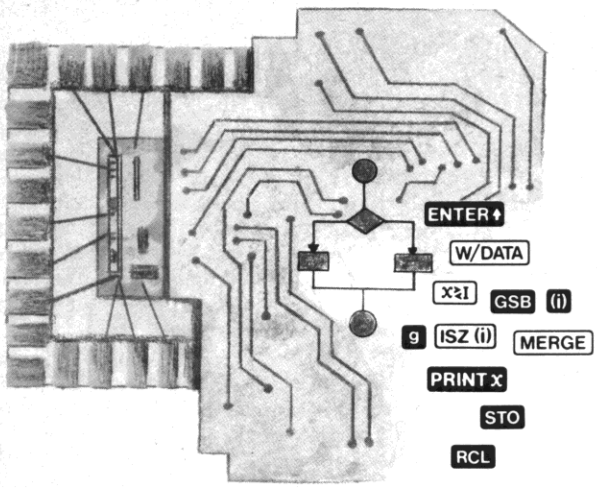


Featuring, this issue:

HEWLETT PACKARD

January-February 1982 Vol. 6 No. 1
ISSN 0730-2037

HP Key Notes

Published Quarterly | By Subscription: $5/yr. | **$2.00**

HP-IL Introduction

On December 21, 1981, the *Hewlett-Packard Interface Loop* concept became a reality. This date marked the beginning of a new era in low-cost systems for electronic data transfer and electronic control.

Now, your HP-41 hand-held computer can easily communicate with, and control, any Hewlett-Packard Interface Loop (HP-IL) peripheral. Present HP-IL peripherals include a printer, a digital cassette drive, a converter, and a digital multimeter.* But that's not all. A video interface has been announced but will not be available until later this year. Also, with the new HP 82938A Series 80 HP-IL Interface Card, your HP Series 80 Personal Computer can interact with your HP-41 through HP-IL, and it can act as a *controller* of the HP-IL loop. And, with the addition of the new HP 82182A Time Module, your HP-41 becomes a care-free, *time-based controller*. Ah, but this is jumping ahead; let's discuss the fundamentals.

The HP-IL concept revolves around a two-wire cable that can connect as many as 30 HP-IL devices in a closed series loop. HP-IL is a "master-slave" interface, meaning that one active controller regulates the information flow between all of the devices in the loop. Data and commands move around the loop in one direction.

The reason we use such terms as "the controller" and "a device" is because HP-IL is a *general-purpose interface*. The elements of the system that you choose to use will be tailored specifically to your application. This is the flexibility of the HP-IL concept.

At the core of this new concept is the new HP 82160A HP-IL Module. This module fits into any of the four HP-41 ports, and it expands the HP-41 with a powerful set of functions for interaction with the devices in the loop. Now, printers, mass-storage devices, and a variety of instruments can be controlled by the HP-41 through one port!

** This digital multimeter is not available through HP Calculator Dealers. Contact your local HP Sales Office for more information and prices.*

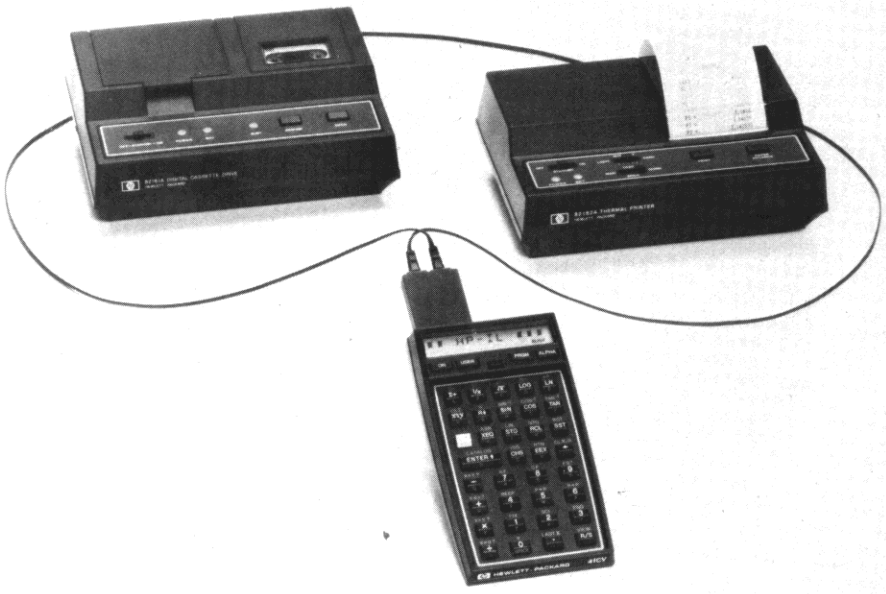
THE CASSETTE DRIVE

The new HP 82161A Digital Cassette Drive is an HP-IL mass-storage device. Each of the digital mini-cassettes driven by this device can store up to 131,072 bytes of data, programs, status information, and directory in files that *you* name. Just think: *131,000 bytes is over 50 times the full RAM capacity of the HP-41CV!* In fact, we have stored on just ¾ of one mini-cassette, **all of the programs in every HP-41 "Solutions" book!** This is why we call it *mass* storage. Also, the HP 82161A Digital Cassette Drive is fast, with an average file access time of 13 seconds, and a 30-second maximum rewind time. Yet, accessing a program file is easy. With the Digital Cassette Drive in the loop, simply place the name of the file in the ALPHA register, key in [XEQ] "READP", and the program is available in the RAM of your HP-41 within seconds. Other files are accessed (and recorded) just as easily by executing different HP-IL commands.

THE PRINTER

The new HP 82162A Thermal Printer/Plotter is an HP-IL compatible device that is similar in appearance and function to the HP 82143A Thermal Printer/Plotter. In addition to *all* of the convenient features of the HP 82143A Thermal Printer/Plotter such as double- and single-wide printing, right and left justifying, 128 standard characters, and special character building, the new HP-IL printer has a FORMAT statement that allows you to automatically center words on the page and to easily right- and left-justify two columns. Plus, the new HP 82162A Thermal Printer/Plotter has a 101-element buffer, and a STANDBY position of the on-off switch helps conserve battery power.

(Continued)



All prices in this newsletter are suggested retail prices excluding applicable state and local taxes—Continental U.S.A., Alaska, and Hawaii.

THE MULTIMETER AND THE CONVERTER

With HP-IL, your HP-41 can control, and take data readings from, the new HP 3468A Digital Multimeter. Or, if you have a considerable knowledge of electronics and interfacing techniques, you can use the new HP 82166A HP-IL Converter to connect your own 8 or 16 bit-parallel device into the HP-IL system.

A general application of such an HP-IL system would use the new HP 82182 Time Module, which is discussed in the following article, to "wake-up" the HP-41 in the middle of the night, while you are at home sleeping. Automatically, then, the HP-41 would begin to execute a control program. HP-IL allows you to program the HP-41 to power-up devices in the loop, take readings from instruments, make data listings, read subprograms from mass storage devices, and interact with other devices in the loop as prescribed by the programs. The extensive programming capabilities of the HP-41 allow it to make critical decisions based on the data received from the instruments in the loop and interact with other devices in the loop, accordingly. Then, with its job complete, the HP-41 can power-down the devices in the loop, including itself. When you arrive at work in the morning, after a good night's rest, a printer listing awaits you, documenting the events of the night as monitored by the HP-41!

SEE YOUR HP DEALER...

At this point, you have read the introduction to an entirely new concept in electronic communication and control that represents a major technological breakthrough. This is a good point to sit back, relax, and think about the implications of the HP-IL concept. Just think: now, with the HP products that were announced on December 1, 1981, thousands of electronic control applications can be tackled by portable, affordable, devices.

And, this is only the beginning. Hewlett-Packard is committed to continuing its quality support of HP-IL. The devices that are compatible with the HP-41 as a mainframe controller will be compatible with future hand-held computers and personal computers with input/output capabilities. Many HP divisions—in the instrument and computer areas—will be introducing HP-IL devices.

We have enough space to give you only a small taste of the broad capabilities of HP-IL. See your local HP Dealer for a look at the new HP-IL system, and see if it will meet *your* challenge; we'll bet it can—and does!

Three New Modules Introduced

Along with HP-IL, Hewlett-Packard introduced three new modules for the HP-41 System. These modules serve to expand the conveniences of the HP-41, both on its own and as a mainframe controller of HP-IL.

TIME MODULE

The new HP 82182A Time Module gives you the ability to incorporate precise time and date into your calculations. With this module, your HP-41 can be a clock, a calendar, a stopwatch, an alarm, and there's much more.

You can set up to 253 separate alarms in the HP-41 (depending on the amount of memory available), and any alarm can be set to repeat itself again and again after a desired time-interval. An alarm will activate whether or not the calculator is on. And, when you set an alarm, you can choose to make it either a tone alarm, a message alarm, or a control alarm. When a control alarm activates, the HP-41 "wakes up" and immediately begins execution of a specified program. This feature makes the Time Module a valuable component of *any* HP-41 controlled HP-IL system.

The Time Module uses a quartz crystal for dependable accuracy. And, with the module's "built-in" programmable accuracy factor, your HP-41 can "fine tune" your Time Module, thus giving it a time-keeping ability that will amaze even the most scrutinizing of accuracy buffs.

There is no other hand-held computer like the HP-41. And now the HP 82182A Time Module takes the HP-41 one more step forward.

EXTENDED FUNCTIONS/ MEMORY MODULE

The new HP 82180A Extended Functions/Memory Module expands your HP-41 with 47 new functions and 889 bytes of "extended" memory.

Such new functions as "programmable size" and "programmable assign" allow you to change the memory configurations and keyboard-function assignments under program control. Plus, there is a "SIZE?" function that returns the present number of allocated data registers to the X-register. We know, from the number of "SIZE?" routines that you have contributed to KEY NOTES, that this is a popular function. Other popular functions include the "register-swap" and "register-move" functions that help you to easily manipulate blocks of data registers.

Other new "memory-management" functions give you access to, and control over, extended memory. This extended memory brings file-structure and file-management capability to the HP-41. Three types of files can be created in extended memory: program files, data-storage files, and ASCII files. Now, the HP-41 can store and manipulate ASCII files, thus giving it a communication link with larger computers.

There are 889 bytes of extended memory in the HP 82180A module, and the memory-management functions contained in this module also give you access to even more memory housed in the new HP 82181A Extended Memory Module.

EXTENDED MEMORY MODULE

Each new HP 82181A Extended Memory Module adds 1,666 bytes of extended memory to the HP-41. Used together with a new HP 82180A Extended Functions/ Memory Module, up to two HP 82181A Extended Memory Modules may be added to the HP-41, equipping the computer with 4,221 bytes of extended *Continuous Memory*. With the 2,233 bytes of Random Access Memory in an HP-41CV (or HP-41C with a Quad RAM), you can have a total of *6.4K bytes of pocketable, Continuous Memory!!* (An HP 82180A Extended Functions/Memory Module must be in your HP-41 before it can interact with an HP 82181A Extended Memory Module.)

SEE YOUR HP DEALER...

But, we can't possibly describe the full advantages of every one of these new modules. We suggest that you spend some time talking with your local HP Dealer to get a better idea of what can be done with these new and exciting products. You'll soon realize that the range of applications is nearly endless. Only *your* imagination will impose restrictions on what can be accomplished with the new modules... and HP-IL.

Don't forget to keep your address up to date. Send changes to the address on the back cover.

Corvallis Library Corner

The Users' Library is a service to HP-67/97/41 programmable calculator owners. As a member of the Users' Library, one of your major benefits is that you are in communication with HP calculator users all around the world. Programs that are the result of many hours of brain-work by fellow HP calculator users are yours at an almost insignificant price. And, although anyone may order programs from the Users' Library, there are countless advantages to becoming a member. Read on ...

LIBRARY SUBSCRIPTIONS (CORVALLIS)

There aren't many investments you can make that give you an immediate return of over twice the invested value, but a subscription to the Users' Library is an investment that does just that, and better. In the United States and Canada, the fee for a one-year subscription to the Users' Library is $20.* This $20 fee places you on the mailing list to receive the *Catalog of Contributed Programs* ($10) and one update, and the Programmers Reference Guide (available March 1: a $10 value). Plus, you will presently receive a coupon good for four free programs, each program valued at $6. And, for the year of 1982, HP KEY NOTES will be sent to you free ($5 value) if you are a member of the Users' Library. Now, if you get out your HP calculator and total these figures, you will find that for an initial investment of $20, you get a package with a monetary value of $49. And the monetary value is only a small part of the true value of being a Users' Library member.

If you live outside the U.S. or Canada, the fee for a one-year subscription to the Corvallis Users' Library is $30* because of considerably higher postage and handling charges. Though you don't double your investment here, you still have everything to gain by becoming a Users' Library member.

And, you have a choice of how you renew your subscription. You can pay either the full subscription price ($20 in the U.S. and Canada, $30 elsewhere) and receive a coupon for four free $6 programs ($24 value), or if you don't need the four free programs, you pay only half of the original subscription price ($10 in the U.S. and Canada, $15 elsewhere).

You can subscribe to the Users' Library by using the form on page 15 or you can charge it to your major credit card over the phone. (Call 800-547-3400, except from Alaska and Hawaii; in Oregon call 503-758-1010.)

The greatest advantage of Library membership is being able to choose from a large collection of software that is immediately ready to serve you in your application. The savings in time and effort are worth much more than the small membership fee.

You also should look into the possibility of the Library fee being income tax deductible. You might even be able to deduct the cost of some purchased Library programs.

ORDERING PROGRAMS

HP-67/97 and HP-41 programs featured in KEY NOTES are now available from both the Library in Corvallis and the Library in Geneva. **Readers in Europe should order from Geneva (address on back cover) to get quicker service.** Readers elsewhere should order from Corvallis, where programs cost $6* each and each program includes documentation and prerecorded magnetic cards; for HP-41 programs, this $6 price includes bar code.

Whenever possible, use the Users' Library Order Form in your *Catalog of Contributed Programs* to place orders for programs you see in KEY NOTES. If you do not have an order form, a plain piece of paper with your name and address and the program numbers you desire is certainly adequate. Make certain that your address is legible and complete.

Mail your order and a check or money order to the Corvallis or Geneva address shown on the back cover of KEY NOTES. Don't forget to include your State or local taxes. Or, in the U.S., you can place your order by calling toll-free: 800-547-3400, except from Alaska and Hawaii (in Oregon call 503-758-1010).

When ordering from outside the U.S., **attach your payment to your order.** Much time is wasted and orders are held up, trying to match orders and checks that are sent in separately. Your payment can be in the form of an International Money Order, a Foreign Draft, or the equivalent. *Any payment must be in U.S. dollars, drawn on a U.S. bank,* otherwise it will be returned to you. Another option for payment is to use such major credit cards as American Express, VISA, or MasterCard.

