latter take a series of ten measurements on a manually operated precision decade resistance box at regularly spaced time intervals, and then plotting the resistance as a function of the time. With this experience and a discussion of the GPIB instructions involved, the extension to other applications is readily apparent.

The program can be readily adapted to determining the equivalent conductance as a function of concentration of strong and weak acids or bases, and true thermodynamic equilibrium constants extrapolated to zero concentration, as in the physical chemistry laboratory course. A separate publication of those results is planned.

A copy of the listing is available from the author upon request.

*Department of Chemistry
University of Wisconsin - Oshkosh
Oshkosh, WI 54901
phone: (414) 424-1480/1400

Software QUERIES and REPLIES

Send software queries, rebuttals and information to Ken Loach, Department of Chemistry, University of Delaware, Newark, DE 19711. (Ken is on sabbatical leave from SUNY at Plattsburgh for the 1985-86 academic year.)

Hardware QUERIES

Hardware queries and replies should be directed to: Jim Beatty, Chemistry Department, Ripon College, Ripon, WI 54971.

Please send me your hardware problems. Chances are there is a fellow reader who has already solved your problem. If you have solved your own problem and think that there are readers who would benefit from your solution, please send me the information for this column.

HQ-6 (Sept. '85)

A number of readers have inquired about using a digital multimeter (DMM) with Apple equipment for experiments similar to my article on page 16 of the March 1985 issue of this publication.

A HQ-6 (Sept. '85)

Apple Clocks The Apple IIe and IIc do not contain usable internal clocks as Commodore machines do. A review of popular clocks for the Apples other than the IIc is found in InCider, April 1985, page 46. Clocks which are available for the IIc are usually installed as part of an RS-232-C cable. See advertisements in publications such as A+ or InCider or your local Apple dealer.

A DMM with RS-232-C The DMM described in my interfacing article had an IEEE-488 interface. "Is there a DMM available with an RS-232-C interface?" I have an RS-232-C interface card in my IIe or my IIc or Commodore-64 has an RS-232-C interface built in." Omega Engineering markets a 4 3/4 digit DMM's with RS-232-C, IEEE-488 or Centronics interfaces in them for $735. If you are interested, write: Omega Engineering, One Omega Drive, Box 4047, Stamford, CT 06907 and request their bulletin on the M2110 DMM.

WHO DONE IT?

WHO DONE IT? information should be sent to the appropriate section editor (Hardware or Software - see QUERIES).

WHO-135 (Sept '85)

The Fall 1985 issue of the SIGUCCS Newsletter (Volume 15, No. 3) features a 14 page article by Derek Bok, President of Harvard University, entitled "Looking into Education's High-Tech Future" reprinted from Harvard Magazine. In the article Bok states "In theory, at least, the new technology, has the power to transform the nature of the university." He states that in the past "technology has failed to live up to its early promise for three reasons: resistance by teachers, high cost, and the absence of demonstrable gains in student achievement. There is as yet no clear evidence that computers and videodiscs will meet a happier fate." President Bok discusses some of the documented successes of computer use and some of the possibilities for eliminating drudgery, the use of computer-assisted instruction, the possibility of developing and using expert systems. (D.R.)

WHO-136 (Sept '85)

The May/June issue of ACCESS: The Journal of Microcomputer Applications in Engineering and Science (Volume 4, No. 3) is devoted to applied statistics. Many of the articles discuss programs and routines and provide BASIC listings. There is an article and programs on "Scaling Cartesian Graphics and Providing Convenient Unit Scale Values" (similar to the program which appeared in the June issue of this Newsletter). Other articles provide programs which calculate the probability of obtaining a given value of F, t, Z or Chi-Square; fits data to normal, log normal, gamma and Cauchy distributions; and calculates the area under the t-curve. The March/April issue of ACCESS was devoted to numerical analysis. (D.R.)

MICROCOMPUTERS IN PHYSICAL CHEMISTRY ANIMATIONS

by Larry M. Julien*

For the past several years, I have been introducing students to the use of microcomputers by assigning projects to create interactive animated simulations of physical chemical phenomena. The following describes the evolution of this process.

The first projects were carried out as senior studies. They consisted of programs that could number crunch and were used to evaluate and simulate theoretical and laboratory data. The students used Apples and were successful at modelling kinetic and equilibrium systems. There were six to ten students over a two year time span that worked on this type of project.

The next significant step in the evolution of this work involved the introduction of students to three CROMEMCO microcomputers. These microcomputers had very good color and animation capabilities. However, there were some major disadvantages: the systems were expensive; we were using an outdated language (FORTRAN IV); and the CROMEMCO's were not widely available for others to use the software that was produced from this work.

This stage was important for two reasons. First, twenty to twenty-five students were involved as a part of special projects in the third quarter of the regular physical chemistry laboratory course, where they worked in groups of two or three. Second, we became convinced that interactive animated simulation in color contributed to the success of our efforts.

The third stage of development was the widespread appearance of IBM PC's on our campus. Although at that time they did not have the capabilities of the CROMEMCO systems, they were more available for student use. This permitted the inclusion of a required computer project as an integral part of the regular physical chemistry lecture course. That was done in the third quarter of the course, consisting of ninety-five students, and counted as twenty percent of each student's grade. They used primarily Pascal and Basic as the programming languages.

The preceding describes the development of the introduction of our students to microcomputers via creation of interactive animated simulations for physical chemistry. Now I will describe some of the characteristics of and observations garnered from these projects.

The students involved in these projects were split about 20/80 chemist/chemical engineer and the class sizes ranged from 100 to 140. The engineers had had a one quarter course in programming mostly for main frame computers. The chemists normally had not had that background. I did not observe any significant differences in performance between students in the two groups. I have noted that for the past two years more students have a working knowledge of Basic. I expect this trend to continue. In any case, the lack of prior programming skills did not prevent the completion of the projects.

Each project contained the following steps:

1. Create a visual model.
2. Create a mathematical model.
3. Connect the models so that the visual model interacts with the mathematical model.
4. Provide parameters in the program which the user can change.

