Macintosh Laboratory Automation: Three Software Packages

BARBARA ANN JEZL

To many people, "laboratory automation" brings to mind high-profile, high-ticket items such as bar codes, robots, and laboratory information management systems (LIMS). Fundamentally, however, laboratory automation is the use of machines and instruments to enhance and expedite the scientific process. Computers and "intelligent" instruments containing computers are often used to "automate" the scientist's tasks. Common laboratory tasks include the performance of numerous, often repetitive experiments and the collection of data from those experiments. The use of computers to control experiments and collect the data in machine-readable form allows greater reproducibility and precision, reduces human error and exposure to hazardous environments, permits higher rate and resolution of data collection, and allows unattended operation in some instances. Having the experimental data in the computer also facilitates the automation of the data analysis (*1*), retention, recall, and reporting steps of the scientific process.

To perform the acquisition and control operations, the computer must be able to transmit instructions or control signals and receive data and status signals from the experiment. Modern digital computers handle information as digital bits, bytes, or words (*2*). "Dumb" instruments (those lacking internal microcomputers), machines, tranducers, and sensors involved in experiments operate with analog voltages or current signals (*3*). These signals can be converted to digital values (*4*) meaningful to the computer, and the computer's commands can be converted to analog form for the instrument, machine, or experiment, by devices called analog-to-digital converters (A/D or ADC) and digital-to-analog converters (D/A or DAC). These converters can be part of the external computer system, or if the equipment is "intelligent," the converter may be inside the instrument (*5*) or machine. In that case, the already digitized information may be communicated to an electrical gateway, commonly called a bus (*6*), through the external ports of the computer and equipment.

Since its introduction in 1981, the IBM PC has become the predominant personal computer (7) in the laboratory. A large established base of hardware and software for data collection, control, and data analysis currently exists. Much of the hardware is tied to the internal IBM PC ISA bus standard (AT bus), which was to be discontinued by IBM in the evolving PS/2 line. Third-party software and hardware vendors and makers of compatible computers, feeling that their value-adding products deserved some credit for the success of the IBM PC, decided to diverge from the prescribed path and create their own bus standard for use with the existing hardware sold by the vendors and owned by millions of users. By the end of 1989, the "Gang of Nine" (*8*), as this group is called, were introducing products for this renegade extended ISA or EISA bus, bypassing the new IBM MicroChannel (MCA) bus.

At the same time the Apple Computer's Macintosh was establishing itself in the world of scientific and technical computing. Traditionally, the Macintosh's strength has been in the areas of quality word processing, presentation graphics, and desktop publishing (*9*). This has been of value to the laboratory scientist, as well as the business person and journalist. The mouse-oriented, bit-mapped drawing capability, clipboard transfer facility, easily integrated and utilized laser printers, and variety of scalable scientific fonts make it especially valuable for automating the necessary tasks of writing scientific papers and reports, and preparing presentations. The friendly Graphical User Interface (GUI, pronounced "gooey") and consistent human interface design standards (*10*) make it easy to use and popular with customers with all levels of experience.

Physical and chemical modeling programs are available to take advantage of the excellent graphics capabilities of the machine. The first volume of an Apple Science CD ROM disk was released at the end of 1989. The variety of data analysis and plotting programs available range from spreadsheets to symbolic manipulation programs. Unix coprocessor boards for Macintosh NuBus slots allow access to high-end workstation software, and the speed of reduced instruction set computing (RISC), with the convenience of the Macintosh GUI. However, most scientists and engineers familiar with IBM-based PCs have only recently been aware of the Macintosh in the data-acquisition and control area.

Before the Macintosh II was introduced, the Macintosh's capabilities for data acquisition and instrument control were severely constrained. Data and control signals to and from external devices could only take limited routes. Data could be entered through a keyboard, copied from a disk, or transmitted through one of the standard ports of the machine (SCSI, printer, or modem). The open NuBus slots inside the Macintosh II allowed the addition of real-time cards that could measure or emit voltage and current signals or digital pulses, permitting direct interfacing with machines and instruments.

Unfortunately, the non-NuBus Macintosh SE, SE/30, and portable all have their own individual, internal open buses that are different from each other. All will take cards, but not the same ones. The production of cards for the non-NuBus systems has been limited, possibly discouraged by the expectation of lower sales volume and shorter product life of the smaller computers. Before the open bus, it was necessary to have interface boxes, or intelligent cards in the scientific instruments or automation machines, to do the conversion. The data then were sent to the computer or commands came from the computer through one of the external ports. This is still the only route that can be used universally for all Macintosh models.