Orders are usually shipped within 2 working days after they are received in Corvallis. However, if you need a program yesterday, call us today at 503-757-2000, extension 3371. Although we can't get it to you yesterday, we'll do our best to get it in the mail today.

SUBMITTING PROGRAMS

Programs submitted to the Users' Library should be on Hewlett-Packard standard Library submittal forms, or they should include, at least, the documentation required by those forms. To maintain the high quality of the programs submitted to the Users' Library, we encourage you to

(Continued)

* *U.S. dollars. Orders from anywhere outside the U.S. must include a negotiable check (or money order), in U.S. dollars, drawn on a U.S. bank. All orders from anywhere outside the U.S. must include an additional 10 percent fee for special handling and air mail postage. (For example, an order for two programs = $6 × 2 = $12 + $1.20 = $13.20 total.) If you live in Europe, you should order KEY NOTES programs directly from the Geneva UPLE, but make certain you make payment as required by Users' Program Library Europe; the above $6 fee is good only for orders to the Corvallis Library.*

closely follow the *Users' Library Contributor's Guide for the HP-41, HP-67, and HP-97.* Complete and orderly documentation is essential to ensure the acceptance of a program into the Library.

We also encourage you to read the ongoing KEY NOTES column, "In the Key of HP." This column addresses some of the things we look for when we are reviewing programs that are submitted to the Users' Library.

Programs that are submitted to the Library for the HP-67 or HP-97 must include magnetic cards, and HP-41 programs must include either magnetic cards, **reproducible** bar code, or a data mini-cassette for use with the new HP 82161A Digital Cassette Drive. (The cassette will be returned to you.) It would take far too long to review and check all the many program submittals if we had to key them in line by line. Also, there is always an increased chance of error when someone keys in handwritten keystrokes.

The management of the Users' Library reserves the right to reject programs which, in its opinion, do not represent a significant contribution, are not clearly or sufficiently documented, or are not otherwise appropriate for the Library.

CURRENT LIBRARY NEWS (CORVALLIS)

January 1982 marks the tenth anniversary of Hewlett-Packard personal computing products. Although the Users' Library is a little younger, we are going to celebrate in a big way! Our most exciting news is for our authors. Now, they can make "points" with the Library! For every submitted program accepted by the Library, we will award the author *one point.* These points are redeemable for Hewlett-Packard accessories AND calculators. One point may be redeemed for a Solutions Book, one mini data cassette, or a custom keyboard—just to name a few of the items available. One point is also redeemable for the traditional "four free $6 programs coupon," or save the points and redeem them for such large items as an HP-41CV or even an HP-85 Personal Computer!! The complete list of products available for the "point" system will be sent out with each accepted submittal. (This list is available from the Library upon request.)

NEW LIBRARY CONTEST

To further celebrate our tenth anniversary, the Library is sponsoring another contest. Beginning March 31st, and through August 31, 1982, the Library will award eight new HP 82182A Time Modules and either two new HP 82161A Digital Cassette Drives or two new HP 82162A Thermal Printer/Plotters (including HP-IL Modules) to the authors of the ten programs judged most outstanding each month by our review committee. The programs will be judged on merit with the same guidelines as with the last contest. For documentation forms, or further information, contact the Users' Library.

NEW TAXES SOLUTIONS BOOK

As unpleasant as tax time can be, your HP-41 is again equipped to help you through the paper-work. The *HP-41 1981 Taxes Solutions Book* (00041-90403) will be available February 1 and will cost $12.50.* Contact your local HP Dealer, or send your order directly to the Users' Library. **If you order through the Library, please include a $3.50 Handling Charge** and any applicable State and/or local taxes.

ORDER STATUS ETC. (CORVALLIS)

The Library is remaining current on orders and coupons, thanks to a great staff (that always seems to include ambitious temporary help). Remember, call the Library directly—503-757-2000, extension 3371 NOT TOLL-FREE—and your order will ship that same day (credit card or purchase orders only).

Our quantity discount offer was so popular that we are extending it through our tenth anniversary year. Order six or more Library programs, and deduct 25% (Library programs only).

Programs are now available from the Library recorded on a data mini-cassette for use with the HP 82161A Digital Cassette Drive. The mini-cassettes are $9.50* each, with a program order from the Library, plus the price of each program requested. (For example: One cassette plus one program = 9.50 + 6.00 = $15.50.) Take advantage of the *quantity discount* and fill a cassette today. Each cassette will hold up to 250 programs, depending on program length.

The Library is here to support Hewlett-Packard programmable calculator owners. If you have suggestions for future Library services, let us know.

HP-11C AND HP-12C LIBRARY (CORVALLIS)

A Users' Library for the HP-11C and HP-12C slim-line calculators is just around the corner. You will be reading more details about the establishment of this new library in upcoming issues of HP KEY NOTES. Please do not write or call until we publish details, as those details are still not firm.

NEW PROGRAM PRICES

Inflation affects all of us, and the Users' Library is no exception. Our cost of reproducing programs has gone up like everything else. But, rather than make a sweeping increase in price on all of the programs in the Library, we have decided to graduate the prices according to our reproduction costs. Many programs will still be $6.* The more lengthy programs will be price accordingly at $8, $10, $12, etc.* Prices will be listed in each *Catalog.* This is the fairest pricing method that we know of, and we know you'll agree that the convenience of having all that information available is well worth the small price of each program.

NEW PROGRAMS (CORVALLIS)

Here are some recent submittals to the Corvallis Users' Library. All of the programs featured in this issue are available worldwide, but before you order, be sure to read (above) "Ordering Programs." And, remember that where additional Memory Modules are listed as necessary to run a program on the HP-41, you do not need them if you are using an HP-41CV or a Quad-RAM.

(41) Particulate Removal Cost #01100C (Price: $8*)

The costs are determined for cleaning particulate matter from stack gas by electrostatic precipitation or fabric filters. Inputs include cost parameters, gas flow rate, percent particulate removal, fabric air-to-cloth ratio, and gas temperature. The flyash resistivity, ESP plate area, bag life, and baghouse pressure drop are estimated. *Required accessories: 3 Memory Modules, and a printer is helpful.* (646 lines, 1618 bytes, 16 pages)

Author: **Norman S. Charles**
Houston, Texas

* * * * * * * * * * * * * * *

(41) Smooth Earth Diffraction #01085C (Price: $8*)

The Smooth Earth Diffraction program predicts the radio propagation loss relative to free space for smooth earth conditions, or for conditions of uniform roughness, of a path slightly beyond the radio horizon. This program uses a simplified method, developed by Vogler (1964), for the standard Van der Pol-Bremmer diffraction theory, as modified by Norton (1941). The program has a self-check for validity. Inputs include data on the effective earth radius. *Required accessories: 1 Memory Module.* (356 lines, 707 bytes, 27 pages)

Author: **John L. Roth**
Newbury Park, California

* * * * * * * * * * * * * * *

(41) Architectural Perspective #01120C (Price: $6*)

This program calculates two-point perspective views. Data may be input directly, or up to 239 data points may be stored in memory and/or written on magnetic cards. Data is grid-based; input and output grids may be dissimilar. Heights may be input as grid values or as elevations (in feet or meters). *Required Accessories: 1 Memory Module.* (269 lines, 476 bytes, 12 pages)

Author: **Daniel R. Tindall**
Minneapolis, Minnesota

* *U.S. dollars. Orders from anywhere outside the U.S. must include a negotiable check (or money order), in U.S. dollars, drawn on a U.S. bank. All orders from anywhere outside the U.S. must include an additional 10 percent fee for special handling and air mail postage. (For example, an order for two programs = $6 × 2 = $12 + $1.20 = $13.20 total.) If you live in Europe, you should order KEY NOTES programs directly from the Geneva UPLE, but make certain you make payment as required by Users' Program Library Europe; the above $6 fee is good only for orders to the Corvallis Library.*

(41) Flipo #01477C (Price: $12.00*)

If you liked the game program "Reversi," you will flip over this one. Flipo is an implementation of a board game, best known in the U.S. as OTHELLO® which is marketed by Gabriel Industries, Inc. This program offers you many options. For instance, there are three playing modes: no printer connected, printer connected and ON, and printer connected and OFF. And, you can change the playing mode at will during the game.

This program allows you to review the board and the last two plays in any playing mode, and it offers you five playing levels. The code has been optimized for execution time; a game can be completed in as little as 15 minutes, depending on the playing mode and your response time. *Required accessories: 3 Memory Modules.* (612 lines, 1,792 bytes, 40 pages)

Author: **Robert Swanson**
Portland, Oregon

LIBRARY CONTEST WINNERS ANNOUNCED

In KEY NOTES V5N2 we announced the Users' Library **Program Submittal Contest**. In the following issue, V5N3, we announced the winners for the month of September. In this issue, we have the winning programs for October and November.

As you know, three programs were to be chosen on merit each month. You will notice that four programs are featured as the winners for the month of November. Two of these programs were contributed by the same author, and we chose to feature both of them. Here, then, are the winners for October 1981.

* * * * * * * * * * * * * * *

(41) Analysis of Laboratory Strength Test Data #67000-99952 (Price: $10*)

This package is made up of nine interrelated programs that deal with the analysis of data obtained from laboratory rock strength tests. Three programs take the Mohr-Coulomb failure criterion to determine angle of friction and cohesion. Three programs deal with nonlinear failure analysis of intact or jointed rock. One program develops the Mohr envelope, one determines material constants, and one tabulates principal stresses. Programs can be run together without re-entry of data. Programs run with or without the printer; a neat printout format is used. For those with only 3 Memory Modules, a shorter program is described. *Required accessories: 4 Memory Modules.* (862 lines, 1796 bytes, 43 pages)

Author: **John L. Gilby**
Maidenhead, Berkshire, England

(Congratulations, Mr. Gilby! You have done an excellent job of documenting this program. Your example problems are clear and concise and it is handy that you have included data-review and error-correction routines. This program is a winner, for sure—Ed).

(67/97) Gear Mesh Design-Spur Gears #67000-99953 (Price: $10*)

Given a minimum of basic gear information, this program calculates the data required to design a meshed pair of full-depth involute spur gears. Outputs include testing pitch ratio, backlash, contact ratio, distance between centers, and more. Also, this versatile program allows the user to manipulate the design through various inputs in order to customize the gear pair. Any potential design errors are flagged for user decision. This program does not work on the HP-41. (411 lines, 39 pages)

Author: **Michael R. Cascini**
Cedar Rapids, Iowa

(Congratulations, Mr. Cascini! You thought of all the angles on this program. You have included many pages of clearly stated, typed documentation. You defined all your terms and included a flow chart of the program. Also, this program is compatible with either customary U.S. units or Metric units. It is easy to see why this program won. Anyone with a knowledge of gear mesh design will want this program in their HP-67/97 program library—Ed.)

* * * * * * * * * * * * * * *

(67/97) Sheep Production Model #67000-99954 (Price: $10*)

This program simulates both physical and financial aspects of sheep flock management. This allows users to compare differing management strategies and different flock types and, from this information, operate their flock more efficiently. This program does not run on the HP-41. (1116 lines of programming on 5 cards and over 80 pages of documentation)

Author: **John Pauley**
Tasmania, Australia

(Congratulations, Mr. Pauley! It is obvious that hours and hours of work went into preparing this splendidly documented program. You have included flow charts, term definitions, and example problems. You have a table of contents, appendices, the whole works. This is an unquestionable winner. Any sheep-farmer or agricultural extension officer will jump at the chance to have information like this at his or her fingertips—Ed.)

* * * * * * * * * * * * * * *

Now, we have the Users' Library **Program Submittal Contest** winners for November 1981.

* * * * * * * * * * * * * * *

(41) Circular Plate with Hole-Plate 2 #67000-99948 (Price: $10*)

This program will provide the user with displacement, slope, and stress calculations at any radius on a circular plate with a center hole, *considering the outside edge free and the inside edge either simply supported or fixed.* The user may apply a uniform pressure load of any band-width, a ring load, or a combination of both anywhere on the plate. *Required accessories: 4 Memory Modules and Printer.* (1034 lines, 1683 bytes, 22 pages)

(41) Circular Plate with Hole-Plate 1 #67000-99949 (Price: $10*)

This program is nearly the same as -99948 (above). It performs the same function on a circular plate with a center hole, *considering the outside edge either simply supported or fixed and the inside edge free. Required accessories: 3 Memory Modules and Printer.* (815 lines, 1405 bytes, 20 pages)

Author: **Steven A. Porter**
Houston, Texas

(Congratulations, Mr. Porter! Both of these programs have been well thought out. Your documentation is neatly typed and includes some very explicit drawings. Your example problems are clearly stated and diagrammed. These programs are, no doubt about it, winners. Any mechanical engineer would be at a loss without these programs—Ed.)

* * * * * * * * * * * * * * *

(41) Physical Property Estimation #67000-99950 (Price: $10*)

This program will calculate estimates of the point properties—critical temperature, critical pressure, critical volume, and acentric factor. Plus, the temperature correlated estimates of liquid density, heat of vaporization, vapor pressure, and surface tension, and the liquid and vapor properties of viscosity, and heat capacity also are calculated. Inputs required are the molecular weight, normal boiling point, and a knowledge of the molecular structure. *Required accessores: 4 Memory Modules.* (987 lines, 2087 bytes, 34 pages)

Author: **Robert Wooley**
Midland, Missouri

(Congratulations, Mr. Wooley! It would take someone with a broad expertise to come up with this interesting and useful program. Your documentation is very clear, plus, you have included extensive flowcharts and tables. I'm sure that the thousands of Chemists and Chemical Engineers will appreciate all the hours of work that you put in to develop this program—Ed.)

* * * * * * * * * * * * * * *

(41) Buttress Design #67000-99951 (Price: $10*)

Cut slopes in bedrock having an adverse direction of bedding (dip out of slope) are commonly stabilized through the construction of a fill material buttress. Soil parameters and slope dimensions are sized to prohibit translational failure parallel to the planes of bedding. *Required accessories: 4 Memory Modules.* (1006 lines, 1799 bytes, 33 pages)

Author: **Stephen Milazzo**
Long Beach, California

(Continued)

(Congratulations, Mr. Milazzo! This is a very well documented program. Your diagrams are interesting and explicit. You have clearly defined all your terms and abbreviations. Any civil engineer will be able to recognize the enormous amount of work that you have saved them by developing this winning program. Keep up the good work—Ed.)

The Users' Library was swamped with an extraordinary number of program submittals in December. At the time that this issue was going into production, neither the December contest winners nor the **Grand Prize** winners of the three marvelous **HP-85 Personal Computers** had been determined. We will feature these winners in the May 1982 issue (V6N2), so please bear with us until then ...

Custom Keyboards Announced

Hewlett-Packard recently announced the introduction of the HP 82504A Custom Keyboard. Its flexible membrane completely covers the existing HP-41 keyboard and provides the final, professional touch in customizing the remarkable HP-41 calculator.