The significance of these steps is that the students are required to visualize the chemical phenomena and make the mathematical connections. I believe that this enables them to understand ideas and concepts more readily.

At the end of the quarter, each group of students did a "show and tell" presentation of their work before the entire class. This is important as it provides a strong incentive to complete the project and is the first part of evaluation of the results. I, as the instructor, completed the evaluation and assigned the grades for the projects.

I acted as an advisor on the chemical concepts and ideas. I did not give advice on details of programming: I expected each student to develop the necessary skills on their own, as they did. Most of my time was spent consulting with them on effective visual and mathematical models.

I hope that this sharing of my experience will give you some ideas on how to incorporate microcomputers into your classes. I have found it to be very rewarding and look forward to future hardware and software developments that will allow us to better describe and teach chemistry. Walt Disney is at our fingertips. Let us take full advantage of this opportunity.

*Department of Chemistry
and Chemical Engineering
Michigan Technological University
Houghton, MI 49931

"WISE" AUTHORING OF SOFTWARE
by M. Lynn James*

Historically, there have been several impediments to the widespread adoption of computer-based teaching in chemistry. The high cost of hardware, the fact that few chemists have the necessary expertise to develop and/or adequately utilize software in chemical education, and the availability of relatively little quality software have been among the principal impediments. With the advent of the microcomputer and the resulting drop in price, the cost of hardware has been considerably reduced. Efforts by individuals and groups, such as the workshops sponsored by the Committee on Computers in Chemical Education and Project SERAPHIM, have educated many chemists in the use of computers. Some very good software is currently available but this has required enormous effort by a few very dedicated programmers. Time estimates for producing such software can range from 50 to more than 300 hours of programming for each hour of useable software. Obviously, not every qualified programmer is willing to put in such time.

In addition to the expertise and time commitment necessary, a major hinderance to the future production of a large amount of quality software is the tedious nature of programming in BASIC, Pascal, assembly and machine languages typically used for developing much of the software. The task is particularly tedious when graphics are included in the program. One approach in alleviating this problem is the use of an authoring language. Authoring languages come in a variety of forms ranging from ones, such as PHOENIX, that are designed for use on main frame systems to those, such as Super Pilot, which work on micros.

Typically, those designed for use on main frames (1) do not give access to the full capability of the computer, (2) tend to optimize some important subset of machine capabilities, (3) are based on higher level languages with powerful commands available for generating text and graphics displays, accepting and judging responses, and carrying out a variety of branching, logging and score keeping functions, and (4) can become difficult to use as enhancements and additions are made to the basic syntax to increase their power. Furthermore, being main-frame based they are not useable in a classroom or laboratory setting.

On the other hand, authoring systems implemented on micros can be easily taken into the classroom or laboratory and are typically menu driven and therefore easily learned and used, but they are not too powerful in terms of the number and sophistication of the display and logical functions that they support.

A compromise that has the best features of the main frame and microcomputer authoring languages without many of their disadvantages is accomplished through hybrid authoring languages such as WISE (WICAT Interactive System for Education). This specific authoring language is menu driven, highly prompted, easy to learn, highly productive, uses a sophisticated instructional strategy which involves complex displays and logic that can be implemented much more quickly than in other authoring languages, and gives a range of options and ready access to a wide variety of programming utilities and tools that offer the user complete access to anything the computer can do.

Characteristics of WISE are its power which allows the author to perform any function of which the computer system is capable as part of any screen display (frame) in any lesson. It runs on a 32/16 bit processor supported by a true multi-user, multi-tasking operating system. WISE allows for three levels of authoring. The first level uses a highly prompted set of menus and requires no coding. This level allows for the presentation of information and free-response or multiple-choice types of questions, each allowing for the incorporation of a wide variety of graphical displays.

The second level of programming, which is based upon a Pascal-like language, allows for the types of calculations important in chemistry and includes the above features but in addition permits conditional branching, use of thousands of system variables (string, real, integer, long integer, and pool), calling the logic and display of other frames as subroutines, control of external devices such as videodisc players, and manipulation of common variable pools accessible to any set of simultaneous users.

The third level involves an External Program Frame Mode and enables the author to use anything the computer hardware and software is capable of doing as part of any frame of any lesson.

WISE includes a powerful Character Set Display Editor which allows for the following features:

1. The author is able to define an unlimited number of special symbols or character sets, consisting of any number of key assignments, down to the single pixel definition level. All such symbols can be varied in size and are moveable and rotatable by the Display Editor with no requirement for coding.

2. Sophisticated displays can be generated and manipulated including graphics, text, and animations.

3. Primitives include dots, lines, boxes, circles, center-point arcs, three-point arcs, text (in any font size, or orientation), and fitted-spline curves.

4. Modifiable attributes of any graphics objects include line width, line pattern, and different types of fills.

5. Manipulation of the order in which objects appear on the screen can be easily accomplished.

6. Any group of objects can be defined as a "segment" and treated as a single object without losing the capability of manipulation of its separate parts.

7. All objects can be scaled including scaling to negative values for mirror-image effects.

8. All objects can be copied into any other frame of any lesson, rotated, moved, and animated including multiple object animation with scaling, motion along defined paths (a very attractive feature for use in chemistry), and rotation, leaving a trail or not, saved in author defined graphics libraries and bit mapped.

9. Digitized audio, videodisc and videotapes displays may be included.

Function Key Editor features allow the author to define any set of keyboard and numeric keypad keys as special function keys with a great variety of purposes.

Student Response Modes may be typed in as alphanumeric responses, or as screen locations through predefined keys or touch screen, and time limit features for responses can be easily included.

Answering Processing includes the ability to search for key words ignoring all or certain other words, unlimited synonym dictionaries may be defined, and spelling tolerances may be set by the author.

Through these features, productivity of quality software is greatly increased and authoring requires much less skill in programming and becomes a much more rewarding experience.