Two communications standards are most commonly used for these remote systems: serial (such as RS-232C or RS-422) and IEEE-488 (GPIB or HPIB). The IEEE-488 is a fast, reliable communications bus, popular in the electronics testing area. Fourteen other devices can go on one bus and are addressed by unique identifying numbers. It is a more expensive protocol to implement and support and requires that the communicating instruments and machines be nearby. The Macintosh can communicate with these devices through GPIB cards or GPIB interfaces connected through the serial or SCSI ports.

Serial communications are more common in the scientific research laboratory. External modems, terminals, local Ethernet connections, and many printers use RS-232C (*11*). It is the communications option most commonly available for intelligent scientific instruments such as electronic balances, chromatographs, scintillation counters, titrimeters and assay automation machines. The

Scientific Computing Division, Central Research and Development, E. I. du Pont de Nemours and Co., Inc., Wilmington, DE 19880–0320.

Macintosh RS-422 modem and printer ports can communicate with these through a serial printer cable.

Data-Acquisition Hardware: MacADIOS II Board

The three software packages chosen for this review, LABTECH NOTEBOOK, LabVIEW, and Parameter Manager PLUS/TALK, all can use the GW Instrument's (GWI) MacADIOS II board (*12*) installed in the test Macintosh II. The packages were chosen because all of them are produced by software houses that are independent from the hardware vendor. The ability to mix and match software and hardware guarantees the optimal selection and price situation for the customer and tends to keep the vendors trying harder.

The basic MacADIOS II board has 16 single-ended (8 differential) multiplexed analog inputs with selectable ranges of 0 to 10 volts or −10 to +10 volts, and amplifier gain settings of 1, 10, and 100. A sample-and-hold circuit maintains the signal for the successive approximation converter. The capacity can be expanded to 112 single-ended (56 differential) channels with the addition of three multiplexer daughterboards (integrated circuit modules that plug into slots on the main board). Faster (833 versus 142 kHz) or higher resolution A/D converters (16 versus 12 bit) are available on daughterboards. The conversions can be triggered by software command, transistor-transistor logic (TTL) signal, or on-board counter-timer pulse.

There are two independent 12-bit digital-to-analog conversion channels on the main board. Remote sensing allows feedback to provide a full voltage signal at distances greater than 30 m. The settling time for a full-scale step change is 9 μs. A daughterboard is available with two 16-bit DACS.

In addition, there are eight nonlatching TTL-compatible digital inputs (DIN) and eight latching digital outputs (DOUT) on the main board. With three optional daughterboards, this can be expanded up to 54 DIN and 56 DOUT. Signal break out boxes are available for easy access to the A/D, D/A, DIN, and DOUT signals.

This board can be placed in any NuBus slot inside the Macintosh II and its NuBus relatives. It should be placed as close to the fan as possible because it draws a significant amount of power with the daughterboards. The slot to the right of it cannot be used if the daughterboards are installed because of physical constraints.

The vendor supplies software libraries for control of the board from Microsoft QuickBASIC, THINK's Lightspeed C, MPW C, MPW Pascal, Language Systems' MPW FORTRAN, absoft MacFortran/020, and Turbo Pascal programs. GWI provides the Softpanels program free of charge with many of its products and also sells a general-purpose acquisition program, MacADIOS Manager II, SoftPanels, and specialized software oriented toward audio and digital-signal processing and neuroscience research.

LABTECH NOTEBOOK

Laboratory Technologies shipped the first Macintosh version of their popular IBM PC program, LABTECH NOTEBOOK (*13*), in November 1988. It runs on a Macintosh II or higher and can collect data from the Data Translations DT-2211 (the only card supported initially), the GWI MacADIOS II and Jr boards, the IOTECH and SCSI-GPIB interfaces, and the National Instruments NBGPIB, NB-DMA-86, NB-MIO-16, NB-AO-6, NB-DIO-24, and DIO-32 boards. The program costs $995 and includes one driver. Additional drivers cost $150 each (*14*).

Too frequently, when software is "ported" from a character-oriented system such as the IBM PC to a graphics-based system, it is left with a textually oriented user interface and does not take full advantage of the windows and menus of the GUI. Laboratory Technologies did not fall into this trap in their conversion. LABTECH NOTEBOOK makes full use of the graphics-interface capabilities and is easier to use than the MS-DOS version. Acquisition and control channels can be given names instead of numbers. Under Multi-Finder, a dynamic plot of the run can be displayed in real time in a movable, sizable window while the user works on unrelated activities in another window. It is operated from the Macintosh menus and does not require programming. The display screen design option allows the plot areas to be moved and sized with the mouse.