This new Custom Keyboard can be printed with any customer-designed nomenclature, including logos, trademarks, alphanumerics, and other special symbols. And up to five colors can be printed on each Custom Keyboard.

Obviously, we cannot economically produce such products as the new Custom Keyboard in small quantities of one or ten or even fifty. This new product is a special order item, and it comes in minimum quantities of 250. For more information about this exciting new product, contact your nearest Hewlett-Packard Sales Office.

(By the way, you can get a single Custom Keyboard—with the standard key configuration—by earning 1 point from the Users' Library!—Ed.)

In the Key of HP

Most of the ideas presented in KEY NOTES are contributed by you, and they represent the ways in which you approach solutions to programming problems. This column addresses some of the common inconsistencies that the Users' Library finds in user-submitted programs. Also in this column, we answer some of the common questions that you ask, and we present some ideas that we think you will find useful.

In this issue, **John Loux**, a Technical Advisor in the Users' Library discusses application pacs (ROM) for the HP-41 and the utility of the **COPY** function. Plus, we have added some handy tips near the end of the column.

PROGRAMS IN ROM

HP-41 application pacs and custom program modules are general-purpose packages written either by Hewlett-Packard or independent groups. Their programs and functions are permanently written in plug-in modules. These ROM-resident programs can be accessed and used in much the same way and for all of the same reasons that user-written programs can be used. Although, for all user-intensive purposes ROM (read only memory) programs are very similar to their RAM counterparts, they also differ in some important ways.

The acronym RAM stands for "random access memory" and is used most often when referring to the read/write memory of a calculator or computer. Loosely, random access means that information stored anywhere within memory can be accessed with equal ease. All user-alterable aspects of the HP-41 (programs, data, flags, the current **SIZE**, etc.) are maintained in RAM.

ROM is similar to RAM in that it can be randomly accessed and is capable of maintaining the same kinds of stored information. Although ROM is capable of supporting many kinds of information, its *read-only* nature limits its usefulness in maintaining such information as data and status. Consequently, related functions are not supported. The two types of information that are supported by ROM are user-language programs and micro-code* functions.

Except for certain aspects of their internal representations, ROM programs are identical to RAM programs. ROM programs can be run, single-stepped through, printed on a printer, and can be randomly access via the **GTO** (go to label) and **GTO.** (go to step) commands. ROM programs cannot be traced,** cannot be edited or deleted, cannot be positioned-to by using CATALOG 2, and cannot be packed. In short, you can do nothing with a ROM program but run or view it while it still resides in ROM.

Because ROM is internally different from RAM, the HP-41 references it differently. One manifestation of this difference is seen in how the execution of a ROM program is represented *in program memory*. The programmer must key in [XEQ] [ALPHA] followed by the label characters and [ALPHA] just as would be done if a RAM alpha label were accessed, but before the calculator stores the keystrokes as a program line, it searches its memory for the label in the order of the three catalogs; i.e., **CAT 1**—internal RAM, **CAT 2**—external ROM, and **CAT 3**—standard functions (internal ROM). If the corresponding label does not exist in RAM but does in ROM, the calculator translates the **XEQ** command into an **XROM** command before storing it as a program line. **XROM**'s are provided so that the calculator will know exactly where to look for the label when the execution is performed, thereby *decreasing access time.* The ROM number and the function number within the ROM are recorded in the **XROM** statement and can be seen if the program step is viewed with the required ROM removed. The user will note that the corresponding process does NOT occur if a programmatic **GTO** is performed to a ROM label. Thus, every time such a **GTO** is encountered, the calculator must search CATALOGs 1 and 2, taking more time. Coupled with the fact that the **GTO** function will leave the program pointer in ROM (a generally undesirable occurrence), it becomes obvious that the use of **GTO** to access ROM programs should be limited.

The other kind of information that is supported by ROM is the microcode function. These functions are virtually identical to standard (**CAT 3**) functions, and are NOT programs. They are invoked in the same fashion that standard functions are, via an **XEQ** or key assignment, and their mnemonics are displayed with no preceding **XEQ** or **XROM** when their respective ROM's are attached. Function mnemonics will revert to **XROM**'s if their resident ROM's are not plugged-in to the calculator. Functions can be distinguished from programs in CATALOG 2 by their lack of the text character (ᵀ) which precedes each program label.

COPY

There will be times when a ROM program will not be useful as written but would be if it were modified. In this instance, having a modifiable copy of the program in RAM is desirable. Fortunately, the HP-41 supports this action with the standard function **COPY**. The **COPY** function is not programmable.

When executed, **COPY** prompts for a global label resident in the ROM program of interest. The input label need not be the first nor the only global label in the program. The function then makes an identical copy of the entire ROM program in RAM (if there is room), beginning at step

* *User language is that which the user implements when steps are keyed in program mode. Microcode is machine language; that low-level language that the machine actually implements when it performs an operation. Normally, the user cannot access or write functions in microcode.*

** *Tracing is the ability of all HP-41 compatible printers to list the executed steps and important intermediate results of a running program.*

01 and terminating with the **END**. The copy will never write over an existing RAM program, because *an **END** is always loaded before the incoming program.* The **END** will be stored in program memory even if the preceding program is already terminated by an **END** and even if there is no preceding program. These solitary **END**'s can be positioned-to using CAT 1 and can be deleted just as you can delete any program step.

The resulting RAM copy is identical to the ROM version in terms of the functions represented, including any **XROM**'s that may (and probably do) exist in the ROM program. Thus, the RAM version is still ROM-dependent. This may or may not be desirable, depending on your needs. For example: if the program that is copied acts as a controller and calls ROM routines to do certain subordinate tasks, alterations to the in-RAM program may be made as long as they are compatible with the running of the called subroutines. On the other hand, if you desire to alter or write your own subroutines, you should replace the **XROM** program calls with appropriate **XEQ** statements that call your own routines. Because **CAT 1** is searched before **CAT 2** when a subroutine call is performed, your customized routine may have the same label as the ROM program label you are replacing.

Note that if you desire to **COPY** and run all or part of a ROM program without having the ROM physically attached, all **XROM**'s involved should be translated to **XEQ**'s and all called subroutines should be **COPY**'ed. Note also that any function (as opposed to program) calls must also be replaced by calls to user-written routines. If such a programmatic **XROM** is encountered without the ROM in place, the calculator will display "NONEXISTENT." To verify that this occurrence is the result of a forgotten **XROM**, simply put the calculator into program mode at the point at which the program stopped, and read the displayed step.

If the display "RAM" occurs, then the program you have just tried to **COPY** has either already been loaded into RAM (perhaps as a portion of a program that was already **COPY**'ed using another of its global labels) or another program of the same name already exists in RAM and uses the same global label.

THE USERS' LIBRARY

COPY'ed programs are acceptable to the Library as long as they represent a significant change from the original program. **COPY**'ed programs should be documented as such, with the ROM name and any duplicate labels given prominence. Highlights of the differences between the contributed program and the ROM program should also be given.

Programs that make use of ROM routines as subroutines are also acceptable. The ROM functions and programs that are used should be listed and the ROM requirement must be given under the "Necessary Accessories" heading.

Following, now, are the tips we promised at the beginning of this article.

ENTER 100 ÷

Q: What is the shortest and fastest way to divide by 100 in a program?
A: There are three frequently used methods for dividing by 100.

1. 100
   /
   length: 4 bytes
2. 1E2
   /
   length: 4 bytes
3. 1
   %
   length: 2 bytes

Method number 2 is faster than method number 1, but method number 3 is the shortest and fastest way to divide by 100 in a program. You can use this "percent" method to divide by other numbers, too. For instance, 1000 ÷ becomes .1 %; 50 ÷ becomes 2 %; and, 500 ÷ becomes .2 %.

ADD A LOCAL LABEL

Many of the routines and programs that we receive in the mail have in them **GTO** statements that call global labels in the same program. For example, we wrote this simple counting program.

```
01 LBL "COUNT"
02 VIEW X
03 1
04 +
05 GTO "COUNT"
```

This routine can be shortened by 4 bytes, and the execution time noticeably reduced, by *adding* a local numeric label (00-14) and replacing that *global label* **GTO** in line 05 with a *local label* **GTO**. Choosing to use LBL 01, we have:

```
01 LBL "COUNT"
02 LBL 01
03 VIEW X
04 1
05 +
06 GTO 01
```

It is always advantageous to use local labels when making jumps within a program. Global labels should be reserved for jumping between programs and for making routines and programs accessible from any part of program memory.

In short, don't use **GTO** "a global label" in a program if that global label appears in that program. Instead, add a local label before or after the global label and use: **GTO** "the local label."

For more details on this subject read V5N2, "In The Key of HP."

Corvallis Library Now Accepting "SP"

Your Corvallis Users' Library will now accept well-documented programs that contain Synthetic Programming ("SP") lines. **But before you rush to your files to submit such programs, make sure you read and understand the following reservations.** And keep uppermost in your mind those initial words: "well-documented."

WHAT IS IT?

"Synthetic Programming," as applied to the HP-41C or HP-41CV, is the use, in programs or manual operation, of any of the following: synthetic functions, synthetic key assignments, and nonstandard alpha strings and labels. The word "synthetic" derives from the user-initiated synthesis of HP-41 instruction bytes into combinations that are not available with normal keystrokes.

As a simple example of what we have thus far discussed, consider nonstandard alpha strings. Each HP-41 instruction byte has a corresponding alpha character. The ALPHA keyboard has keys available for 59 of these characters. However, as users of the HP 82143A Printer/Plotter function BLDSPEC can attest, the HP-41 display is capable of forming additional characters. And so, using Synthetic Programming techniques, any of 19 additional characters can be included in alpha displays, program text lines, and global label names. Furthermore, any of the 128 printer characters can be represented in a program text line, which can result in a significant savings in program bytes through reduced use of ACSPEC and of flag 13. But that's a simple example. A more elaborate example of Synthetic Programming follows...

HOW DOES IT WORK?

The program "BYTE" provides a quick and precise means of computing the number of bytes in a user program (or between any two points in program memory), without use of the printer. At each of two different program lines, the user presses a key that he/she has assigned to the synthetic function "RCL b" (which recalls the value of the internal program counter to X). Then, "BYTE," in an execution time of less than 3 seconds, returns to X the number of bytes between the two program lines. This very useful result is impossible to obtain using normal HP-41 functions.

Look carefully at lines 10, 11, 12, 13, 26, and 32. Of this group, line 11 is a nonstandard program text line, as described above; line 32 is simply a normal "1 E3" program line, shorn of the superflous "1" (thus saving a byte); the rest are new two-byte lines, usually called "synthetic functions." These functions provide access to the so-called "system scratch registers," which are a group of 16 registers (track 1 of a card reader WSTS card) including the

(Continued)

user RPN stack, the ALPHA register, the program counter and subroutine return stack, the 56 user and system flags, key assignment information, and the current memory allocation data. For example, line 13 causes the exchange of the contents of the X-register with those of the d-register, which contains the 56 user and system flags. The name "d-register" comes from the display of the associated functions. The "x<>" prefix and the "d" postfix—seen normally only in "LBL d"—have been combined into a synthetic function. There is no normal function postfix associated with the program code that results in the "[" in line 10—the use of the printer symbol "[" is just an accident. (In the HP-41 display, line 10 shows as "STO M.") Lines 13 through 26 take a number from X, swap it with the contents of the flag register, change its value by manipulating individual binary bits (i.e., by setting and clearing flags), and finally restore it to X while returning all flags to their original states.

```
01♦LBL "BYTE"    20 FS?C 18
02 XEQ 01        21 SF 17
03 X<>Y          22 FS?C 19
04 XEQ 01        23 SF 18
05 -             24 FS?C 20
06 CHS           25 SF 19
07 RTN           26 X<> d
08♦LBL 01        27 10
09 CLA           28 *
10 STO [         29 INT
11 "├♦♦♦♦"       30 LASTX
12 X<> [         31 FRC
13 X<> d         32  E3
14 FS?C 15       33 *
15 SF 13         34 DEC
16 FS?C 16       35 7
17 SF 14         36 *
18 FS?C 17       37 +
19 SF 15         38 END
```

There are numerous other examples of the capability of Synthetic Programming to expand the already impressive power of the HP-41. This makes it desirable for the Users' Library, in its role of providing a medium for users to exchange programming ideas, to accept programs containing synthetic code. However, the decision to do so had to be made with great care, because it is not possible that HP can "support" the use of Synthetic Programming in the traditional manner that HP supports its proprietary products. HP takes pride in the reliability of its calculators and the ease of their use, guaranteeing that each calculator will function and perform exactly as described in its corresponding owner's handbook. The operation of synthetic program lines cannot be assured in this traditional HP manner; first, because the corresponding internal execution processes have not been studied or documented and second, because there has always been the chance that future improvements to the HP-41 would eliminate some or all Synthetic Programming capability. Furthermore, HP-Corvallis does not have the personnel resources to provide written or telephone instruction to customers in the mysteries of Synthetic Programming. Finally, there are "occupational hazards" associated with Synthetic Programming; misuse of synthetic functions can result in loss of memory contents or temporary system lock-up, problems which HP traditionally takes great pains to remove from their calculators. (Note: These problems can cause inconvenience, but there is no physical danger to the calculator.)

Therefore, in consideration of all of the above facts, the Corvallis Users' Library will accept HP-41 programs containing synthetic code, but only with the following reservations:

1. This new policy of Hewlett-Packard (HP) does NOT constitute an endorsement or recommendation of Synthetic Programming. Hewlett-Packard will not "support" Synthetic Programming, neither by guaranteeing that synthetic functions will operate on all present and future HP-41's, nor by providing instruction in Synthetic Programming to customers. HP will, however, refer customers to independently available references on Synthetic Programming (see below).
2. Programs containing synthetic code will be specifically and clearly identified in the Users' Library *Catalog* or in KEY NOTES, to warn potential users that the code cannot be keyed into their calculators by using standard techniques. However, customers not familiar with Synthetic Programming techniques will be able to use the card reader or wand to enter the programs into their calculators.
3. Program authors must include in their program documentation a clear and precise identification of all synthetic program lines, including specifications of the byte structure of the lines. And all inquiries about the programs will be referred directly to the authors.

WHERE CAN I GET MORE INFORMATON?

In KEY NOTES V4N3p8 we reviewed the book: *Synthetic Programming on the HP-41C,* by **Dr. William C. Wickes**. The book describes the complete theory and application of Synthetic Programming, starting from the beginner's standpoint. This 96-page book is $11.00 postpaid, by surface mail, anywhere in the world. For air mail, add: U.S., Mexico, and Canada $1.00; for Europe and South America $2.00; for elsewhere $3.00. Order it from Larken Publications; 4517 N.W. Queens Avenue; Corvallis, Oregon 97330 U.S.A. Make sure your check or money order is in U.S. dollars, drawn on a U.S. bank.