*Department of Chemistry
University of Northern Colorado
Greeley, CO 80639

ELECTROLYTE DEMONSTRATION

by John Hnatow*

The purpose of this demonstration is to replace or to supplement the conventional light bulb conductivity apparatus used in general chemistry demonstrations. If one has access to microcomputers in a chemistry setting, this demonstration will process the collected data in a very efficient and organized way and allow students to conceptualize a very important theory. Most chemistry instructors not familiar with simple interfacing techniques will be quite surprised at both the simplicity of the programming and the low cost of the materials. I developed this program at the Allentown College/Moravian Academy Microcomputer Interfacing Curriculum Development Institute in 1984 and have had time this school year to refine it.

The BASIC program which must be loaded is included with these instructions.

```
  1 REM ELECTROLYTE DEMONSTRATION
  4 REM BY JOHN HNATOW, EMMAUS HIGH SCHOOL, EMMAUS, PA 18049
 10 HOME
 20 HGR
 30 HCOLOR = 7
 35 HOME : VTAB 24: INPUT "PLOTTING INTERVAL IN SECONDS IS "; A
 40 FOR J = 0 to 278
 50 X = J
 60 Y = PDL (1) * 159 / 255
 65 HCOLOR = 0: HPLOT X + 1,Y: HCOLOR = 7
 70 HPLOT X,Y
 80 FOR I = 1 TO A * 750: NEXT I
 90 NEXT J
 95 HOME : VTAB 24: PRINT "PRESS ANY KEY TO CONTINUE."
100 GET A$
110 GOTO 35
120 END
```

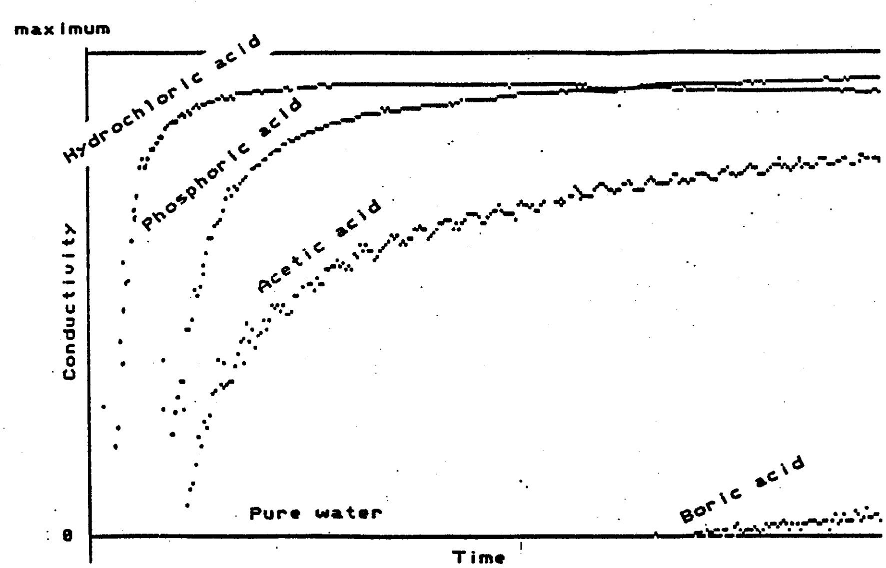
Two electrodes are connected to pins 1 and 10 of the game I/O connector of an Apple II+ or IIe. These electrodes may be platinum of nichrome wire (or even paper clips if your budget is limited). The electrodes are now immersed in distilled or deionized water and a conductivity vs time graph is plotted. Operational definitions for non-electrolytes, weak and strong electrolytes are established as 0.1 M solutions are added at a constant rate to 100 cc volume of pure water. Increases in conductivity or decrease in resistivity are observed as dissociation or ionization occurs in solution. The program allows placing each plot over the other to obtain relative differences between the conductivities.

Two alligator clips may be connected to a 10 pin DIP header (Radio Shack part no. 276-1980). The DIP header should be inserted into the game I/O connector.

The program will indicate a "plotting interval in seconds" prompt line, and the demonstrator selects the time interval. An interval of 0.15 is recommended in order to expedite the demonstration, and to avoid some potential problems caused by possible redox reactions at the electrodes. Initially, reference lines for no conductivity and highest conductivity can be established on the graph. First, run the program without any solution between them and then, run it while touching the electrodes together.

Start with pure water with the electrodes in position. Now, non-electrolytes, such as sucrose, can be added to the pure water by simply adding a spoonful or a small volume of the substance. A nice feature of the program is that a pointer will move across the screen if super position occurs.

In preparation for this demonstration, fill four labeled burets with 0.1 M solutions of acetic, boric, phosphoric, and hydrochloric acids and set them aside. The electrodes should be placed at fixed distances in a beaker containing pure water. The use of a magnetic stirrer is recommended. The acids must be added from a buret at a rate of approximately one drop per second. It is very important to rinse and dry the electrodes after each acid is run. Begin each new acid addition in a beaker rinsed and filled with pure water. A very weak electrolyte (such as boric acid) may not allow any change in conductivity on the first run. However, running the program a second time while the weak electrolyte is still being added at one drop per second should produce desirable results. A sample of a printout from a typical demonstration is shown below.



Since the Apple supplies 5 volts DC to the electrodes, in many instances gases may build up on the electrodes or a precipitate may coat the electrodes and drastically decrease the conductivity. To stop plotting before the plot has progressed to the end of the screen, simultaneously press CONTROL and C. There are now two options. If the plot is to be continued from the same place, type CONT and press RETURN. If a new plot is desired, type RUN 35.

After the demonstration is performed for the class, a sample printout is supplied for the students to label and discuss. Strong, weak, and non-electrolytes are more easily understood after operational definitions are developed with the aid of this microcomputer demonstration. More advanced high school classes might attempt to determine the acid dissociation constants of the weak acids from the results, and perhaps additional experiments.

Emmaus High School
Emmaus, PA 18049

FOOD FOR THOUGHT

We are, whether fully conscious of it or not, already in an environment for higher education that represents the most drastic change since the founding of the University of Paris and Bologna .... some eight or nine centuries ago.
Steven Muller, President of Johns Hopkins

Five decades of research suggest that there are no learning benefits to be gained from employing different media in instruction, regardless of their obviously attractive features or advertised superiority .... The best current evidence is that media are mere vehicles that deliver instruction but do not influence student achievement any more than the truck that delivers our groceries causes changes in our nutrition.
Richard Clark, Review of Educational Research,<u>53</u> ,445-450 (1983)

The issue is not whether the computer can be made to think like a human, but whether humans can and will take on the quality of digital computers.
J. David Bolter, "Turing's Man", 1984, p. 150.