The initialization of the "Setups" (Fig. 1A) and "Schedules" (Fig. 1B) necessary to make a run requires menu selection and interaction with a fair number of windows and nested dialog boxes. This may be confusing to individuals who are not familiar with LABTECH NOTEBOOK operation on the IBM or the Macintosh GUI conventions. Having experience with the Macintosh and the IBM version of LABTECH NOTEBOOK, I found it easy to use. Macintosh "purists" may find the apparently random mixture of windows and dialog boxes irritating, but this does not interfere with functionality or the ease of use of the program. The vendor provides courses at the customer's location for $1000 and offers regularly scheduled IBM LABTECH NOTEBOOK courses.

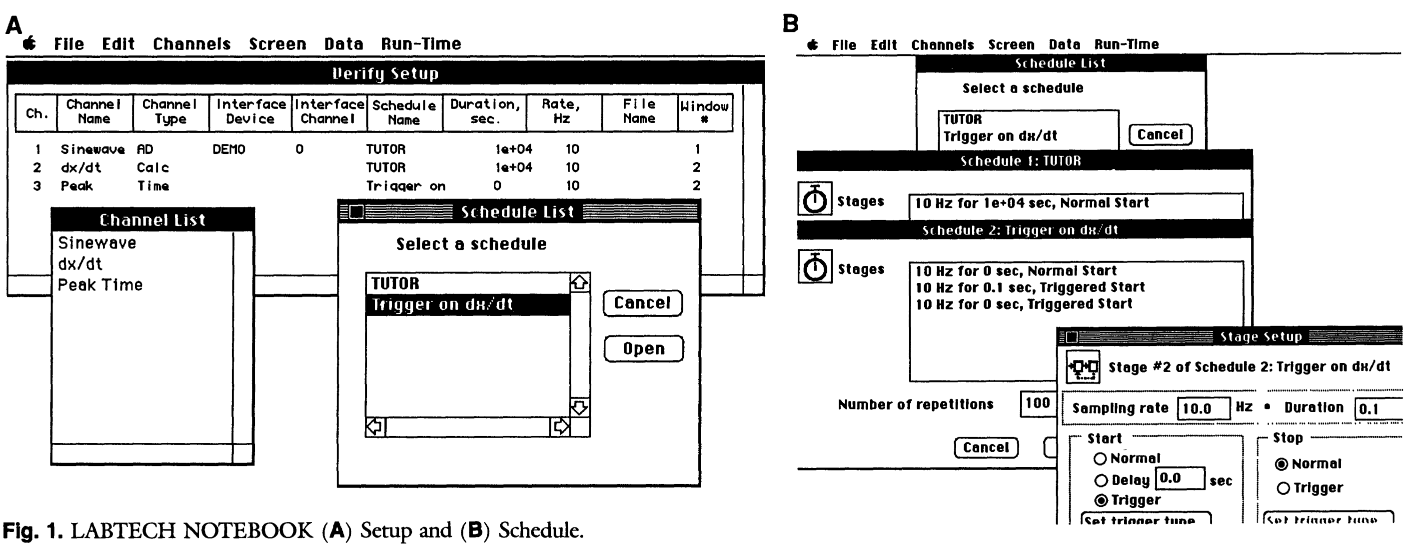


Fig. 1. LABTECH NOTEBOOK (**A**) Setup and (**B**) Schedule.

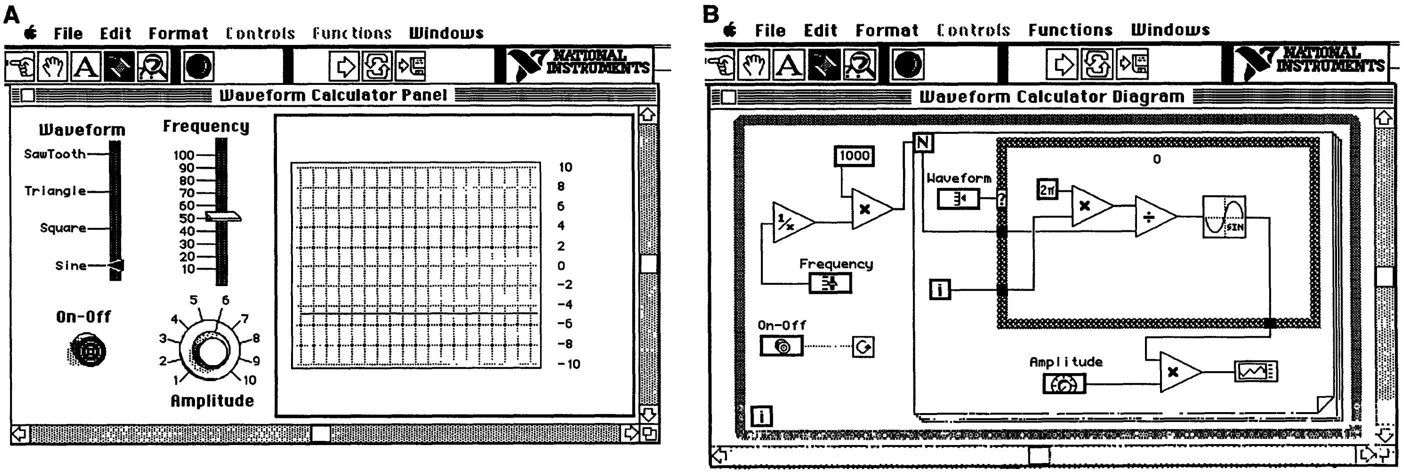


Fig. 2. LabVIEW Virtual Instruments. (**A**) front Panel and (**B**) block Diagram.