Another source of information about Synthetic Programming is the *PPC Calculator Journal,* published by the independent users' club, PPC. For more information about PPC and a sample issue of the club's newsletter, send a self-addressed, large (folded) envelope (9 × 12 inches; 23.8 × 30.5 cm) with first-class postage for 2 ounces (56.7 grams) to: *PPC Calculator Journal;* 2545 W. Camden Place; Santa Ana, California 92704 U.S.A. If you live outside the U.S., make sure you include a legible address label and international postal coupons for 56.7 grams (2 ounces). A letter is not necessary and will only slow the response.

A third reference for Synthetic Programming is the book reviewed in KEY NOTES V5N3p9. It is: *Calculator Tips & Routines Especially for the HP-41C/41CV,* and it was edited and compiled by **John S. Dearing**, a member of PPC and a resident of Corvallis, Oregon. The book is 136 pages and mainly a fabulous collection of tips and routines from past KEY NOTES, HP manuals and books, the *PPC Calculator Journal,* and other sources. It includes routines that contain synthetic code, but it is not a treatise on Synthetic Programming. The book is $15 postpaid to the U.S., Canada, and Mexico; $20 air mail postpaid elsewhere. Make sure your payment is a check or money order in U.S. dollars, drawn on a U.S. bank. Order it from: Corvallis Software, Inc.; P. O. Box 1412; Corvallis, Oregon 97339 U.S.A.

If you are in a hurry to get one of these two books, try calling your local college bookstore or your local HP Dealer before you order by mail. Some of these stores are now carrying calculator books.

"Free" One-Year Subscriptions ...

At one time, several years ago, we *did* offer a "free one-year subscription" to HP KEY NOTES. However, for many reasons, that offer was soon changed to read: "KEY NOTES is published periodically and is presently free to you, an HP-41 owner." Or, in the case of earlier calculators, it states: "... to the owners of HP-67 and HP-97 calculators."

We bring this to your attention because of the recent addition of a subscription fee for KEY NOTES. Some readers have written to question us about the balance of their "free one-year subscription," and we have had to remind them that the card formerly packed with each new card-programmable HP calculator does NOT offer a "one-year" subscription.

Today, and in the foreseeable future, the only free subscription to KEY NOTES is the one offered if you are a member of the Corvallis Users' Library in 1982 or if you *join* the Users' Library in 1982. We have not made commitments past 1982 because of rapidly changing postage, freight, and printing costs. Perhaps later this year we can be more specific about multi-year subscriptions at reduced rates. But don't worry, because we will surely let you know of any changes in KEY NOTES.

The Issue on Back Issues

Because many new owners of HP-41 hand-held computers want to retrieve the information presented in earlier issues of KEY NOTES, we will make back issues available starting March 1, 1982. However, because it is economically and physically unfeasible to reprint all of the back issues of KEY NOTES, we are going to offer from Volume 3 Number 3 to Volume 5 Number 3 as relevant back issues. The V3N3 issue introduced the HP-41C, so coverage back to that point should satisfy most new subscribers to KEY NOTES.

Below are the contents of each available back issue, followed by a price schedule.

V5N3 SEP-DEC 1981 (16 pages)
- Library Corner
- First Contest Winners Announced
- Toward More Secure Bar Code
- PPC Conference Held Here
- Library Contest Continues
- In the Key of HP (HP-41 Functions)
- Book Reviews
- Back Issues of KEY NOTES
- Tenth Anniversary Calendar Ready
- Routines, Techniques, Tips, Et Cetera
- KEY NOTES Subscription Plan

V5N2 MAY-AUG 1981 (16 pages)
- Library Corner
- Do You Really Know Your HP-67/97?
- HP-41 "Bush Computer"
- Library Contest Announced
- More Software = More Solutions
- In the Key of HP (Discussion on Labels)
- Magnetic Card Erasure
- Petroleum Fluids Pac Released
- Routines, Techniques, Tips, Et Cetera
- Tenth Anniversary Calendar Announced
- KEY NOTES Subscription Plan

V5N1 JAN-APR 1981 (16 pages)
- Library Corner
- KEY NOTES Going to Subscription
- Quad RAM Questions
- Generating Bar Code
- HP-41 Flags—Part 2
- Custom Services Sells Solutions
- Routines, Techniques, Tips, Et Cetera
- HP-41 Subroutines
- HP-41 Function List

V4N3 SEP-DEC 1980 (12 pages)
- Library Corner
- "Petals Around the Rose"
- HP-41C Flags—Part 1
- Is the HP-65 Dead?
- "Roll" Your Own Bar Code
- "I Owe It All to My HP"
- It's "That" Time Again!
- Book Reviews
- "25 Words" (More or Less!)
- New Products, New Prices

V4N2 JUN-AUG 1980 (12 pages)
- Wand Functions
- New HP-41C Power Source
- Library Corner
- How Small Can You Write?
- Efficient Use of HP-41C Status, Data, and WALL Cards
- A Special Program (HP-67/97)
- The Wizard of Programming
- Book Reviews
- "25 Words" (More or Less!)

V4N1 MARCH 1980 (12 pages)
- Important HP-41C Information
- Fitting 67/97 Programs Into the HP-41C
- "25 Words" (More or Less!)
- Russian Calculators?
- Library Corner
- About Batteries ... Continued
- Accessories Hot Line
- Indirect Addressing
- Book Reviews
- We Get Letters

V3N4 NOVEMBER 1979 (12 pages)
- Library Corner
- HP-41C Tips and Techniques
- (67) Twenty-Element 4 × 5 Matrix
- Book Reviews
- HP-41C Tips From an Owner
- Randomly Yours
- We Get Letters
- "25 Words" (More or Less!)
- HP-41C Owner's Handbook Addendum

V3N3 AUGUST 1979 (12 pages)
- The HP-41C Defined
- A Users' View of the HP-41C
- The Designer's View
- HP-67/97 Compatibility
- Software for the HP-41C
- Library Corner
- Another Excellent "Calculator" Book
- Random Numbers, Means, Regressions etc.
- About Batteries
- More About Merging (HP-67/97)
- "25 Words" (More or Less!)

For the convenience of "newcomers," the column "25 Words (More or Less!) was the forerunner of the present column, "Routines, Techniques, Tips, Etc." These columns contain programming information for the HP-67/97 and the HP-41.

Prices for KEY NOTES back issues are as follows. All prices include first-class or air mail. Payment must accompany your order and must be a check or money order in U.S. dollars drawn on a U.S. bank. Or you may use your American Express, VISA, or MasterCard account; be sure to include your account number and card expiration date. Your order will be promptly mailed in an envelope.

NO. OF ISSUES	U.S., MEXICO CANADA	ALL OTHER COUNTRIES
1	$2.00	$3.50
2	$3.50	$5.00
3	$5.00	$6.50
4	$6.00	$8.00
5	$7.00	$9.00
6	$8.00	$10.00
7	$9.00	$11.00
8	$10.00	$12.00

Remember, these back issues will NOT be available until March 1, 1982.

Do *you* have an application for an HP-67/97 or HP-41 that is unusual, clever, or interesting? We'd like to hear about it.

KEY NOTES Corrections

Before the V5N3 issue of KEY NOTES, our collection of stamps from Ireland and Scotland was a little low. But, thanks to an error that appeared on page 12 in column a, we now have plenty of Irish and Scottish stamps. It is great to know that KEY NOTES is read *carefully* by so many people throughout the world, especially those who know where Loch Ness *actually* is located!!

Also, in column a on page 12, there is a typographical error in the second contribution by **Roland Waldi**. The trigonometic keystrokes for the algebraic function $X/\sqrt{1-X^2}$ should be [SIN^{-1}]; [TAN]. And, when you are using these simplified trigonometric expressions, remember that X must be less than 1 if $1-X^2$ appears in the algebraic expression or if your calculator is set to radians (RAD) mode.

In **Vally Lambrechts** routine (p 14a), line 11 should be the card reader function RSUB. That XROM 30,04 is the XROM code for this function, as you can find on page 32 of your *HP 82104A Card Reader Owner's Handbook.*

Cube-Puzzlers Rescued!

In the last issue (V5N3p16) we told you about, and showed you, "Kolb's Kube." (Got a lot of comments about it, too!) So what does one do for an encore to that remarkable creation? Fortunately, we received an absolutely remarkable HP-41 program that "solves" Rubik's Cube. (We have several, but this one is superb.) And if you recognize the author's name, you'll realize he is one of the winners of the Users' Library **Program Submittal Contest**, reported elsewhere in this issue.

Congratulations, **Mr. Gilby**, your program not only works but also shines as a living example of excellent documentation. Here's the abstract.

(41) Rubik Cube Solution #01342C (Price $12*)

Rubik's magic cube has taken the world by storm, recently. This program solves the famous cube from any position. The user enters the initial colours of the faces and the HP-41 proceeds to solve the cube, using three subroutines. The output is a series of instructions informing which face should be rotated. The program is fully illustrated, and the notation used is explained so that it will now be simple to solve the cube. This solution will not break any world records, however, because it is relatively slow. *Required accessories: 4 Memory Modules and Card Reader (Printer optional); and, of course, a Rubik Cube!* (1213 lines, 1918 bytes, 36 pages)

Author: **John L. Gilby**
Maidenhead, Berkshire, England

** U.S. dollars. See note at bottom of page 3.*

(Ton yhW)$^{-1}$

Why not sell your software through Hewlett-Packard Dealers?

- Can you identify needed professional solutions?
- Can you write quality software?
- Do you have the resources (time, media duplication, order taking, brochures, etc.) to sell your solutions in volume through HP Dealers?
- Can you support your product after the sale?

If so, you may qualify for the HP PLUS Program (KEY NOTES V5N2p5c). First Principles Software did. First Principles is a new, full-time venture (about a year old) for **Pat Imbimbo**, a registered engineer. His seven Civil Engineering programs for the HP-41 deal with such problems as hydrology and retaining wall construction. Pat is looking to build on these initial programs to offer a range of software in Civil Engineering and construction. (See the new Users' Library *Catalog* for more information on these programs.)

HP PLUS software was developed by an independent software supplier for operation on HP computer systems. The supplier is solely responsible for its software and support services. HP is not the manufacturer or co-developer of such software or support. HP disclaims any and all liabilities for and makes no warranties, expressed or implied, with respect to this software. Distribution of this product or information concerning this product does not constitute endorsement of the product, the supplier, or support services. The customer is responsible for selection of the software he or she purchases.

HP is interested in *all* professional HP-41 solutions. Of particular interest are wand-based inventory, cassette-based statistics, sales management, in-flight aviation, taxes, and agriculture. Interested parties should contact:

Jack Peters (Dept. 5360)
Hewlett-Packard Company
1000 N.E. Circle Blvd.
Corvallis, OR 97330
(503) 757-2000 Ext. 2207

Editorial

If you are reading KEY NOTES for the first time, welcome to our elite group. If this is your first "paid-subscription" issue, thank you for your faith in, and support of, KEY NOTES. We have some interesting surprises for you in 1982, and I am certain that you will find your investment in KEY NOTES a wise choice.

Mr. Edward R. Bettinardi of Denver, Colorado, tried a recommendation from V5N1p6c and found to his dismay that it didn't work. We are grateful that he took the time to inform us of his discovery. In the referenced issue, we reported on an eraser kit that removed "permanent ink" from magnetic cards. By "permanent," we *meant* "India ink," used in drawing and drafting trades. Unfortunately, that eraser kit removes *only* India ink. It will not touch such inks as that used in Sanford and some Pilot and Schwan felt-tip pens. When they named *those* inks as "permanent," they were serious! Thanks, again, Mr. Bettinardi, for telling us about that problem.

LETTERS TO KEY NOTES

When you address letters to KEY NOTES, you should refrain from including anything not associated with the newsletter. Questions about the calculator or its operation should be addressed to Customer Support and questions about the Users' Library should be addressed to that function. Also, questions about future products cannot be answered; Company policy permits me to discuss only those products that have been released. Federal regulations also prohibit discussing future products.

Letters to the editor should be addressed to:

Henry Horn, Editor
HP KEY NOTES
Hewlett-Packard Co.
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 U.S.A.

We cannot guarantee a reply to every letter, but we do guarantee that every letter will be read by the editor or technical editor (**Ted Wadman**), and as many as possible will be answered in KEY NOTES or in a personal response. Please be sure to put your return address on the face of your letter. Letters sometimes get separated from envelopes.

READER FEEDBACK

You send us many letters that extol the virtues of KEY NOTES, so we know that you *like* the newsletter. However, never hesitate to write and tell us about what you *would like* to see in a future issue—or issues. We are open to all and any suggestions.

HP-67/97 Routines and Tips

This idea came to us some time ago from **William Pinnick** of Columbus, Ohio. It is applicable to many situations.

(67/97) One of the beautiful aspects of the HP-67 and HP-97 calculators is the ability to increment and decrement the I-register to control program flow and looping. This ability is significantly enhanced by indirect increment statements and indirect decrement statements for additional looping control.

As an example, I believe I have found the lowest possible number of statements necessary to control three levels of looping—only 18 steps! Assuming *all loops begin at the same value* and count down, a unique exit is achieved by the DSZ step for the three counters: I, primary register 0, and secondary register 0. The positions of these three counters make them uniquely suited for this purpose; thus, the program is:

```
*LBLA     STO0
 STOE    *LBL1
 STO0     STOI
*LBL0    *LBL2
  P≠S
```

Beginning after LBL2, key in the body of the nested loop, making calculations based on the changing values of the I-register, primary register 0, and secondary register 0. The I-register changes most rapidly, primary register 0 second, and secondary register 0 changes most slowly. After the calculations, close the loops with:

```
          GTO1
 RCLE      P≠S
 DSZI     DSZi
 GTO2     GTO0
 DSZi      RTN
```

The initial counter value must be in the X-register when the nested-loop begins.

This routine was sent to us from Benton, Illinois, by **Robert M. Blake**. It refers back to the last issue of KEY NOTES (V5N3).

(67/97) In response to **David Whyatt's** request (V5N3p12b), perhaps the following routine will be beneficial.

```
001  *LBL9     014   X>Y?
002   STOI     015   GTO9
003      4     016    CLX
004    X≠I     017   DSP9
005   X<0?     018    RTN
006   DSZI     019  *LBL9
007   ENT↑     020      -
008    ABS     021   STOI
009    LOG     022   ISZI
010   X<0?     023  *LBL8
011   GTO8     024     R↓
012    INT     025   DSPi
013   RCLI     026    RTN
```

The major advantages of this routine being it is shorter by 11 program lines. Also, it needs only two labels, one storage register (I), and it does not use flags. The time it takes to return a decimal number is equal to Mr. Whyatt's; however, whole numbers are returned much quicker.

(The purpose of this routine is to format data in the form F5.X, meaning only 5 spaces will be occupied by the digits, the sign (if there is one), and the decimal point. Thus, 26.8842 is displayed as 26.88, −26.8842 is displayed as −26.9, and 2.68842 is displayed as 2.688. The format assumes X to be less than 10,000 and at least 0.00005. The field length can be changed by changing line 003; for instance, F7.X would require line 003 to be 6—Ed.)