WORKSHOPS, MEETINGS, CONFERENCES & COURSES

Please send information to Donald Rosenthal, Editor. Describe the program, include location sponsoring group, dates, <u>costs</u> and who to contact for further details (name, address, and phone number). Information should be sent as far in advance as possible.

January 20 - 21: SERAPHIM - C.C.C.E. High School Teacher Workshop at East Side High School, 1201 Southwest 45th Terrace, Gainesville, FL 32601
This workshop has two basic objectives- To illustrate how microcomputers can be of use to a chemistry teacher and to provide an opportunity to preview representative samples of commercially available software. Carol Bowers (Spring Valley High School, Sparkleberry Lane, Columbia, SC 29223 is the workshop leader. Contact Susan Zoltewicz (home - (904) 372-7946 or school - (904) 372-0447)

April 13 - 18: 191st ACS National Meeting in New York, NY. Symposia, general papers, poster sessions and exhibits. See Comments from the Editor section on page 2 for further details.

July 27 - 31: 9th Biennial Conference on Chemical Education at Montana State University, Bozeman, Montana 59717.
Programs for secondary school, 2-year college and 4-year colleges are planned. Contact Ken Emerson at the above address.

July 31 - August 2: Eighth C.C.C.E. National Computer Workshops at Montana State University, Bozeman, Montana.
Workshops: A. Design and Development of Computer-Assisted Instruction Programs (Alfred J. Lata, University of Kansas) B. Microcomputer Interfacing (James O. Currie, Jr., Pacific University) C. The Computer as a Tool (M. Lynn James, University of Northern Colorado). Contact M. Lynn James, Department of Chemistry, University of Northern Colorado, Greeley, CO 80634 for additional information. Telephone (303) 351-1285 or 2559. Additional information will appear in a subsequent issue of this Newsletter.

Handling Upper and Lower Case on Apple Computers

by Richard Cornelius*

The current Apple computers, the Apple II and IIc, properly display both upper and lower case letters, and the SHIFT key works to permit a user to select the case. On earlier Apple computers, the II and II Plus, lower case cannot be displayed on the text screen, and lower case letters cannot be typed from the keyboard, regardless of what is done with the SHIFT key.

Two problems frequently appear due to this incompatability between computers:

1) Programs written on the IIe or IIc may use lower case on the text screen. When these characters are printed on the screen of an Apple II Plus, they appear as an uninterpretable combination of inverse numbers and special characters.

2) Software written on an Apple II Plus may expect all letters typed at the keyboard to be upper case. Such software might demand "Y" or "N" as an answer to a question while rejecting lower case "y" and "n". Sometimes software will even tell users to place the caps lock key down instead of properly handling whatever is typed.

This article describes how both problems may be managed for programs on Apple computers, and gives the code required in both Applesoft BASIC and assembler. The assembler code can be used in conjunction with many BASIC programs that others have written, requiring no or very few changes to the primary BASIC program.

Listing 1 shows a BASIC program for which the display appears in mixed upper and lower case on an Apple IIe or IIc but in upper case only on an Apple II Plus. The CALL 768 within the program makes the display operate this way. The command PR#0 may be used to restore the unmodified display. In the same program, a CALL 802 turns the keyboard input (for GET or INPUT statements) into entirely upper case on any of the three models, independent of whether the SHIFT or caps lock key is used. The command IN#0 returns the SHIFT and CAPS LOCK keys to their normal function.

The BASIC program POKEs into memory a short machine language program that examines characters which are sent to the screen or typed and changes lower case into upper case when necessary. Listing 2 shows the annotated assembler code which accomplished the two tasks of handling the display and input. A CALL 768 hooks the routine at $30B to process output. The model of Apple computer in use is determined by an examination of the value found in location 64435 ($FBB3). Apple IIe and IIc computers (including the enhanced version of the IIe announced in March 1985) have the value 6 at this location. Apple II Plus computers have the value 234 ($EA) here. If the value 234 ($EA) is not found, then the routine does no further work. If 234 ($EA) is found, then the code from $314 to #31E changes all lower case letters into upper case letters. A CALL 804 hooks the routine at $32B to handle input. This code is active on any model of computer, but it will have no opportunity to find lower case input on an Apple II Plus. For generality, the routines have not been hooked up to DOS. The result is that they operate properly under both DOS 3.3 and ProDOS, but they may have the effect of turning off DOS. Rehook DOS 3.3 by using a CALL 1002 and rehook ProDOS with a CALL 39565. Usually RESET will also rehook DOS.

Apple II Plus computers can be modified to better handle upper and lower case letters. A lower case display chip, available for about $30.00, permits a mixed upper/lower case display. One way of enabling the SHIFT key is to connect one wire of the keyboard to the connection used for game paddles. Usually the flag input at 49251 ($C063) is used so that the operation of paddles need not be affected. Then a short machine language input routine similar to the one used here can examine this location to determine whether the SHIFT key is down. Some word processors (such as Screenwriter) and quiz maker programs (such as All of the Above) permit the user to specify whether such changes have been made. Either modification may be made and used independently, but the combination of the two overcomes one of the most important shortcomings of the Apple II Plus computer compared to the Apple IIe.

The technique of having a special machine language routine to filter the input of output is not limited to controlling the case of letters. An example of the application of this technique may be found in the software package Concentrated Chemical Concepts published by John Wiley & Sons. This software package was written and published before Apple IIe computers were available. Several programs in this package use the shift key to signal that a number should be printed as a superscript. For example, when the SHIFT key is held down while the "2" key is pressed, the programs print a superscript 2. On the Apple II Plus computer, pressing the SHIFT key with the number 2 produces quotation marks. Thus the programs print a superscript 2 whenever the quotation marks are typed. On the Apple IIe, pressing the SHIFT key with the number 2 produces the at sign, @. A revised edition of the software handles the change in keyboard layout with no changes to the BASIC code. Instead a short machine language routine for handling input turns @ into quotation marks <u>if</u> the computer in use is an Apple IIe or IIc.