Creation of a Setup involves the definition and assignment of the collection, control, and calculated channels, at least one run Schedule, and optional realtime screen displays and data-storage files. Data can be written to an ASCII file with tabs so it can be read with EXCEL. High-speed acquisition can only take place from one board and cannot have real-time display or data storage occurring during the collection. Normal-mode acquisition rates exceed the IBM LABTECH NOTEBOOK rate (>100 Hz). The high-speed acquisition rate is limited by the speed of the converter, and the run duration is limited by the memory size. I have collected from two channels at 20 kHz each for short periods of time (several seconds).

"Schedules" define the various phases of the acquisition with rate, duration, and triggering information supplied for each phase. The schedule can loop indefinitely, allowing the DAC to be used as a function generator. All of the channels can use the same or independent schedules. Schedules can be saved and reloaded independently of the Setup. The Setup is executed by selecting the Go menu option.

Currently, the Macintosh version of LABTECH NOTEBOOK does not have several of the more valuable features of the IBM version: serial support, program shell capability (*15*), and a large selection of independent drivers for a variety of cards and front-end boxes. The version used for this evaluation, shipped in October of 1989, still contained an awkward cosmetic bug (*16*) that has been in the package since it first shipped a year previously. It is very difficult for the manufacturer to test every combination of features in a package of this complexity. Unfortunately, many software houses rely on the users to find these problems. The best defense for a new customer is to ask how many packages have been purchased and get the names of customers who are in their own organization or who are using it in a manner similar to the way they want to—and contact them! It also helps to arrange to have the vendor or a distributor demonstrate the equipment and software in your lab, on your experimental problem. At a minimum you should insist on buying it on approval and testing it before the trial period elapses.

The requirement of a Macintosh II or higher may eliminate this package from some shopping lists. It is, however, a very promising package, and Laboratory Technologies has a very good track record in producing easy-to-use, well-supported, general-purpose laboratory automation software. Since they do not produce a hardware product, their commitment to supporting many interfaces gives their products great flexibility.

LabVIEW

The National Instruments LabVIEW (*17*) data collection, analysis, and instrument control package can be credited with giving high visibility to Macintosh laboratory automation even before the Macintosh II appeared. The unique graphical programming approach of this $1995 package excited a lot of interest when the only I/O options were serial or GPIB communications through the serial or SCSI ports of Macintosh Pluses. It requires 1 megabyte of memory and a hard disk. It also currently supports all the National Instruments Macintosh II data-acquisition and instrument-control boards and the GWI MacADIOS II and MacADIOS Jr boards.

To use an engineering metaphor, LabVIEW programs and routines are called Virtual Instruments. The user interface window is called the "front Panel" (Fig. 2A), the code window is called the "block Diagram" (Fig. 2B), and the code is a wired data-flow diagram. Either the front Panel or the block Diagram can be examined and modified by the user (*18*). Settings on the Panel can be changed before or during a run by manipulating the "Controls," icons that resemble knobs, switches, and slides. The programming is done with icons and "wires." Icons can represent other Virtual Instruments, vendor-supplied drivers, other hierachically nested modular instruments, logical or calculational modules, input (control) modules, and output (display and database) modules.

The analysis options are very complete, featuring statistical, waveform, matrix, vector, signal processing, and curve-fitting libraries. The data flow approach is handled very consistently with the wiring metaphor. There is type-checking, and values can be "bundled" to form structure-like (*19*) data pipelines from icon to icon. Unfortunately, in the current version the wires are inelastic, so the icons cannot be rearranged without "cutting the wires," moving the icon, and rewiring. [This problem is being corrected in the upgrade (LabVIEW 2) that is currently being shipped.] Data flow produces a parallel mode of execution that is very different from normal sequential processes. A "sequence" construct is available to force serial execution where it is necessary. Scheduling is done with the loop or sequence constructs and flags.