Routines, Techniques, Tips, Etc...

The routines and techniques furnished in this column are contributed by people from all walks of life and with various levels of mathematical and programming skills. While the routines might not be the ultimate in programming, they do present new ideas and solutions that others have found for their applications. *You might have to modify them to fit your personal application.*

It is probable that you will find this next routine to be a useful addition to your catalog of programs. It was submitted by **Serge Drogi** of Stavold, France.

(41) Just a word to the HP-41 users. It is possible to calculate a probability with the Binomial Law [(?): in french "loi Binomiale"] without using any register outside of the stack. So, this program will be a good example of the advantages of RPN to all users.

The expression is:

$$P(k) = \frac{n!\, p^k (1-p)^{(n-k)}}{k!*(n-k)!}$$

It is interesting to search for what happens in the stack. At the beginning the stack must be set up with; X:k, Y:p, Z:n and at the end, the answer, P(k), is in X.

```
01*LBL "BINLAW"     16 CLX
02 X<>Y             17 LASTX
03 STO T            18 -
04 X<>Y             19 FACT
05 Y↑X              20 ST/ Y
06 X<>Y             21 CLX
07 LASTX            22 LASTX
08 RDN              23 1
09 FACT             24 R↑
10 ST* Y            25 -
11 CLX              26 X<>Y
12 LASTX            27 Y↑X
13 R↑               28 *
14 FACT             29 .END.
15 ST/ Z
```

* * * * * * * * * * * * * * *

Here is a routine that will please those of you who own a printer. It is a simple but useful routine that was sent to us from Piacenza, Italy, by **Steve Tendon**.

(41 with printer) This "special-build-special" routine makes special character building easy. First of all, I have named each dot of the print column (see page 62 of your printer manual) with a letter, like this:

A□
B□
C□
D□
E□
F□
G□

Instructions for use are as follows:

1. Load program.
2. USER (on); ASN "SPBLDSP" and "BLDSPEC" to, for example, SIN and COS, respectively.
3. Press SPBLDSP.
4. Press the letter-keys that correspond to the dots you want to darken (if it is a blank column skip this step).
5. Press BLDSPEC.
6. Repeat steps 3, 4, and 5 until all columns have been input; print the special character with ACSPEC, ADV; or with ACSPEC, PRBUF, etc.

```
01*LBL "SPBLDSP"    15 GTO 00
02 0                16*LBL E
03 STOP             17 16
04*LBL A            18 GTO 00
05 1                19*LBL F
06 GTO 00           20 32
07*LBL B            21 GTO 00
08 2                22*LBL G
09 GTO 00           23 64
10*LBL C            24*LBL 00
11 4                25 ST+ Y
12 GTO 00           26 RDN
13*LBL D            27 STOP
14 8
```

John Hendricks, who lives in San Carlos, California, developed this quick and useful routine to enhance your HP-41 printouts.

(41 with printer) This routine converts and prints, with an HP 82143 Printer/Plotter, a degrees, minutes, and seconds X-register value to the form DD°MM'SS.F" without utilizing any storage registers.

For HHhMM'SS, substitute 104, ACCHR for lines 08 through 15.

```
01*LBL "PRDMS"      24 *
02 CF 28            25 ENTER↑
03 CF 29            26 INT
04 FIX 0            27 CLA
05 INT              28 ARCL X
06 ACX              29 ACA
07 LASTX            30 CLX
08 2                31 39
09 SKPCOL           32 ACCHR
10 6                33 RDN
11 ACCOL            34 FRC
12 9                35 1 E2
13 ACCOL            36 *
14 ACCOL            37 SF 28
15 ACCOL            38 SF 29
16 RDN              39 FIX 1
17 ACCOL            40 CLA
18 RDN              41 ARCL X
19 SKPCOL           42 ACA
20 RDN              43 34
21 ABS              44 ACCHR
22 FRC              45 PRBUF
23 1 E2             46 .END.
```

Here is a routine that is 83 bytes long, but has the potential of saving many registers in a program that handles a lot of data. It is similar to a routine, by **Leonard Cordwell**, that we printed in V5N2. This version was sent to us by **Nai Chi Lee** of Stony Brook, New York.

(41) This routine can pack positive integers from 0 to 999,999. (However, due to round-off error in LOG, the two numbers must not both be 999,999). It is 83 bytes long, but can be reduced by using local labels. To pack, place number abcdef in X and uvwxyz in Y. Then XEQ "IN." The resulting number in X is of the form: a.bcdefuvwx Eyz.

If the number uvwxyz is less than 100,000, it is normalized by adding 1E5 to it. This is flagged by the negative sign.

To unpack, just place the "combined number" in X and XEQ "OUT." The original two numbers are returned to X and Y respectively.

```
01*LBL "IN"         26*LBL "OUT"
02 1 E5             27 ENTER↑
03 /                28 ABS
04 X<>Y             29 LOG
05 1 E2             30 INT
06 STO T            31 10↑X
07 /                32 ST/ Y
08 FRC              33 X<> L
09 ST* Z            34 X<>Y
10 RDN              35 SIGN
11 LASTX            36 X<0?
12 INT              37 ST- L
13 1 E9             38 X<> L
14 /                39 ABS
15 +                40 1 E5
16 ↕                41 *
17 X<>Y             42 INT
18 X<Y?             43 LASTX
19 ST- L            44 FRC
20 X<Y?             45 1 E6
21 X<> L            46 *
22 R↑               47 ST+ Z
23 10↑X             48 RDN
24 *                49 .END.
25 RTN
```

(This is a nice routine Mr. Lee. However, if you find that 319 registers are not enough, we recommend the new Extended Functions/Memory Module and Extended Memory Module—Ed.)

* * * * * * * * * * * * * * *

In V5N2p12c we published a routine, DV, by **William J. Quinlan, Jr.** of Evanston, Illinois, that creates a divider bar of any length composed of any character on an HP-41 printout. Not satisfied with the methods used in the routine, **Klaus Veil** of Zurich, Switzerland, rewrote it, incorporating Swiss workmanship and the characteristics of flag 22.

(Continued)

(41 with printer) I enclose my version of the DV routine which shows off the features of the HP-41. When the calculator prompts you with SIZE↑CHAR, key in the length of bar you desire, ENTER, the number that corresponds to the special character you wish to use, then press R/S. If you just press R/S when it prompts with SIZE↑CHAR, a 24-character bar will be created in default.

```
01♦LBL "DV"          08 31
02 CF 22             09♦LBL 01
03 "SIZE↑ CHAR"      10 ACCHR
04 PROMPT            11 DSE Y
05 FC? 22            12 GTO 01
06 24                13 PRBUF
07 FC? 22            14 .END.
```

Westchester, Illinois, is the home of **Kenneth Rubin**. Mr. Rubin sent us this statistical routine.

(41) The following HP-41 routine computes permutations and combinations of N items taken R at a time. (Those of us who have done our homework will remember that R<=N<=69.) By virtue of the stack-lift-enabling property of PROMPT, I believe only two ENTER statements are needed. (Alternatively, key in N, ENTER, ENTER, R, ENTER, and omit lines 02-07.) Only the stack is used.

```
01♦LBL "PRM"         12 RDN
02♦LBL 01            13 FACT
03 "N?"              14 R↑
04 PROMPT            15 /
05 ENTER↑            16 RTN
06 "R?"              17♦LBL "CMB"
07 PROMPT            18 XEQ 01
08 ENTER↑            19 R↑
09 RDN               20 FACT
10 -                 21 /
11 FACT              22 RTN
```

* * * * * * * * * * * * * * *

When the HP-41 is running a program and it encounters a GTO or XEQ statement that calls a local label (00-99, A-J, and a-e), it searches ahead for the label, and then it records the distance to the label (in bytes) with the GTO or XEQ statement. This process is sometimes referred to as "compiling," and it speeds-up future executions of the program. (See V5N2p6c, "In the Key of HP," for more information on labels.)

Packing the program erases the recorded distances in GTO and XEQ statements. After PACKING, the first few executions of a program will generally take longer because the calculator must search for any local labels. When all of the GTO or XEQ statements have been executed once, the program is again compiled and the calculator doesn't have to search for local labels until the program is packed or lines are added.

James LeMay, who lives in Houston, Texas, recently wrote to remind us that if we PACK and then compile our programs before recording them on cards, then our programs will execute faster when they are read into the calculator, because local label searches aren't performed in compiled programs. The easiest way to compile a program is to execute it several times, making sure to consider every local branch in the program. Another way that compiling can be done is to position the calculator to each GTO or XEQ statement that calls a local label, then press R/S; let the program run until the "goose" jumps ahead 1 step in the display, then again press R/S. After all of the local branches are compiled, the program is ready to record on a card.

However, you will be able to enjoy the benefits of recording compiled programs (more rapid execution) only if you do not **PACK** or **GTO..** after reading the program into the calculator.

* * * * * * * * * * * * * * *

Ramer Streed had these comments that pertain to a routine we printed in the last issue. Mr. Streed is from North Mankata, Minnesota.

(41) Arnold Hinrichs' display routine to shorten the waiting time for calculations (V5N2p13a) seems to be more complex than necessary. After the first result is obtained, XEQ VIEW X or AVIEW and let the program continue to run. Then place a **STOP** in the program just prior to the **AVIEW** or **VIEW** statement that displays the next result. When the program is restarted, the next result is displayed immediately. NOTE: Flag 21 must be clear for this to work.

From the way this program runs, the HP-41 must have a display register that is independent of the X- and ALPHA-registers.

(Actually, Mr. Streed, the display is not a "register" in the normal sense of the word. It is a separate entity from the memory and stack registers, and it will hold a value, but there is no RAM allocated for the display. It is a function of the display driver in the central processing unit—Ed.)

* * * * * * * * * * * * * * *

Next, we have these comments from Anderstorp, Sweden. They were contributed by **Peter Josefson**.

(41) After having read KEY NOTES V5N1, I tested **Patrick Shibli's** "Scrolling-Routine" on page 12 and discovered this: Whenever a **VIEW** or **AVIEW** instruction is in operation (the "flying goose" is replaced with something else), any ignored operational error (flag 25) causes the display to scroll to the right at each following label, just like the "flying goose" usually does. This goes on until **STOP**, **END**, or **PSE** is executed.

However, the **GTO** "?" instruction in Mr. Shibli's routine causes the HP-41 to search for the NONEXISTENT LBL "?" before it continues. This delay can easily be avoided by replacing **GTO** "?" with, say, **RCL** nn, where nn is a nonexistent register. Here, the HP-41 "knows" instantaneously that the register does not exist and, therefore, the display (max 12 letters!) starts scrolling without noticeable delay.

(Another fast way to create an error (that clears flag 25) is SF nn, where nn is a nonexistent flag—Ed.)

* * * * * * * * * * * * * * *

Frank Wales, of Glaskow, Scotland, recently wrote us to correct an error that appeared in KEY NOTES, V5N2.

(41) Having come across an error in the last issue that I didn't notice on first reading, I thought I had better tell you about it. It concerns what was said about testing complex conditions (V5N2p12c). **Claude Roeltgen** states that if you have two conditionals A and B and you wish to construct the relationship "IF A OR B THEN instruction," then you can express this as: 01 inverse of A; 02 B; 03 instruction. This is correct. For example, the relationship: IF (X>Y? or FS?00) then XEQ 05, is created by:

01 X≤Y? (inverse of first condition)
02 FS?00
03 XEQ 05.

However, Mr. Roeltgen also states that the relationship "IF A AND B THEN instruction" can be constructed by: 01 A; 02 inverse of B; 03 instruction. This is incorrect. The truth table would be:

A	B	instruction performed?
0	0	1
0	1	1
1	0	0
1	1	1

while the truth table for the AND relationship should be:

A	B	instruction performed?
0	0	0
0	1	0
1	0	0
1	1	1

(Very sharp eye, Mr. Wales. The shortest routine that I could come up with for the "IF A AND B THEN instruction" relationship looks like this:

01 inverse of A
02 GTO 01
03 B
04 instruction
05 LBL 01

and I'm anxious to see a shorter version—Ed.)

* * * * * * * * * * * * * * *

Recently, we received two letters dealing with the subject of root-finding routines. The first letter was sent to us by **R. H. Miller** of Oakland, California. Mr. Miller reminded us that the *HP-41C Standard Applications Handbook* has an error in the listing of the "Root Finder" program; steps 44 and 45 should be switched to result in: 44 1E−8; 45 X>Y?. This error was corrected in the May 1981 printing of this handbook.

The second letter was from **Fredrick Öberg** of Linköping, Sweden. He enclosed

a root finder routine that he uses with his HP-41.

(41) This routine is a very tiny root-finder, using the "Regular Falsi" method. It is a useful routine that is a part of the normal status of my calculator. It will stop when an accuracy equal to the **FIX** is achieved. It requires two guesses. The function, in the form F(X)=0 is to be programmed into memory under the global label "Y". The keystrokes are: guess 1 [ENTER], guess 2 [XEQ] [ALPHA] ROOT [ALPHA]. This routine has proved very useful in math tests at school.

```
01♦LBL "ROOT"    17 RCL 03
02 STO 01        18 *
03 X<>Y          19 -
04 STO 02        20 X<>Y
05 XEQ "Y"       21 ENTER↑
06 STO 03        22 X<> 03
07 RCL 01        23 X=Y?
08♦LBL 00        24 GTO 01
09 VIEW X        25 -
10 STO 01        26 /
11 XEQ "Y"       27 RND
12 STO Y         28 GTO 00
13 RCL 02        29♦LBL 01
14 *             30 RCL 01
15 RCL 01        31 RTN
16 STO 02
```

* * * * * * * * * * * * * * *

Now, from Rochester, New York, we have this contribution by **Jefferey Smith**. This is a nice routine for obtaining a vector magnitude or the hypotenuse of a right triangle. It is a monadic function, meaning that it places X in LASTX and preserves the Y, Z, and T registers.

(41) Some of your readers who are interested in coordinate geometry might like the following routine, which will compute $\sqrt{X^2+Y^2}$. The Y, Z, and T registers are unaffected and the original X is placed in the LASTX-register. Pressing [A] while in user mode will initiate the computation. User-definable key [a] may be used to conveniently swap the X and LAST X registers. The trigonometric mode doesn't matter, but very large numbers tend to degrade the accuracy of the result.

```
01♦LBL "MAG"     04♦LBL a
02 R-P           05 X<> L
03 P-R           06 RTN
```

* * * * * * * * * * * * * * *

Here's a set of routines that was sent to us by **Gary G. Price** of Madison, Wisconsin. Those who don't have the Math Pac, but would like to have the hyperbolic functions programmed into their HP-41 will find this set of routines useful.