Listing 1. Applesoft Program Showing Flexibility in Treating Upper and Lower Case.

```
100  REM TEST PROGRAM

110  FOR SPOT = 768 TO 825: READ CODE: POKE SPOT,CODE: NEXT
120  DATA 169,9,133,54,169,3,133,55,96,134,6,174,179,251,224,
          234,208,11,201,225,48,7,201,251,16,3,56,233,32,166,
          6,76,240,253,169,43,133,56,169,3,133,57,96,32,27,253,
          201,225,48,7,201,251,16,3,56,233,32,96
130  HOME
140  CALL 768: REM USE PR#0 TO CANCEL
150  PRINT "After a CALL 768 is executed, this"
160  PRINT : PRINT "program will display only upper case"
170  PRINT : PRINT "on an Apple ][ Plus."
180  PRINT : PRINT : PRINT : PRINT
190  CALL 802: REM USE IN#0 TO CANCEL
200  PRINT "Once a CALL 802 is executed, input will"
210  PRINT : PRINT "automatically be converted to upper"
220  PRINT : PRINT "case on any kind of Apple computer."
230  PRINT : INPUT I$
240  PRINT : PRINT I$
```

Listing 2. Assembler Program for Changing Input and Output to Upper Case.

```
SOURCE FILE: I/O
0000:            **********************************
0000:            *                                *
0000:            * Routines for handling upper &  *
0000:            * lower case on Apple ][ Plus,   *
0000:            * //e, and //c computers         *
0000:            *                                *
0000:            **********************************
0000:            *
0036:            CWSL     EQU  $36       ;output hook
0038:            KWSL     EQU  $38       ;input hook
FDF0:            COUT     EQU  $FDF0     ;sends character to screen
FBB3:            IDBYTE   EQU  $FBB3     ;test for model of Apple
FD1B:            KEYIN    EQU  $FD1B     ;routine to get keystroke
0006:            SAVEX    EQU  $06       ;temporary storage
----- NEXT OBJECT FILE NAME IS I/O.OBJ0
0300:                     ORG  $300
0300:            *
0300:            **** This routine handles screen output.
0300:            *
0300:A9 09       HOOKOUT  LDA  #>OUT     ;Hook routine
0302:85 36                STA  CWSL      ;called 'OUT'
0304:A9 03                LDA  #<OUT     ;to handle output.
0306:85 37                STA  CWSL+1
0308:60                   RTS
0309:86 06       OUT      STX  SAVEX     ;Save X-register.
030B:AE B3 FB             LDX  IDBYTE    ;Is this computer a
030E:E0 EA                CPX  #$EA      ;][ Plus?
0310:D0 0B                BNE  SEND      ;No.
0312:C9 E1                CMP  #$E1      ;Check whether
0314:30 07                BMI  SEND      ;character is
0316:C9 FB                CMP  #$FB      ;lower case.
0318:10 03                BPL  SEND
031A:38                   SEC            ;Make lower case into
031B:E9 20                SBC  #$20      ;upper case.
031D:A6 06       SEND     LDX  SAVEX     ;Restore X-register.
031F:4C F0 FD             JMP  COUT      ;Send character to screen.
0322:            *
0322:            **** This routine handles keyboard input.
0322:            *
0322:A9 2B       HOOKIN   LDA  #>IN      ;Hook routine
0324:85 38                STA  KWSL      ;called 'IN'
0326:A9 03                LDA  #<IN      ;to handle input.
0328:85 39                STA  KWSL+1
032A:60                   RTS
032B:20 1B FD    IN       JSR  KEYIN     ;Get a keystroke.
032E:C9 E1                CMP  #$E1      ;Check whether
0330:30 07                BMI  RETURN    ;character is
0332:C9 FB                CMP  #$FB      ;lower case.
0334:10 03                BPL  RETURN
0336:38                   SEC            ;Make lower case into
0337:E9 20                SBC  #$20      ;upper case.
0339:60          RETURN   RTS

*** SUCCESSFUL ASSEMBLY: NO ERRORS
```

*Lebanon Valley College
Annville, PA 17003

Dennis Sievers, from Central Community High School in Breese, Illinois, joins the ranks of our reviewers this month. Dennis has broad experience with microcomputers, having used the Apple, Commodore, IBM PC, IBM PCjr, as well as some single board systems, and is familiar with BASIC, Pascal, FORTRAN, and assembly language. He is also a feature editor for the Secondary School Chemistry Section of the Journal of Chemical Education. His major area of computer interest is interfacing, and judging from his first review, his expertise in this area should make him an excellent contributor to this column.

If you wish to comment on this column or suggest books for review, please write to Dr. Harry E. Pence; Book Review Editor, C.C.C.E.Newsletter, Department of Chemistry, SUNY-Oneonta, Oneonta, NY 13820. Of course, it would be a pleasure to receive the names of other individuals who wish to volunteer to serve as reviewers.

THE APPLE CONNECTION
by James W. Coffron
SYBEX, Berkeley, CA
1983, paperback, $12.95
Reviewed by Dennis Sievers*

Sybex has been publishing high quality books for computer users for some time, and The Apple Connection continues this pattern. The main thrust of the book is to provide users with an introduction to computer interfacing. The book is elegant in its simplicity yet quite thorough in its treatment of computer interfacing projects.

The first several chapters are devoted to an introduction to computer architecture and the use of the expansion slots for interfacing. To this end, the book uses the Creative Microprocessor Systems I/O card (CMS I/O card). While the name and address of the manufacturer of the card are given, no cost is provided. Without the card to work with the first three chapters are difficult to comprehend. Considering that the board is used as a trainer, it might well have been wise to show the reader how to build such a board from components. The balance of the first three chapters deals with the programming needed to access the peripheral slots. Clear, concise and detailed presentations are given to such topics as the means of addressing each slot, interpretation and weighting of the data lines and the BASIC commands needed to perform these tasks. Additionally, several BASIC programs are provided along with flow charts to demonstrate the desired concepts. These programs include turning on a Light Emitting Diode (LED), turning on the LED's in a wave type pattern, slot detection and a decimal to binary conversion program.