Data acquisition with the icon and drivers provided by GWI works, but the documentation is not totally clear and the implementation is complex, so modification of the sample Virtual Instruments provided is not straightforward. The National Instrument cards are probably a better choice (unless you need the excess capacity provided by the extra MacADIOS daughterboards). The new, National Instruments NB-A2000 board [1 MHz, with "direct memory access" (DMA); see below] is an attractive offering for the high-speed field. I have not had a chance to use it, however. Having the drivers written by the software provider is almost always a more desirable option. Hardware is usually relatively static and well specified by the hardware vendor. The software vendor knows the internal requirements of the program and also controls the changes that will be made in the future.

LabVIEW is a very capable data acquisition, control, and analysis package from a highly respected hardware and software manufacturer. The need to "program" to get anything done can be a negative point for some people. Courses are available regularly at the vendor's site in Austin and can be given on location for a fee. Even individuals experienced with traditional programming languages can find the engineering metaphor, the data flow parallelism, the need for Macintosh drawing skills, and the lack of elementary computer-aided software engineering (CASE) tools awkward. The LabVIEW 2 upgrade and the emphasis on user "starter kits" should help address some of these concerns.

Parameter Manager pmPLUS/pmTALK

Genrad Inc. acquired the pmPLUS/pmTALK data-acquisition and analysis packages from Rebus in 1989 (*20*). The pmPLUS/pmTALK combination is necessary to input data from serial and GPIB buses (National Instruments) and the Strawberry Tree and MacADIOS II and MacADIOS Jr boards. It costs $795 and must be purchased from GWI to get MacADIOS support. The analysis-database package alone costs $595. The packages can be used on all Macintoshes from the 512E on but must have at least an 800-kbyte drive. A hard drive is recommended.

The vendor calls pmPLUS a "technical spreadsheet" and the main file, called the Database, looks a great deal like one. Every object (ITEM) about which the user may want to collect data has its own Database. There are six "documents" in the Database: Information, Parameters, Measurements, Notebook, Results, and Figures. The Information "document" stores an ID string, the location of the ITEM, a lock, a picture of the ITEM, a short written Measurement procedure, and the measurement interval. pmPLUS also automatically stores the number of parameters, number of measurements taken so far, time of last measurement, and the disk location of the ITEMS file.

The Parameters "document" allows the definition of the data type, units, source, alarm limits, alert limits, and plotting template for each column of the spreadsheet. The data displayed (Fig. 3A) in the Measurements "document" (spreadsheet) can be entered by hand, come from an ASCII file, be linked relationally to a column in another Database, be a calculated value, or be supplied from the serial port or a board through the pmTALK module. The spreadsheet limits are 256 columns (Parameters) by 32K rows (chronological data points). The Notebook "document" allows the entry of dated observations and text explanations of numerical codes used in the spreadsheet table.

The Results file is a series of slides of graphs and reports. Each slide is the product of an Analysis or Report request. The Analyze menu option allows Trend, Forecast, Hi-Lo Limit, Compressed, Histogram, and Statistical Process Control charts for selected combinations of parameters (columns). Statistics and Correlation reports are also available. A header can be added to the graphs and the slides can be rearranged and printed in a manner similar to that used to manipulate slides in a presentation program.

The pmTALK communications module is accessed from the menu (if it has been purchased with this option). The boards and communications with the serial port are handled with command procedures written in the pmTALK macro language (Fig. 3B). To create a procedure, commands are selected from a scrolled list with the mouse and a dialog box appears to ask for the required parameters (such as sample rate, number of channels, and so forth). The pmTalk version from GWI comes with a sample procedure for accessing the board.

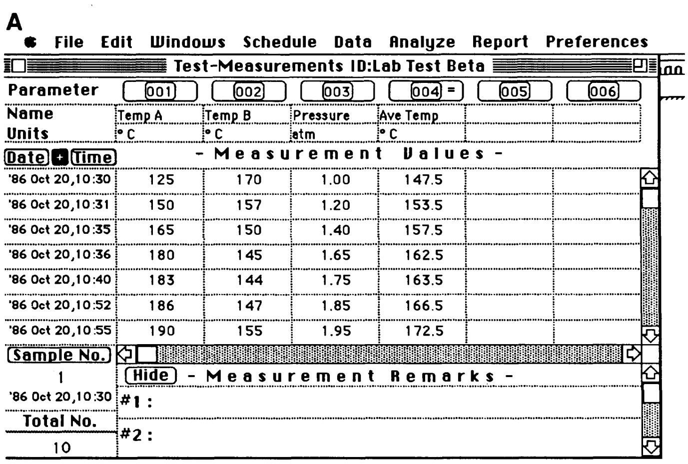
In conclusion, pmPLUS offers many easily obtained riches in the handling of data display and analysis. Although I did collect data from my board and put together several Results files, I have not had the opportunity to test a fraction of its capabilities, but I know it has been used quite satisfactorily with data loggers. Genrad offers phone help in getting communications to work in difficult cases. Special front-end communication procedures can be contracted.