(41) The following are suggestions for your "routines" column in HP KEY NOTES. They are routines to compute the hyperbolic functions with an HP-41. Their features include:

1. Each makes no calls to others, so persons who routinely need one or two need not tie-up program memory with superfluous ones.
2. Each requires only the stack and leaves at least the Y-register intact. The Math Pac uses register 00.
3. Each has been honed to minimize the number of bytes required.
4. Labels that are more verbose than necessary are used here to show correspondence to the routines available in the Math Pac.

```
01♦LBL "TANH"    28♦LBL "ASINH"
02 ST+ X         29 ENTER↑
03 E↑X-1         30 X↑2
04 RCL X         31 1
05 2             32 +
06 +             33 SQRT
07 /             34 +
08 RTN           35 LN
09♦LBL "ATANH"   36 RTN
10 LN1+X         37♦LBL "COSH"
11 LASTX         38 E↑X
12 CHS           39 1/X
13 LN1+X         40 LASTX
14 -             41 +
15 2             42 2
16 /             43 /
17 RTN           44 RTN
18♦LBL "SINH"    45♦LBL "ACOSH"
19 E↑X-1         46 ENTER↑
20 RCL X         47 X↑2
21 LASTX         48 1
22 E↑X           49 -
23 /             50 SQRT
24 +             51 +
25 2             52 LN
26 /             53 RTN
27 RTN
```

And, in response to this set of hyperbolics, the technical staff of KEY NOTES combined the SINH, COSH, and TANH routines into the program below called HYP.

(41) With this program keyed into your HP-41, simply [XEQ] [ALPHA] HYP [ALPHA] to turn the [SIN], [COS], and [TAN] keys into [SINH], [COSH], and [TANH]. (The function of the keys changes but the white plastic letters on the keys stay the same.) If you don't wish to have TANH available, then save some program space by deleting lines 01-03, and 15-19. If you would like to call these functions as subroutines from other programs, then add the following global labels: LBL "TANH" above LBL J; LBL "SINH" above LBL H; and LBL "COSH" above LBL I.

```
01♦LBL J         12 +
02 ENTER↑        13 2
03 SF 00         14 /
04♦LBL H         15 FC?C 00
05 SF 01         16 RTN
06♦LBL I         17 X<>Y
07 E↑X           18 XEQ I
08 ENTER↑        19 /
09 1/X           20♦LBL "HYP"
10 FS?C 01       21 SF 27
11 CHS           22 RTN
```

The Time Module for the HP-41 makes all those timer routines that we have published in past issues of KEY NOTES obsolete (Whew!!). This is not to say that they won't have nostalgic value; sure, the children will love to see how we *used to* turn our calculators into timers.

This next routine is not a timer routine. It was sent to us by **Basil Allsop**, from Kikuyu, Kenya. You might be able to use this idea to spice-up some of your programs.

```
01♦LBL "↑"       09 R-P
02 SF 25         10♦LBL 01
03 " 0 0 0 0 0 0" 11 TONE 4
04 AVIEW         12 P-R
05 GTO "?"       13 R-P
06♦LBL 00        14 GTO 00
07 TONE 5        15 RTN
08 P-R
```

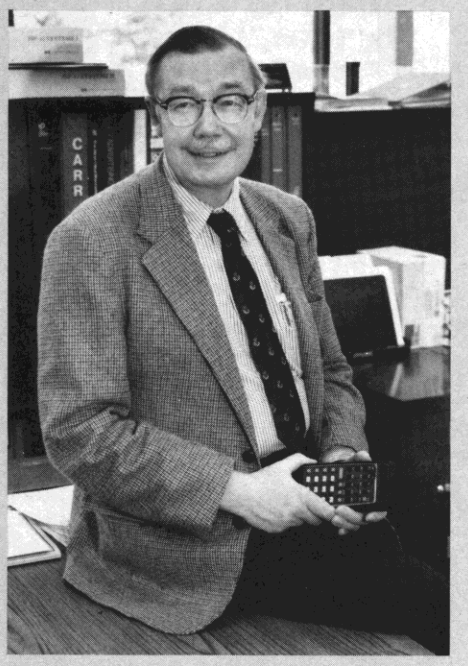
(Steps 08, 09, 12, and 13 really don't do anything but slow the execution of the program. If SF 99 is used at step 05 instead of GTO "?", the slow global label search is avoided—Ed.)

* * * * * * * * * * * * * * *

And, for you HP-65 owners who say we never give you anything, we have this routine. It was contributed by **Bob Flye** of Longview, Washington.

(65) This routine converts slope distance and percent slope to horizontal distance. Place the slope distance in the Y-register and the percent slope in the X-register.

```
LBL    23
A      11
EEX    43
2      02
f      31
R→P    01
R↓     35 08
X≥Y    35 07
f⁻¹    32
R→P    01
RTN    24
```



Where is the First HP-35?

Why is the gentleman in this photograph smiling? You would smile, too, if you had, in *your* hands, **the very first HP-35 calculator sold in January 1972**! Would we fool you about that? Read on ...

Late last year, as we were approaching the tenth anniversary of Hewlett-Packard calculators, we thought of various ways to celebrate the event. So we looked through the records and found the very first sale for an HP-35. Lo and behold! The owner is **Dr. Russell J. Donnelly**, a Professor of Physics at the University of Oregon in Eugene, Oregon, which is but a short 40 miles south of Corvallis.

So we invited Dr. Donnelly up to Corvallis and shot this photograph of him and his trusty HP-35. It is somewhat "worn" in some areas, but it works just as well as the day it was purchased. And, no it is NOT for sale—for any price. In fact, it now resides in a bank, safe and sound for posterity.

Not only is it a coincidence that the "first" HP-35 is so close to "home," but also that the owner's son works for Corvallis Division!

New Accessories Released

The following accessories are now available for the HP-41 Calculator/Computer System and the HP Interface Loop (HP-IL).

HP 82175A Black Thermal Paper. Box of 6 rolls. For use with the new HP 82162A or older HP 82143A Thermal Printer/Plotters. This is NOT recommended for the HP-97.

HP 82176A Digital Mini-Cassettes. Box of 10. For use with the new HP 82161A Digital Cassette Drive. Tapes support a standardized information format. Cassettes have increased capacity (131,072 bytes) compared to a magnetic card (224 bytes).

HP 82167A 0.5 Meter HP-IL Cable
HP 82167B 1.0 Meter HP-IL Cable
HP 82167D 5.0 Meter HP-IL Cable

These accessories should now be at your local HP Dealer. Remember: if you order from the factory, you will have to pay an additional $3.50 handling charge.

Even More About Batteries

FACTS ABOUT PACKS

The HP 82120A Rechargeable Battery Pack that became available in June of 1980 is an economical power source for those who use a card reader with an HP-41 or for any heavy user of the HP-41, with or without peripherals. The pack is more compatible with the HP-41 and card reader than are the N-cell alkaline batteries because the nickel-cadmium batteries in the pack maintain a higher voltage throughout the discharge cycle, and the card reader demands a relatively high available voltage. The N-cell alkaline batteries are an excellent power source when the card reader is not used because they offer many, many hours of steady, lower-voltage power. When the HP-41 is used alone, or with peripherals other than the card reader, it makes very efficient use of the low-voltage power supplied by the N-cell alkalines. For the average user without a card reader, the N-cell alkalines may prove to be the most economical power source.

COMMON QUESTIONS

Here are answers to five of the most common questions about batteries in the HP-41:

1. How long will my rechargeable battery pack last?

Under normal conditions, a battery pack will last from 500 to 1000 charge cycles. One full charge cycle constitutes fully charging the battery pack and using the calculator until the BAT indicator comes on. Once fully charged, a battery pack will provide the user with approximately 10 hours of continous operation on an HP-41 with no attached peripherals before requiring a recharge. With the continuous use of peripherals, the hours that the battery pack provides will decrease. Six charge cycles will approximately equal the life of one set of N-cell alkalines if you use the HP-41 continuously in a running program with no attached peripherals. Two charge cycles will approximately equal the life of one set of N-cell alkalines if you use the HP-41 continuously with the card reader. And, if you use the HP-41 continuously with the wand, then five charge cycles will approximately equal the life of one set of N-cell alkalines. These numbers are only approximations, but they are valuable for determining which power source is the most economical for you.

2. How can I get the maximum life from my rechargeable battery pack?

Fully charge the battery pack. Disconnect the calculator from the recharger and use the calculator until the BAT indicator comes on in the display, then repeat the cycle. The battery in your car will last the longest if it is never thoroughly discharged, but this isn't the case with nickel-cadmium batteries. Once you start the discharge cycle it is best to continue to use the calculator until BAT comes on before recharging. Don't let nickel-cadmium batteries go completely dead, though. This may cause the polarity of one or more of the cells to reverse, which makes the pack impossible to recharge.

3. What if I want to leave the calculator and charger continually plugged into the wall; at my desk, for instance?

Leaving the calculator and recharger plugged in is common, and it will not harm the batteries or recharger. This maintains a full charge in the batteries. But, remember, when you unplug the calculator overnight it starts into the discharge cycle and it's best to use it until BAT comes on.

4. Why don't I get the full expected operating time from my battery pack?

The operating time from your battery pack depends on several factors—ambient temperature, age (the number of charging cycles), the type of operations being performed (programming vs. simple addition vs. printing), etc. Also, nickel-cadmium batteries can develop a condition called "memory" that is a temporary loss of charging capacity. To illustrate, suppose your calculator is kept on the recharger continually, except for a 20-minute period each day when it is used on battery power alone. Over a long period of time (months), the batteries will begin to "remember" the capacity that they are expected to deliver each day—20 minutes—and they will deliver no more than that amount. If you alternate using the recharger and battery power for random periods, then no harm is done. It is only the repeated discharge cycles of the same duration that produce the "memory" effect. Full charging capability can be restored to a good battery with several full charge cycles as described in question 2.

5. What do I do if the card reader stops in the middle of reading a card because of a low battery? I can't turn off the calculator, and I am not supposed to remove the card reader while the calculator is on.

Remove the card from the card reader with a firm steady pull on either end of the card. It may take a little force but it won't harm the card or the card reader. You should now be able to turn off the calculator. Turn the calculator back on. If the BAT annunciator doesn't come on, then you can continue to use the calculator alone for some time. Don't try to read a card again until the charger is plugged-in or fresh batteries have been installed.

KEY NOTES Subscription Plan

You were informed in the last two issues that, because of skyrocketing inflation, we would soon be charging a subscription fee for HP KEY NOTES. This notice and a Subscription Order is being repeated in this *last free issue* for those who might have missed it or who are reading KEY NOTES for the first time. In February 1982, Volume 6 Number 1 will be mailed ONLY TO THOSE WHO SUBSCRIBE BEFORE THAT TIME. Below are more details.

FOR U.S. AND CANADA

In the United States and Canada the subscription fee will be $5* for one year. For that fee you will receive four issues of KEY NOTES a year.

We will accept subscriptions any time after November 1, 1981. Just fill in the Subscription Order, then mail it (or a photocopy) and a check or money order for $5* to the Users' Library in Corvallis (address on back cover).

All copies of HP KEY NOTES distributed in the U.S. and Canada will be sent by first-class mail. (By U.S., we mean any address with a U.S. Post Office ZIP code.)

If you are a member of the Corvallis Users' Library and live in the U.S. or Canada, you will receive HP KEY NOTES *free* for the first year (1982).

On January 1, 1982, all current members of the Corvallis Users' Library will be added to the HP KEY NOTES subscription mailing list for one year—free of charge. Such people do not have to send in the Subscription Order. We will automatically put you on subscription for 1982. Also, everyone who joins the Users' Library in 1982 will receive a free one-year subscription to HP KEY NOTES.

FOR EUROPE (UPLE)

If you live in Europe and receive HP KEY NOTES through the Users' Program Library Europe (UPLE) in Geneva, Switzerland, you will presently continue to receive HP KEY NOTES, and you will receive it free in 1982. The issues you receive will be printed in the U.S. and bulk-shipped by air freight to the Amsterdam mailing house. INDIVIDUAL COPIES CANNOT BE OBTAINED FROM CORVALLIS unless you are a paid-up member of the Corvallis Users' Library. If and when this plan changes, you will be notified in advance.

You do not have to send in the Subscription Order.

FOR ALL OTHER COUNTRIES

If you live in Mexico, South America, Africa, Australia, New Zealand, or Asia, or in any country not covered above, please fill in the Subscription Order on this page and mail it to the nearest Hewlett-Packard office. This will assure that you will continue to receive HP KEY NOTES in 1982. If you cannot determine where to send the form, send it to Corvallis, and we will see that it gets to the right location. Depending on where you live, you might be asked to pay a mailing fee in order to continue receiving HP KEY NOTES.

Of course, if you live anywhere outside the U.S. and are a member of the Corvallis Users' Library, you do not have to send in the Subscription Order. You will receive HP KEY NOTES in 1982 as part of your Library subscription.

FOR THE FUTURE

Effective January 1, 1982, we will include a copy of the current KEY NOTES and an invitation to subscribe to it inside each HP-41, HP-67, and HP-97 carton.

AND FINALLY . . .

We want you to know that, as a result of this subscription plan and future plans, KEY NOTES will only get better. You will get it on a regular schedule. There are still a few "bugs" to iron out of some overseas shipments, but we *are* making progress. Right now, Europe receives KEY NOTES five times faster than just a year ago. And perhaps we can improve on *that* record.

Without the economic problems that have haunted the newsletter of late, and by using better classes of mail and freight delivery, you will find that KEY NOTES will be a regular companion—there when you count on it. And with many exciting improvements in the next year, you won't want to miss getting KEY NOTES.

** U.S. dollars. Orders from anywhere outside the U.S. must include a negotiable check (or money order), in U.S. dollars, drawn on a U.S. bank. Payment must accompany your order.*

Before You Fill-In Your Order...

Do not mark the first option if you choose the third option. KEY NOTES *is included* (for 1982) in a Corvallis Library subscription. In other words, do not send $25 or $35 with your order.

Do not send cash if you can avoid it. Use a check or money order, payable to Hewlett-Packard in U.S. dollars, and drawn on a U.S. bank. Or use your American Express, MasterCard, or VISA charge account and furnish your name exactly as on the card, plus the card number and expiration date.

Do not send puchase orders or request invoices for the $5 KEY NOTES subscription. If you want your company to reimburse you, use the Subscription Order as an invoice or use it to apply for Petty Cash funds.

Do mail your order to: KEY NOTES Subscription; Hewlett-Packard Company; Corvallis, Oregon 97330.

Our thanks to all of you who have already sent your checks and orders. As of November 2 (the start of our fiscal year) we have begun entering KEY NOTES subscriptions for 1982.