Several interfacing problems are examined in the subsequent portions of the book. The design of the hardware and software for a home security system is most impressive. The author provides an exact schematic diagram for the interface board and in a stepwise manner, leads the reader through the steps in software development. The circuit is used to detect the unauthorized opening of any of nine windows or four doors and sounds an alarm if this occurs. Having followed the directions provided, I now have a very useful security system for less than $30.00. Control of an AC load is also treated. A complete discussion of line isolation and the use of solid state relays (opto-isolators and opto-couplers) is presented prior to the actual design of a circuit to control a lamp.

The last three chapters offer a tutorial in the construction and use of analog/digital, digital/analog converters and transducers. The AD 570 and AD 558 integrated circuits are used for these purposes. The interface that results from this project is a complete eight bit A/D-D/A conversion board having unlimited potential. The author carefully takes the reader through the construction, programming and use of the board as an electronic thermometer and also a device controller. It cost me under $45.00. to build the board. This is considerably less than comparable commercial units.

The treatment of each topic is well documented and illustrated with many examples. The use of flow charts to examine the construction of BASIC programs enhances the reader's understanding of the program's function and operation. Some limited knowledge of electronics is assumed. Fundamental interpretation of schematic figures and assembly techniques are not illustrated. I did not find this a particular problem as the information is available from other sources. The schematic diagrams and the presence of data specification sheets enable the reader to modify the projects to fit specific needs. While it is possible to gain some insight and understanding without the CMS I/O card, it is an advantage to use it. The reader must wait for the card to arrive once ordered, thereby postponing use of the book. Programs to display data to a graphic screen were missing. Considering the popularity of graphics, this omission is unusual.

These minor objections do not detract from an excellent resource for newcomers to interfacing. Each topic considered also includes ideas for expansion and modification. I have found several interesting applications for the security program and dozens of uses in the laboratory for the A/D-D/A board. This book certainly belongs on the shelf of anyone interested in computer interfacing and computer control.

NOTE: This review concerns The Apple Connection. There are similar titles organized and arranged along the same lines as the volume reviewed here which provide data for the VIC-20 and IBM PC.

*Central Community High School
Route 50 West
Breese, IL 62230

DATA FILE PROGRAMMING ON YOUR IBM PC

by Alan Simpson
SYBEX, Berkeley, CA
1984, paperbound, $15.95
Reviewed by Harry E. Pence*

Many important computer applications require that data be stored in or retrieved from data files. Learning these techniques is frequently the next step the student takes after mastering the fundamentals of a programming language. Simpson's book shows how to use data files with the BASICA interpreter on the IBM PC. An understanding of this process should also be helpful when working with other programming languages.

The book does an excellent job of explaining both sequential files (where the individual data records are always accessed in sequence), as well as random access files (where the records may be accessed in any order). In both cases, the most difficult concepts for the neophyte to grasp are the relationships involving the RAM memory, the disk, and the buffer that acts as a go-between to pass information between these two. The author uses a large number of excellent diagrams to clarify these distinctions. Indeed, this is one of the strongest features of the book.

Many practical examples are chosen to demonstrate the characteristics of the two types of files. For instance, a grade book program is used to develop the essential features of sequential files. Thus, the reader not only learns the principles but also creates a potentially useful utility for his computer. A program to keep track of sales records illustrates the use of random access files. This program will probably be less useful to readers of this journal, but the use of random access files to create a library reference system is described in the appendix. This should be extremely valuable for many chemists. Of course, commercial software is available to do both of these jobs, but many readers will find considerable satisfaction in knowing that they have created their own system and can design it to suit their own needs.

The author assumes "a simple knowledge of the fundamentals of BASIC (pg. xiv)," but it would probably be helpful to have a more extensive background. Although a brief summary of BASIC is provided as an appendix, those who have not done much work with this language should be prepared to look up some material in a standard text. This is especially true in the library reference system, where there are several errors in the program listing.

Simpson is to be complimented for using a step-wise approach to develop the sample systems. He first divides the problem into a sequence of smaller projects, then creates program modules to fulfill each objective. This method makes it less difficult to comprehend the material and also much easier to debug the result. A diskette can be purchased that includes all of the programs. Unless you are a good typist, using this would probably save some frustration, but it might not be as effective for learning how to use data files. There are few better tests of your understanding than trying to locate your typo that is crashing the program. For those who choose the easier way, the author initially presents each of the major techniques by means of a short program which can be used to gain an understanding of the concept.

In general, the sample programs are sophisticated enough to be quite useful. The library utility includes not only routines for up-dating and deleting, but also sorting on one or two fields, printing, and searching records for keywords. Another appendix lists a mailing list management program, which provides sorting, editing, and searching capabilities. The programs do not include much provision for error trapping, and though this might be reasonable if the only person who will use the program is the one who has written it, most users will probably wish to add more extensive error procedures.

Simpson has written a clear, readable book that does a sound job of introducing the basic principles of random access and sequential file handling procedures. In addition, when the book is completed the reader will have gained several helpful utility programs. The presentation does have a few minor shortcomings, but it would be a reasonable purchase for those who wish to extend their knowledge of BASIC to more advanced programming procedures.

*Department of Chemistry
SUNY-Oneonta
Oneonta,NY 13820

COMPUTERS IN THE LABORATORY

by Joseph G. Liscouski (ed.)
American Chemical Society, Washington, D.C.
1984, hardcover, $34.95

Reviewed by Harry E. Pence*

The interfacing of computers and microprocessors with modern analytical instruments has produced the capability to generate large amounts of data very quickly. Unfortunately some chemists have found this new power to be a mixed blessing, for it is often difficult to identify the information content in the resulting array of observations. Equally important, many government agencies, such as FDA and EPA, are requiring more stringent record-keeping standards. The obvious solution to this "data explosion," as it is sometimes called, has been to further expand the use of computers for analyzing, reporting, correlating, and storing laboratory data.