Some of the really unusual pluses of this package are the Database locking and the relational linking to other Databases. pmTALK is oriented toward serial communications and thus toward scientific and process automation. One limitation is that the board drivers for the Strawberry Tree boards and the GWI boards must be acquired independently from the hardware vendors.

Use of Personal Computers for Automation

When personal computers are considered for the collection of irreplaceable data (*21*) or the control of hazardous processes, their

A



B

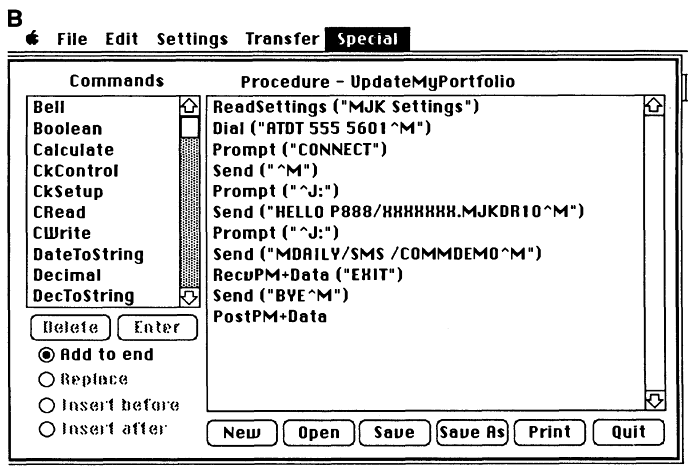


Fig. 3. (**A**) pmPLUS spreadsheet and (**B**) pmTALK macro procedure.

limitations must be kept in mind. They are not redundant, fault-tolerant systems. Almost everyone who has used a PC or Macintosh for a significant amount of time has had the experience of having the system lock up and having to reboot manually. Direct control of critical processes, where injury, costly loss of time, material or equipment, or serious environmental impact are possible, is not prudent. There are a number of good process-control packages for the IBM PC. However, for critical processes, special front-end equipment (*22*) should be used.

There are no process control packages of the same caliber for the Macintosh, yet. When considering the use of the existing packages for control or collection of irreplaceable data, the Macintosh's special limitations should be considered. The Macintosh operating system does not interact with the subsecond world in real time (*23*). The system does not provide preemptive interrupts that allow high-priority events (critical tasks such as data collection or critical error processing) to take over the system on demand. Thus, to have guaranteed timing on data collection or control signals, the signals must be handled by independent, buffered interface boxes or cards. The Macintosh architecture does not support full direct memory access (DMA). It cannot intersperse the DMA and CPU activities through cycle-stealing (slipping data into memory while the CPU is not accessing it), but must "lock out" the CPU temporarily.

Another problem with the Macintosh architecture and operating system is the limited number of serial lines. Apple only supports two, the modem and printer ports. With the printer connected to one port and a modem or network connection (*24*) on the other, there is no room for a connection to an intelligent instrument or front end. Some of the Macintosh data-acquisition packages do not support any serial data collection and control, although almost all support IEEE-488. Apple and third-party vendors have recently released cards with additional serial ports, but they are not currently supported by any of the standard software.

Macintosh software, although usually very easy to use if written to the letter and spirit of the Apple Human Interface Guidelines (*10*), is difficult to program well. The desktop GUI creates some of the difficulty. The special window and menu constructs require experience to manipulate properly (*25*). The programmer cannot control the sequence of events, as in a traditional procedural program, but must wait for, and respond properly to, one of dozens of different actions that the user may perform (*26*). Constraining the user's options (modality) unnecessarily is considered very poor form.

The internal structure of Macintosh programs (*27*) also complicates the task of programming. The program's dialog, window, and menu parameters, icon and font representations, and code are stored as segments called resources. These are moved in and out of the application memory [or heap (*28*)], as needed, to conserve space. Two potential problems arise: the mobile segments are not guaranteed to be found in the same place they were the last time they were used, and memory becomes fragmented as small segments are loaded into spaces left by larger segments (*29*). These situations can cause program crashes if they are not handled properly.

Also, each application (and the system) has a stack in its area where temporary data is stored. If too much subroutine nesting or recursion (*30*) occurs, the stack can grow into the resource area (a stack overflow), write over the code or other resources, and crash the program or system. Knowledgeable and meticulous programming can prevent this in the applications area. Since there is no memory protection on the Macintosh, prevention cannot be guaranteed in the system area. The use of a freshly Restarted system under the Finder can make this an unlikely occurrence.

Amazingly, many fine, reliable programs have been written for the Macintosh. Of course, the reliability necessary for a critical real-time system is different than that required of a low-risk data logging, process monitoring, spreadsheet, or graphics program. It is not a surprise, however, that good, proven Macintosh programmers are in great demand and that Macintosh software often takes a lot longer to be released than expected. The number of Macintosh programs that are released with names that are more cute than descriptive is still rather surprising.