HP KEY NOTES Subscription Order

Yes, I want to continue to receive HP KEY NOTES.
Please enter my order as follows:

(Check or money order payable to Hewlett-Packard)

- ☐ I enclose $5* for a 1-year subscription for the U.S. and Canada.
- ☐ I live outside the U.S., Canada, or Europe and want to be kept on the mailing list.
- ☐ I want to join the Corvallis Users' Library and enclose payment. ($20* U.S. or Canada; $30* elsewhere.)

Your name, complete address, and postal code:

Stock Plotting on the HP-41

Oak Park, Illinois, is the home of **James Grandstaff.** He has developed this routine for use on the HP-41 with the HP 82143A Thermal Printer/Plotter, and it will work with the new HP-IL compatible printer, the HP 82162A Thermal Printer/Plotter.

(41 with printer) While reading HP KEY NOTES, V5N2p12, I came across the item by **W. W. Trotti, Jr.** I wrote a similar subroutine for plotting a weekly stock chart. The subroutine prints the weekly dates, month by month, on the X-axis and the stock values on the Y-axis. It stops to prompt the user for each Y-value. The calendar function takes care of leap years and year-end roll-overs in the graph.

To use this routine:

1. In the X-register, place the date that corresponds to the first stock value you want to plot. The form of the date is MMDD.YYYY (NOT the standard form). For example, 102.1981 starts the stock graph at January 2, 1981.
2. [XEQ] [ALPHA] DATE [ALPHA]
3. [XEQ] [ALPHA] PRPLOT [ALPHA]

The calculator will proceed to ask seven questions and then start prompting for data. Respond to the questions in the following manner:

QUESTION	ANSWER
NAME?	STOCK
Y MIN?	minimum stock price
Y MAX?	maximum stock price
AXIS?	purchase price (between YMIN and YMAX)
X MIN?	day of the month for first entry (between 1-31)
X MAX?	100
X INC?	7 (days per week)

```
01♦LBL "STOCK"
02 STO 12
03 7
04 ST+ 06
05 XEQ 00
06 7
07 ST- 06
08 RCL 12
09 STOP
10 RTN
11♦LBL 00
12 RCL 13
13 RCL 06
14 X<=Y?
15 RTN
16 -
17 CHS
18 STO 06
19 1
20 ST+ 14
21 13
22 RCL 14
23 X=Y?
24 XEQ 03
25♦LBL 01
26 .1
27 STO 15
28 RDN
29 2
30 XEQ 02
31 ISG 15
32 4
33 XEQ 02
34 6
35 XEQ 02
36 9
37 XEQ 02
38 11
39 XEQ 02
40 31
41 STO 13
42 RCL 16
43 FS?C 00
44 STO 13
45 30
46 FS?C 01
47 STO 13
48 RTN
49♦LBL 02
50 X=Y?
51 SF IND 15
52 RDN
53 RTN
54♦LBL 03
55 1 E2
56 RCL 06
57 +
58 RCL 17
59 FRC
60 +
61 1 E-4
62 +
63♦LBL "DATE"
64 STO 17
65 28
66 STO 16
67 RDN
68 1 E2
69 /
70 INT
71 STO 14
72 LASTX
73 FRC
74 1 E2
75 *
76 INT
77 STO 06
78 LASTX
79 FRC
80 1 E4
81 *
82 4
83 MOD
84 X=0?
85 SF 00
86 29
87 FS?C 00
88 STO 16
89 RCL 14
90 XEQ 01
91 .END.
```

Here is a plot of HP stock from August 17, 1981 to December 14, 1981.

```
NAME ?
STOCK                    RUN
Y MIN ?
              39.00      RUN
Y MAX ?
              49.00      RUN
AXIS ?
              46.75      RUN
X MIN ?
              17.00      RUN
X MAX ?
             100.00      RUN
X INC ?
               7.00      RUN

      PLOT OF STOCK
   X <UNITS= 1> ↓
   Y <UNITS= 1> →
    39.0          49.0
                  46.8
   ├-------------┼---┤
17                  x
24             x    ¦
31             x    ¦
 7          x  ·    ¦
14             x    ¦
21       x          ¦
28  x               ¦
 5          x       ¦
12             x    ¦
19          x       ¦
26             x    ¦
 2                  ¦x
 9                 x
16          x       ¦
23  x               ¦
30       x          ¦
 7  x               ¦
14   x              ¦
```


HP KEY NOTES

January-February 1982 Vol. 6 No. 1

Programming and operating tips, answers to questions, and information about new programs and developments. Published periodically for owners of Hewlett-Packard fully programmable personal calculators. *Reader comments or contributions are welcomed. Please send them to one of the following addresses.*

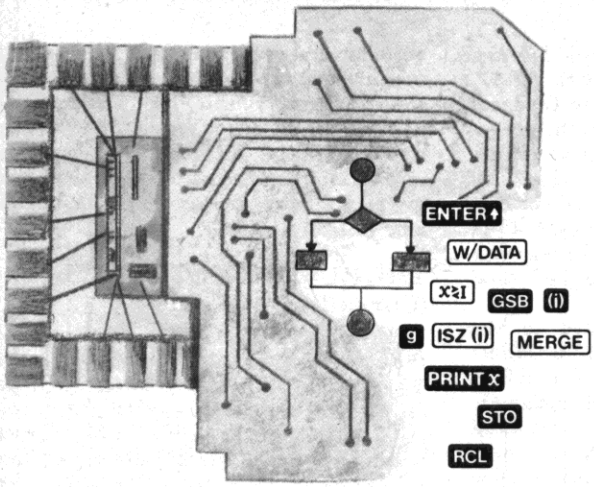
Hewlett-Packard Company
Users' Library
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 USA

Hewlett-Packard SA
Users Program Library Europe
7, Rue du Bois-du-Lan
P.O. Box, CH-1217 Meyrin 2
Geneva-Switzerland

Hewlett-Packard Company
Corvallis Division
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 U.S.A.
Address Correction Requested
Return Postage Guaranteed

FIRST-CLASS MAIL
U.S. POSTAGE
PAID
PERMIT NO. 814
PORTLAND, OR

Printed in U.S.A.



Featuring, this issue:

HEWLETT PACKARD

March-May 1982 Vol. 6 No. 2
ISSN 0730-2037

HP Key Notes

Published Quarterly | By Subscription: $5/yr. | $2.00

HP-41's Again Aboard *Columbia*

Unless you have been hiding in an igloo near the North Pole for the last year or so, you know all about the space shuttle *Columbia*, which we featured on the cover of V5N1. And, because you read KEY NOTES, you know that the astronauts use our HP-41 handheld computers onboard *Columbia* for various flight-related, radio-contact, and backup operations. And, no, their programs are not in the Users' Library nor are they for sale. They contain NASA proprietary information and are for use only on the space shuttle.

For the last *Columbia* flight, near the end of last March, the two HP-41 computers were purchased over-the-counter by NASA from a Houston, Texas, office-equipment store, and were tested rigorously before being approved for flight. They are identical to the hundreds of thousands of HP-41's sold since 1979.

One HP-41 computer, dedicated to what NASA calls the acquisition-of-signal program, was the only convenient means the shuttle crew had to estimate the time, location, and radio frequency of their next contact with Earth. Also, if the astronauts are awakened at night by an alarm, they can tell at a glance how long it will be before they can discuss the problem with Mission Control.

The second HP-41 computer acts as an electronic secretary for the astronauts—reminding them of daily chores with alarms and flashing messages. Each morning, the astronauts programmed their computer with five to ten alarms. That way they didn't have to write down on paper all their scheduled activities. In other words,

(Continued on page 16)

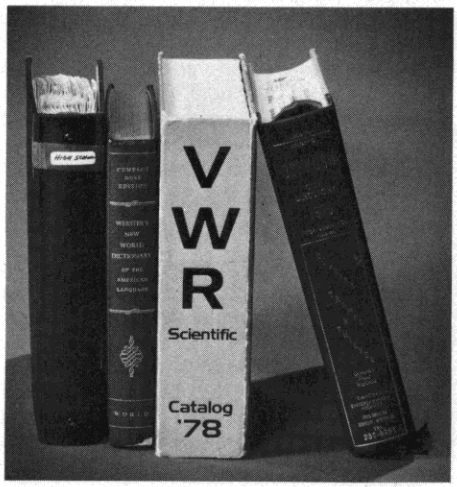
Astronaut Gordon Fullerton aboard *Columbia* on the last flight, using his HP-41. Notice that he is "sitting" in midair, in the "zero" gravity of outer space. (Photo courtesy of NASA.)

We Get Letters...

And we *know* that you are going to not only *like* this one but also *use* it for *your* purposes. There's a lot of emphasis on a lot of words in that opening sentence, but for good reasons, as you will soon learn. Take a very close look at the photograph reproduced here; we'll bet you'll *never* suspect what it represents. Anyway, before you peek inside KEY NOTES, here's the letter we received from **Mrs. Keith Olson** of Cupertino, California.

"Some time ago, my husband bought an HP-41. Soon thereafter he wanted a safe,

(Continued on page 15)



All prices in this newsletter are suggested retail prices excluding applicable state and local taxes—Continental U.S.A., Alaska, and Hawaii.

Corvallis Library Corner

There are now over 4793 HP-67/97 programs and 1730 HP-41 programs in the Corvallis Users' Library, and we get more and more each and every day of the week. And, as much as we are grateful for your excellent output, there are a few subjects we'd like to see covered. For example, **Andrew N.C. Cruickshank**, a Town Planner and Economist who lives in London, England, recently wrote and asked if we had any programs in *his* field—programs such as population projection, property development, regional economic analysis, etc. And we are often surprised that there aren't more programs for farmers.

But one must admit that, all in all, you are a very prolific and excellent society of calculator programmers, and we congratulate each and every author who has taken the time to share a program through the Library.

ORDERING PROGRAMS

HP-67/97 and HP-41 programs featured in KEY NOTES are available from both the Library in Corvallis and the Library in Geneva (except where stated otherwise). **Readers in Europe should order from Geneva (address on back cover) to get quicker service.** Readers elsewhere should order from Corvallis. Each program includes documentation and prerecorded magnetic cards; HP-41 programs include bar code.

Mail your order and a check or money order to the Corvallis or GENEVA address on the back cover of KEY NOTES. Don't forget to include your State or local taxes. Or, in the U.S., you can place your order by calling toll-free: 800-547-3400, except from Alaska and Hawaii. (In Oregon call 503-758-1010, NOT TOLL FREE.)

When ordering from outside the U.S., **attach your payment to your order.** Much time is wasted and orders are held up trying to match checks and orders that are sent in separately. Your payment can be in the form of an International Money order, a Foreign Draft, or the equivalent. *Any payment must be in U.S. dollars, drawn on a U.S. bank, otherwise it will be returned to you.* Another option for payment is to use such major credit cards as American Express, VISA, or MasterCard.

Orders are usually shipped within 2 working days after they are received in Corvallis. However, if you need a program yesterday, call us today at 503-757-2000, extension 3371. Although we can't get it to you yesterday, if you call before 12:00 noon, we'll get it in the mail today.

SUBMITTING PROGRAMS (Corvallis)

Programs submitted to the Corvallis Users' Library should be on Hewlett-Packard standard Library submittal forms, or they should include at least the documentation required by those forms. To maintain the high quality of the programs submitted to the Users' Library, we encourage you to closely follow the *Users' Library Contributor's Guide for the HP-41, HP-67, and HP-97.* Complete and orderly documentation is essential to ensure the acceptance of a program into the Library.

We also encourage you to read the ongoing KEY NOTES column, "In the Key of HP." This column addresses some of the things we look for when we are reviewing programs that are submitted to the Users' Library.

Programs that are submitted to the Corvallis Library for the HP-67 or HP-97 must include magnetic cards, and HP-41 programs must include either magnetic cards, **reproducible** bar code, or a data mini-cassette for use with the new HP-82161A Digital Cassette Drive. (The cassette will be returned to you.) It would take far too long to check and review all the many program submittals if we had to key them in line by line. Also, there is always an increased chance of error when someone keys in handwritten keystrokes.

The management of the Corvallis Users' Library reserves the right to reject programs which, in its opinion, do not represent a significant contribution, are not clearly or sufficiently documented, or are not otherwise appropriate for the Corvallis Library.

THE CORVALLIS CONTEST

The 1982 Corvallis Users' Library Submittal Contest is well under way. The ten winners for March have been chosen and these winners are featured elsewhere in this column. All of our authors are to be congratulated for the fine contributions they have made to the Corvallis Library.

There are 50 more prizes to be awarded in this contest! Last year, 50% of the total contest entries arrived during the last month of the contest, so send your programs in early to increase your chances of winning. This contest runs through August, 1982.

THE POINT PROGRAM (Corvallis Only)

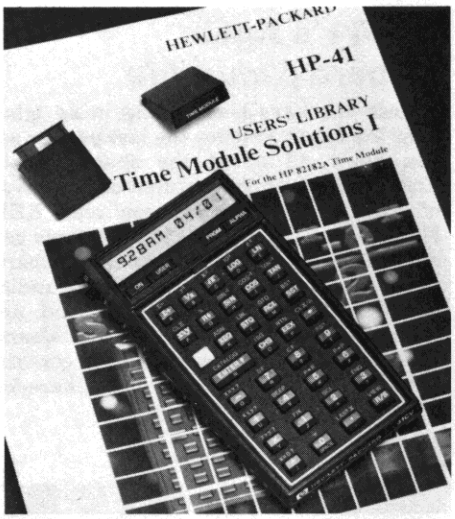
The Corvallis point program has been enthusiastically received by everyone. For those who missed the announcement, the authors of all HP-67/97/41 programs, currently accepted for the Corvallis Library, are being issued point certificates, (one per accepted program). These points are then redeemable, by the Corvallis Library, for a vast array of HP products. Depending on the amount of points you have accummulated, you can choose from products that range from a Solutions Book or a custom keyboard (presently valued at one point) to an HP-85 Personal Computer (presently valued at 100 points)! A complete list of the available merchandise can be obtained from the Corvallis Users' Library.

Now, in addition to the opportunity to win prizes in the contest, many diligent authors are well on their way toward earning enough Corvallis points to "purchase" their favorite HP products. The point program is not retroactive. **We cannot exchange Library coupons for points, nor can we redeem "Users' Program Library Europe" point certificates.**

THE CUSTOM KEYBOARD (Corvallis Only)

Foremost on the list of requested items is the custom keyboard. Everyone wants one! **The keyboard (with the standard key configuration) is presently obtainable only from the Corvallis Library** for one point, or in lieu of the 25% discount offered on orders of 6 or more Corvallis Users' Library programs. In other words, if you order 6 or more programs and *do not* take the 25% discount, you can request a custom keyboard in lieu of the discount.

MINI-SOLUTIONS



Our new Solutions Book, *Time Module Solutions I 00041-90395,* is in stock and is being shipped. It greatly compliments the HP-82182A Time Module, and demonstrates just how versatile and indispensible your HP-41 is. Time Module Solutions I and all solutions books are now available recorded. Magnetic cards are $20* per book; mini-data cassettes are $12* for the first Solutions Book requested and $6* for each subsequent book recorded on the same cassette.