At the fall national ACS meeting in 1984, a symposium was presented on applications of computers in the chemical laboratory. This book is based on ten of the papers from that conference. The various authors discuss a broad range of uses for computers, not just data collection and analysis, and the result indicates the current areas of interest as well as the new developments that can be expected.

These papers suggest that computers will play an increasing role in the laboratory of the future. Analytical instruments will be connected to computer networks and computers will analyze the data that these instruments generate by comparison with standard databases and interpret it with Artificial Intelligence (AI) systems. Robots will be used to perform tasks that are tediously repetitive or especially dangerous. Graphics, simulations, and mathematical analysis will identify data correlations and also model the results of alternative decisions; a procedure already familiar to those who use spreadsheet software for business. Even though no single institution currently integrates all of the applications described, many companies are already using one or more of these techniques. This indicates that such an ideal laboratory may be well on the way to becoming a reality.

As might be expected in a symposium, there is considerable variation in focus and approach from article to article. For example, several of the articles deal with various aspects of instrument networking and Laboratory Information Management (or LIMS) Systems, but the result fails to give a balanced view of this vital area of development. These articles are concerned with specific commerically available systems, and general principles are presented only as they deal with the system under consideration. Thus some significant topics are well covered but others of equal importance are neglected.

The longest article is a discussion of graphics. This article is perhaps one of the most useful for many readers of this journal, The treatment assumes little specialized background and should be excellent for someone who wishes an overview of current graphics techniques. Some presentations seem so short that they only whet the appetite for information, such as a paper on robotics, although one of the shortest articles was perhaps the most prophetic, a description of computer use by chemists in the Corps of Engineers. For the Engineers, their "laboratory" is the entire Ohio River, and computers are used to operate an automated water quality laboratory, assess water quality, and simulate the effects of possible water management decisions. In both cases an expanded treatment would have been both justified and welcomed.

The last three papers in the book deal with organizing and extracting information by means of computers. Clustering methods are demonstrated by showing how they are being used on data from the Merck Index to find structure-effect relationships among various compounds that are used for medicinal purposes. Another paper describes the use of information enhancement techniques to establish optimum conditions for obtaining structural or activity information from voltammetric electroanalytical analyses. The final paper argues that linearity and determinism are often assumed with insufficient justification in data analysis, and since computers allow scientists to develop more realistic models, this freedom should be used more often.

The foreword of the book notes that the individual articles were not typeset but were provided by the authors in camera ready form. In some cases this produces more than the usual number of typographic errors and one article did copy rather poorly, but these problems are only a minor irritation. The index is adequate, but the individual chapter bibliographies are often rather brief and in several cases nonexistent. Few of the authors are associated with academic institutions, an indication of the high level of interest in computer applications found in industrial and governmental laboratories.

Overall, the volume suffers from the typical drawbacks that might be expected from the symposium format; the papers tend to be rather narrowly focused and different authors assume that the reader has varying levels of prior expertise. Most readers will gain some useful information from the book, but it fails to provide the well-integrated survey that is needed. Aside from journal articles, such as those in <u>Analytical Chemistry</u> and the <u>Journal of Chemical Education</u>, there are too few attempts to present an integrated perspective on the current status of computer applications in chemistry. It is regrettable the book cannot be enthusiastically recommended. On the other hand, the articles describe areas where current interest is high, and the book might be considered for purchase on that basis.

*Department of Chemistry
SUNY-Oneonta
Oneonta,NY 13820

PROJECT SERAPHIM

National Science Foundation: Science Education

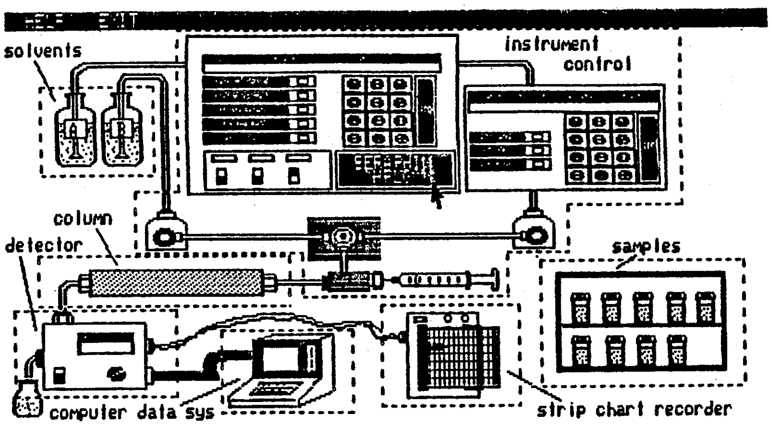
CARRYING OUT POWWOW II: 1985 SUMMER FELLOWS

In mid-May we held our second Powwow, a working meeting of a diverse group of people involved in chemical education, but not necessarily in computer use. The participants were brought together to bring unique viewpoints and talents to one issue: how best to use microcomputers in chemical education.

Five subgroups met in intensive sessions for most of four days to develop plans to produce computer programs in five areas:

- Extraordinary Instrument Simulations
- Advanced Structure and Bonding
- Predicting Inorganic Reactivity
- Inexpensive Interfacing
- KC? Discoverer: Means of Exploring Science with a Computer

The plans and outlines are now in place, albeit being revised and refined, and our Summer Fellows are beginning to turn them into reality. The person carrying out the charge to develop an "extraordinary" instrument simulation is Bob Rittenhouse of Walla Walla College. He is busily programming an IBM PC to transform it into an HPLC. The figure below shows what one of the first screens looks like; a mouse will be used to operate the controls on the HPLC.



William (Flick) Coleman of Wellesley College is working with two student programmers to produce an IBM PC program that will teach advanced spectroscopy and bonding theories. Paul Groves from South Pasadena High School (CA) and Derek Davenport from Purdue University are working together as programmer and idea man on programs to teach oxidation/reduction reactions, acid/base reactions, and prediction of inorganic reactivity.