Conclusion

The three packages, LABTECH NOTEBOOK, LabVIEW, and pmPLUS/pmTALK, display the great variety of the offerings available on the Macintosh in the data collection and control area. LABTECH NOTEBOOK has an easy to use menu-based system, a good selection of real-time boards, and some multitasking capability. It is limited to the Macintosh II and up, does not currently have serial support, and has very limited analysis and reporting capability in the package. LabVIEW is a very capable and flexible system, but it requires an unusual type of programming. The vendor sells interface boards. This can be considered an advantage or a disadvantage, depending on whether you favor single-vendor system integration or market competition. Parameter Manager PLUS provides a well-integrated data-monitoring, analysis, reporting, and database package but requires pmTALK and macro-programmed procedures for data acquisition.

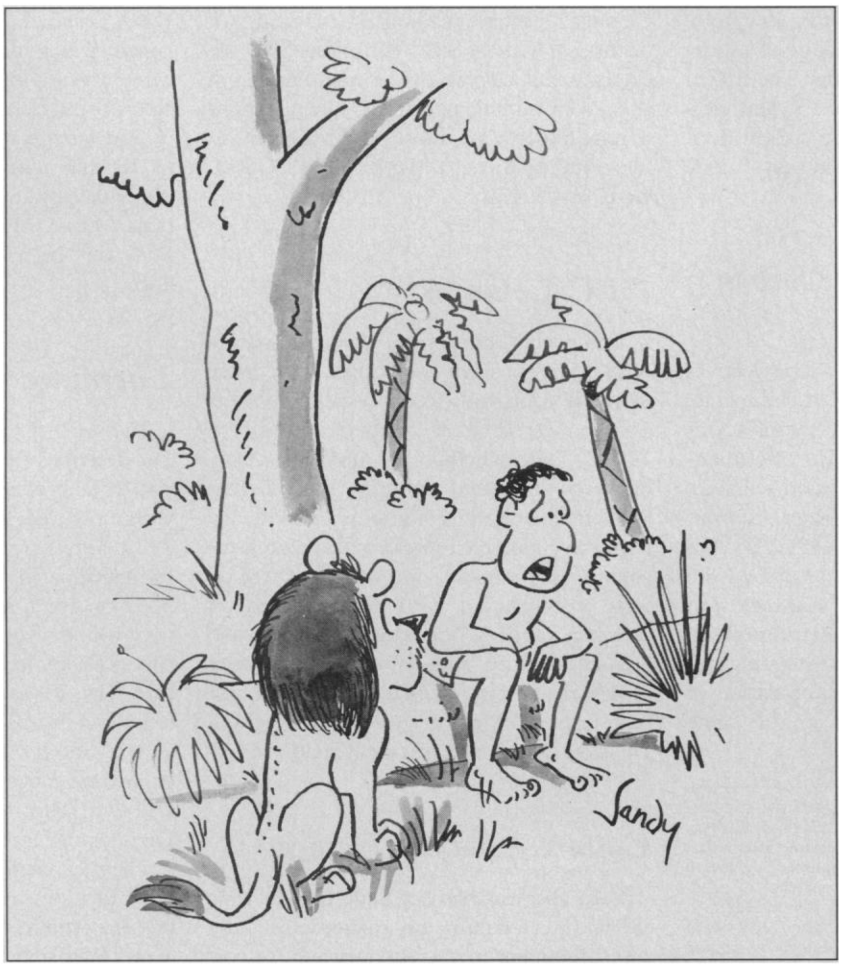
Currently the IBM PC (and compatible) ISA bus machines with PC/MSDOS operating systems dominate the laboratory automation field. IBM's bid for the future of systems under $10,000 is the MicroChannel PS/2 with the OS/2 Presentation Manager. Apple's main thrust is the NuBus Macintosh with the enhanced Apple 7.0 OS. Both offer Unix, although at this stage many personal computer users find Unix to be less than friendly. Suitable GUIs are necessary to sell it to the general market (*31*).

A number of instrument OEMs and VARs have followed IBM down the PS/2, OS/2 path and an equal number have decided not to. Microsoft Windows 3.0, when it is released, may be a safe middle path. The Macintosh is becoming very popular in the scientific office and laboratory. However, the high-end NuBus Macintoshes are still perceived as being too expensive to dedicate to the bench. The restriction of the serial capacity of the computer prevents the monitoring of multiple automated scientific instruments and limits the use of many of the more reliable front-ends. The Macintosh and these packages have presence in some laboratory automation applications and promise for many more. How much more is up to Apple and the software and hardware vendors.

REFERENCES AND NOTES

1. J. Bendat and A. Piersol, *Random Data: Analysis and Measurement Procedures* (Wiley, New York, 1986).
2. Strings of ones and zeros (on-off states). A bit is a single one or zero. A byte is a combination of eight ones or zeros. Word size varies from computer to computer. A word may be multiple bytes (8, 16, or 32 bits) or fractional combinations of bytes (12, 20, or 36 bits).
3. D. Sheingold, *Transducer Interfacing Book* (Analog Devices, Norwood, MA, 1980).
4. A. J. Diefenderfer, *Principles of Electronic Measurement* (Saunders, Philadelphia, 1972).
5. H. V. Malmstadt, Enke, Crouch, Horlick, *Electronic Measurements for Scientists* (Benjamin, Menlo Park, CA, 1973).
6. A bus consists of digital information-transmission hardware and protocols. Computers have (i) internal buses (such as ISA, EISA, MCA, NuBus, Q-Bus, and UniBus) accessed by slots inside the computer case that will accept printed circuit boards (cards) and (ii) standard external buses (such as serial/RS-232C, parallel/Centronix, GPIB/HPIB, and SCSI), which are linked by cables (wires with terminal connectors) through ports (connectors) on the outside of the cases of the different pieces of equipment (computer, printer, modem, or instrument).
7. In this case the Hewlett-Packard noncompatible desktops are classified as specialized laboratory computers.
8. The "Gang of Nine" includes Wyse, AST Research, Tandy, Compaq, Hewlett-Packard, Zenith, Olivetti, NEC, and Epson. See L. B. Glass [*Byte* **14** (no. 12), 417 (1989)] for a discussion of EISA.
9. E. Fenton, *The Macintosh Font Book* (Peachpit,

Berkeley, 1989).
10. *Apple Human Interface Guidelines: The Apple Desktop Interface* (Addison-Wesley, Menlo Park, CA, 1987).
11. M. Seyer, *Complete Guide to RS-232 and Parallel Connections* (Prentice Hall, Englewood Cliffs, NJ, 1988).
12. GW Instruments, 35 Medford Street, Somerville, MA 02143.
13. Laboratory Technologies Corporation, 400 Research Drive, Wilmington, MA 01887.
14. The National Instruments version only supports drivers from that vendor.
15. A shell is an external program written around the run-time system in BASIC, C, or Pascal that allows customizing of the user interface and extended calculational capabilities.
16. The bug is that the new screen design windows would not size until design window exited and reentered.
17. National Instruments Corporation, 6504 Bridgepoint Parkway, Austin, TX 78730.
18. This can be done unless it is locked or programmed in a language other than "G."
19. A structure is a C language–composite data type that is similar to a Fortran record. See S. G. Kochan, *Programming in ANSI C* (Hayden, Indianapolis, 1988).
20. Genrad, Inc., 510 Cottonwood Drive, Milpitas, CA 95035.
21. The replaceability of data often depends on how recently it was collected. Loss of an experimental run because of equipment failure or operator error can often be compensated for if the problem is noticed immediately. A day, a week, or a year later, the resources needed to do the experiment may no longer be available, or conditions may have changed too much for the replacement to be validly included in the original data set. The rapid analysis allowed by computers, which can point out anomalies immediately, can be beneficial in this case, but increased mechanization can increase potential points of failure. Unattended operation can reduce immediate and critical evaluation of and compensation for unexpected experimental contingencies.
22. Front-end hardware includes single-loop controllers, transient recorders, programmable logic controllers (PLCs), and pressure or temperature override or relief hardware (consult your local process engineer for a process-hazards review).
23. Apple has a real-time system called A/ROSE that can be run in place of the desktop or on a coprocessor board. It is not currently supported by any of the standard software packages, however.
24. T. Bove and C. Rhodes, *The Well-Connected Macintosh* (Harcourt, Brace, Jovanovich, Orlando, FL, 1987).
25. Presentation Manager, MicroSoft Windows, and X-Windows programmers also experience this.
26. These include a variety of mouse, menu, window, and text operations.
27. M. Read, *Macintosh Programming Primer* (Addison-Wesley, New York, 1989).
28. S. Chernicoff, *Macintosh Revealed* (Hayden/Apple, Berkeley, 1985), vols. 1 and 2.
29. S. Knaster, *How to Write Macintosh Software* (Hayden, Indianapolis, 1988).
30. Recursion is when a subroutine or procedure calls itself; see P. L. Anderson and G. C. Anderson, *Advanced C* (Hayden, Indianapolis, 1988), p. 67.
31. "(The Mac has) the most important user interface today on the desktop," L. Dooling (of AT&T Unix) [*Byte* **14** (no. 12), 30 (1989)].



"Of course I'm depressed. I just found out I was adopted."