Work on the inexpensive interfacing experiments was started by SERAPHIM Fellow Malcolm Rasmussen, who visited from Australia during Winter Semester. This work is being continued by Pat Barker from Minisink Valley High School (NY) and Ken Hartman from Ames High School (IA), with the assistance of any other fellows or visitors who will act as guinea pigs. (Nancy Whiteside, a graduate student at Purdue, visited for one week and found, and fixed, several bugs.) Pat and Ken are also developing an Interfacing Workshop based on inexpensive materials and our new Laboratory Modules (see description on p.).

David Whisnant from Wofford College (SC) has completed a section of the KC? Discoverer program which should be especially useful in high school or lower-level college courses. This prototype "Help Section" gives appropriate advice when requested by the student and allows instructors to specify and save sets of help messages for different problems.

Tamar (Uni) Susskind from Oakland Community College (MI) has the extraordinarily important job of checking--every program, every screen: does it work? is it accurate? will it crash? can you get out of it? is it classified correctly? She is checking the 1984 Program Contest entries, Apple --> IBM translations, Atari programs, other new programs that are contributed. So student programmers may groan, but programs will not leave here until they are "Uni-proof".

Peter Moskaluk is from Grosse Pointe High School (MI). Last year he set up the computer conference CHYMNET that links together SERAPHIM Workshop Leaders, Advisory Group Members, and staff. This year he is refining CHYMNET, doing some trial expansion of it, and trying to develop a model system for using business-oriented programs such as spreadsheets, databases, graphing programs, and integrated software effectively in the high school chemistry classroom. He will, if successful, plan a Utilities Workshop around these materials.

Except for the inexpensive interfacing materials none of the above programs are yet at the stage where they will be listed in our catalogue. However, we are looking for beta testers--persons who are willing to take a program that is under development, try it out in their classrooms or labs, and tell us what needs to be changed, fixed, or added. If you are interested in performing such a service, please contact Project Headquarters at the address below.

SERAPHIM FELLOWSHIPS AVAILABLE

For those of you who plan ahead, we bring to your attention two kinds of SERAPHIM Fellowships. There are Academic Year Fellowships for high school or college/university teachers to spend one or two semesters in residence at Project Headquarters working on special projects such as the inexpensive interfacing experiments worked out by Malcolm Rasmussen last winter. The next fellowships available are for the 1986-1987 academic year. [During the 1985-1986 academic year Victor Bendall (Eastern Kentucky University), Ralph Gable (Davidson College), and Lucy Pryde (Southwestern College) will be Fellows.]

In addition, there are Summer Fellowships for 1986. These, too, are open to both high school and college/university teachers. We are especially interested in finding high school teachers who would like to work next summer on problems of the type described above.

For more information as well as Application Forms, write to: John W. Moore, Director, Project SERAPHIM, Department of Chemistry, Eastern Michigan University, Ypsilanti, MI 48197.

PROJECT SERAPHIM
National Science Foundation: Science Education

NEW RELEASES

At the present time we are working frantically, with the help of Summer Fellows and an expanded staff of programmers, to have many new materials ready to be released when our new Catalogue comes out in late August. We thought you might like advance notice of what is in the works:

Programs from the 1984 Program Contest: these are mainly, but not exclusively, Apple programs. (They were not ready earlier because we were putting most of our time into setting up the new system of Apple disks, which were reorganized according to ten categories with related programs occupying the same diskette.) Our new Apple software will be organized according to the new system, will have menus, will be debugged, etc. There are at least five full disks of Apple programs that we will release. There will be IBM PC, Atari, and Commodore disks from the Program Contest that will be released, too, and perhaps some translations to the latter machines from our collection of Apple software.

Laboratory Modules: these are a new type of module and represent material developed at Project Headquarters by SERAPHIM Fellows. The written materials have instructions for the construction and use of (inexpensive) computer interfaces in the laboratory. These interfaces employ the game ports of Apple, Commodore, and IBM microcomputers. (See Ken Ratzlaff's descriptions of game-port interfacing in previous issues of this Newsletter for an idea of how such interfaces work.) In addition to showing how to build your own interfaces our materials include write-ups and student directions for several experiments that can be done in high school or freshman college labs using the interfaces. We are now in the final stages of testing and debugging both programs and experiments before release.

Software List: is scheduled to be released in August after its semi-annual revision (and inevitable expansion).

Catalogue: will be more than ready for its semi-annual revision if it is to reflect all of the above additions to our offerings. Needless to say it will be a good bit larger.

Persons on the Project SERAPHIM mailing list will receive news of the revised Software List and Catalogue via SERAPHIM News as soon as these become available. If you are not currently on the mailing list and would like to be, please use the form below to send us your name and address:

Name: ______________________

Address: ______________________

____

____

Mail to:
Project SERAPHIM
Department of Chemistry
Eastern Michigan University
Ypsilanti, MI 48197

SURVEY ON USE OF COMMERCIAL APPLICATIONS SOFTWARE IN EDUCATION

Degree of Use	None	A Little	Moderate	A Great Deal
A. Word Processing				
By you the teacher	( )	( )	( )	( )
By the students	( )	( )	( )	( )
Identify software and hardware				
B. For use with computer interfaced laboratory instruments				
By you the teacher	( )	( )	( )	( )
By the students	( )	( )	( )	( )
Identify software and hardware				
C. Data Base Management Software and Data Bases				
By you the teacher	( )	( )	( )	( )
By the students	( )	( )	( )	( )
Identify software and hardware				
D. Electronic Spread Sheets				
By you the teacher	( )	( )	( )	( )
By the students	( )	( )	( )	( )
Identify software and hardware				
E. Numerical Methods				
By you the teacher	( )	( )	( )	( )
By the students	( )	( )	( )	( )
Identify software and hardware				
F. Graphics and Plotting				
By you the teacher	( )	( )	( )	( )
By the students	( )	( )	( )	( )
Identify software and hardware				
G. Networks and Electronic Bulletin Boards				
By you the teacher	( )	( )	( )	( )
By the students	( )	( )	( )	( )
Identify software and hardware				

Describe uses on an additional sheet of paper

Return this form and descriptions to:

Dr. Donald Rosenthal
Department of Chemistry
Clarkson University
Potsdam, NY 13676