MINI-SERVICES

A mini-cassette duplicating service is a new Corvallis Users' Library offering. Send us a cassette with your favorite collection of programs, and we will copy it for you—in any quantity. The cost for this service is $12* per cassette requested (that $12* includes the price of the cassette). This new service provides an ideal opportunity for businesses, clubs, and organizations to distribute their "custom" software collections to members.

MANY REQUESTS

In response to many customer requests, the Library, long ago, established standards by which programs are accepted for the Corvallis Users' Library. Our review staff checks each program for complete and accurate documentation, friendliness, and ease of use. All programs submitted must be accompanied by magnetic cards, bar code (HP-41), or mini-data cassette (HP-41), and should solve a problem sophisticated enough to warrant its purchase. These guidelines will ensure top-quality software for you.

Our review staff is available to answer any questions regarding either existing Library programs or programs that you are developing to submit to the Users' Library. Call 503-757-2000 and ask for extension 2886 between 9 am and 4 pm.

ORDER TURNAROUND TIME

The Library has finally reached its goal of giving you 48-hour turnaround on mail and TOLL-FREE phone orders, and we are proud to tell you that we intend to maintain that goal.

Orders telephoned directly to the Library (503-757-2000 X3371) always ship that same day, regardless of the number we may receive!

The Corvallis Users' Library is your Library. Write us with recommendations for services or with any suggestions on how we might serve you better.

"Old" Contest Winners

Because of the overwhelming number of programs submitted during December for the 1981 Users' Library Contest, the announcement of that month's winners and the Grand Prize winners did not make the press date for the last issue of KEY NOTES. We are featuring these winners in this issue.

The December winners all chose HP-41CV's as their prizes and, as you know, each of the Grand Prize winners is now the proud owner of an HP-85 Personal Computer!

DECEMBER WINNERS

(41) Sunpath Diagrams #01524C (Price: $10*)

SUNPATH calculates solar altitudes and azimuths for each daylight hour of the day, given date and site latitude. A blank chart is furnished for plotting sunpaths for each month. Illustrated procedures are given for using the results. Another program, labeled SUNTIME, converts from solar time to local standard time and vice versa. *Required Accessories: 2 Memory Modules for either SUNPATH or SUNTIME, 3 Memory Modules for both simultaneously.* (596 lines, 1396 bytes, 23 pages)

Author: **Ross McCluney**
Cape Canaveral, Florida

(41) General Network Reduction Program #01526C (Price: $12*)

This program analyzes networks of up to 80 elements. The allowed network elements are resistors, capacitors, inductors, reactors, and rigid voltage and current sources, in any serial or parallel combination. Output functions are voltages, currents, or impedances in any branch of the network. If the printer is available, either amplitude or phase transfer functions can be plotted. *Required Accessories: 4 Memory Modules for 80 network elements, Printer if plots are desired.* (443 lines, 905 bytes, 32 pages)

Author: **Dieter Lange**
Hamburg, Germany

And, the third winner for December was the game program **Flipo #01477C (Price: $12*)** that we featured in V6N1p5a. The author of Flipo is **Robert Swanson** of Portland, Oregon.

(Hearty congratulations are in order for these three authors! The time and effort that went into writing and documenting these programs is astounding. Thanks, to all three of you; we know you will enjoy using your HP-41CV's—Ed.)

Winning Programs

We announced the 1982 Users' Library Submittal Contest in the last issue of HP KEY NOTES. This contest began in March and it will run through August. Each month, ten winning programs will be chosen on merit by our review panel. And, the authors of these winning programs get to choose a fabulous HP product as their prize. The top two winners every month may choose a prize of either an HP-IL Digital Cassette Drive or an HP-IL Thermal Printer/Plotter (both include HP-IL Modules). The other eight winners may choose either a Time Module or an Extended Functions Module.

Here are the ten winners for the month of March. The authors of the first two programs chose a Printer and a Cassette Drive as their prizes.

(41) Sun Shade #01692C (Price: $12*)

Architects, engineers, and designers of solar-related equipment and structures will find this program useful. Its primary purpose is to compute the shadow cast by a shading device. The shading device may be horizontal, vertical, or oblique. Sun altitude, sun azimuth, and effective sun latitude are computed for daylight hours at any location in the world. Many other solar geophysical parameters are available such as declination, times of sunrise and sunset, Equation of Time, etc. Time may be specified as either apparent solar time or local standard time. A correction to standard time may be made for daylight saving time

Two powerful new design tools are introduced. A dimensionless S/L ratio that lets you calculate shading by a simple multiplication or division, and SHADE LINE, a sunrise to sunset history of a shadow and/or the S/L ratio. *Required accessories: Printer, 3 Memory Modules. Card Reader or Wand recommended.* (624 lines, 1317 bytes, 46 pages)

Author: **Bill Kraengel, Jr.**
Valley Stream, New York

(41) Fire Danger #01586C (Price $12*)

This program computes fire danger for both grasslands and forests; outputs include a numerical index, a hazard rating (LOW, MODERATE, HIGH, VERY HIGH, and EXTREME), rate of spread on level and sloping ground, and, in the case of forest fires, flame height, the hazard of crown fire, and spotting distance. A subprogram computes a drought index required for quantifying forest fires, and another subprogram computes relative humidity from basic meteorological data. Fire danger can be forecast and projected forward from conditions in the morning or on the day(s) before.

The complete program has three parts; the main program—Fire Danger—and two subprograms—Drought Index, and Relative Humidity. The main program can be used separately, or with either, or both subprograms. A flow diagram for the complete Fire Danger program is included. *Required accessories: 4 Memory Modules, and Printer is advantageous but not essential. (863 lines, 2054 bytes, 45 pages)*

Author: **Dr. Wilfred J.B. Crane**
Canberra, Australia

The authors of the following eight programs chose a prize of either a Time Module or an Extended Functions/Memory Module. They are listed in no particular order.

(41) The Ultimate Calendar—A.D. and B.C. #01593C (Price: $8*)

Complete calendar from January 1, 45 B.C. to February 28, 4904, the entire period of the Julio-Gregorian calendar that can be calculated with certainty. Computes days between dates, day of week, Julian day, date of a specified number of days before or after a given date, converts Julian day to calendar date. Rejects invalid dates. Program allows for Roman errors in inserting leap years between 45 B.C. and 8 B.C. as well as the Augustan and Gregorian corrections. *Required accessories: 3 Memory Modules.* (815 lines, 1554 bytes, 17 pages)

Author: **William Hutchins**
Los Angeles, California

(41) 1,2, or 3 Way ANOVA's #01579C (Price $8*)

One-, two-, or three-way ANOVA's (ANalysis Of VAriance), without or with replication (equal or unequal), are calculated by the unweighted means method modified to employ exact total sum of squares. This is a friendly program! Just set factors, levels per factor, whether or not you desire a

(Continued)

printer output (if the printer is attached), whether or not you desire replication (yes or no), and enter the values. The ANOVA table will come out either printed or flashed sequentially. *Required accessories: At least 3 Memory Modules.* (600 lines, 1194 bytes, 17 pages)
Author: **Dr. Nicholas Sinclair**
London, Ontario, Canada

(67/97) Trigg's Trend Analysis #04775D (Price: $6*)

Levey-Jennings control charts are commonly used in laboratories to monitor analytical variation. Unfortunately, they do not permit the simple detection of non-random trends with much sensitivity. This program provides Trigg's technique for the quantitative detection of trends in quality control data. Data may be stored on cards for ongoing evaluation. (105 lines, 7 pages)
Author: **Mike McDonald**
Berkeley, California

(41) Auto Banner #01691C (Price: $6*)

This program, using commands from the Extended Functions Module, will translate an ALPHA string and automatically print a Banner output. This program is incredibly fast. The user simply keys in the ALPHA string, the HP-41 does the rest. In addition to the keyboard characters, 106 of the 127 special characters are also supported by keying in the character number and executing the function XTOA to add the character to the ALPHA string. *Required accessories: One Memory Module, Extended Functions Module, Printer.* (103 lines, 205 bytes plus 92 data registers, 8 pages)
Author: **Christopher Erickson**
Pullman, Washington

(41) Equilibrium Flash #01569C (Price: $8*)

Given the number of moles and the K values (ratio of the fraction of the component in the vapor phase to the fraction in the liquid phase) of up to 10 compounds, this program will compute the equilibrium phase compositions. Ideal K's are calculated if the critical properties and boiling points of the components are furnished. A rugged algorithm is used that always comes to a solution. *Required accessories: 3 Memory Modules.* (487 lines, 1081 bytes, 11 pages)
Author: **Norman Samish**
Houston, Texas

(41) Vented Loudspeaker Box Tunings #01451C (Price: $12*)

Using data on the loudspeaker in question, this program solves for the "optimum" vented enclosure and permits the user to vary the tuning parameters to test alternate tunings. A 1/3-octave response listing is provided and, with the accessory printer, the frequency response is plotted. *Required accessories: 4 Memory Modules.* (660 lines, 1643 bytes, 29 pages)
Author: **Thomas Bouliane**
Buffalo, New York

(41) Symbolic Logic—Summary and Applications #01694C (Price: $12*)

This program is a relatively complete treatment of elementary symbolic logic. Logical operators defined include AND, OR, NOT, IMPLICATION (if and only if), and EXCLUSIVE OR. The operators are based on the definitions of Lukasiewicz (WookashAYveech; the "father" of RPN) and thus hold for one kind of three-valued logic. These same definitions will work for Boolean Logic when the base is 2. Thus the program may be used to simulate digital logic circuits. The user must write the programs to simulate these circuits, but examples are given to show how to do this expeditiously. A "cookbook" of compound conditionals is included in the documentation. *Required accessories: 1 Memory Module.* (186 lines, 398 bytes, 31 pages).
Author: **Edward Keefe**
Ankeny, Iowa

(41) Thermodynamic Properties of Saturated and Superheated Steam #01693C (Price $12*)

Calculate the thermodynamic properties: specific volume, enthalpy, and entropy, of saturated (liquid and vapor) and superheated steam given temperature and pressure. One equation of state, Martin's, is used over the entire range of temperature and pressure, down to $Vr = 0.56$. The calculated properties are within the tolerances given by the International Skeleton Tables (Steam). *Required accessories: 4 Memory Modules; Printer helpful.* (1047 lines, 1765 bytes, 36 pages)
Author: **Robert Wooley**
Midland, Missouri

GRAND PRIZE WINNERS

And now (drum roll), this is the announcement that you have been waiting for. Here are the **Grand Prize winners** of the 1981 Users' Library Contest. **These three winners each received a fabulous HP-85 Personal Computer as their prize!**

(41) Superbeam #01044C (Price: $10*) Author: Steven F. Dusterwald

(41) Acid-Base Factors for Blood and Brain Interstitial Fluid #01030C (Price: $10*) Authors: Thomas Adams and S. Richard Heisey

(41) General Network Reduction Program #01526C (Price: $12*) Author: Dieter Lange

The abstracts for the first two programs were featured in V5N3p4,5. They were winners for the month of September. And, the third program is listed as a December winner in this issue.

We are very happy to express our thanks to these three **(whoops, four)** authors for their contributions to the Users' Library. Through the efforts of these authors and thousands of others, the Library will continue to expand its services to the hundreds of thousands of HP calculator users.

Best Sellers

We have had a lot of queries lately about the popularity of certain programs, and this is usually followed with the question: "What are the most popular programs in the Library?" So we compiled a list of the best-selling programs since January 1982, and here they are. They are presented in numerical order, which is not necessarily their order of sales. Notice that most of them have already appeared in KEY NOTES. We congratulate these authors and encourage them to continue their level of excellence in the future.

HP-67/97 PROGRAMS

- Star Trek Advanced #00369D (Price: $6*) Author: **L.G. Schneider**
- English Metric Conversions #00434D (Price: $6*) Author: **E.R. Kool**
- Stock Selection: Criteria of Ben Graham and James Rea #01544D (Price: $6*) Author: **K.L. Hellams**
- Feeder Sizing and Voltage Drop #01878D (Price: $6*) Author: **N.J. Peros**
- Tape Recorder Counter Conversions #01917D (Price: $6*) Author: **P.M. Gehlar**
- Perspective Plot #02849D (Price: $6*) Author: **L.H. Anderson**
- Oil or Gas Downhole Pressure Build-up Analysis #03246D (Price: $6*) Author: **D.G. Olson**
- Hex/Decimal, Hex Arithmetic for Microprocessors #04350D (Price: $6*) Author: **D.T. Brown**
- Astrophotography Exposure Guide #04551D (Price: $6*) Author: **J.P. Patterson**
- Curve Fits #04719D (Price: $12*) Author: **E.A. Taylor**

HP-41 PROGRAMS

- Simplex Algorithm #00320C (Price: $6*) Author: **L.A. Esterhuizen**
- Wizard of Pinball #00361C (Price: $6*) Author: **C.A. Pearce**
- Plot of 2/3 Functions on one Graph #00732C (Price: $6*) Author: **J.L. Gilby**
- Hunt the Wumpus II #00783C (Price: $6*) Author: **B.J. Wheeler**
- Football Super III #00803C (Price: $6*) Author: **J.P. Dublirer**
- The Caves #00900C (Price: $6*) Author: **J. Surber**
- Microcalc #01115C (Price: $6*) Author: **N.C. Shammas**
- Advanced Star Trek #01321C (Price: $12*) Author: **J.P. Patterson**
- Rubik Cube Solution #01342C (Price: $12*) Author: **J.L. Gilby**
- Phone Directory #01459C (Price: $6*) Author: **J.F. Glass**

** U.S. dollars. Orders from anywhere outside the U.S. must include a negotiable check (or money order), in U.S. dollars, drawn on a U.S. bank. All orders from anywhere outside the U.S. and Canada must include an additional 10 percent fee for special handling and air mail postage. (For example, an order for two programs = $6 × 2 = $12 + $1.20 = $13.20 total.) If you live in Europe, you should order KEY NOTES Programs directly from the Geneva UPLE, but make certain you make payment as required by Users' Program Library Europe; the above $6 fee is good only for orders to the Corvallis Library.*

If *you* would like to see what makes a program popular and would like to order any of the above programs, they are available from the Corvallis Library for the listed prices. However, if you live outside the U.S., don't forget to add an additional 10 percent handling and postage fee for the order, and make sure that payment in U.S. dollars accompanies your order.

Book Reviews

Books are reviewed or announced in KEY NOTES only as a service to our readers. A review here does **not** represent an endorsement by Hewlett-Packard. If you are unsure about the contents of a book, we suggest you first check with a local bookstore; if that fails, write to the publisher. Availability problems also should be addressed to the publisher, not to KEY NOTES.

COCKPIT COMPUTERS, by **Paul Garrison**, a new book just released by the McGraw-Hill Book Company, is a 249-page hardbound book (with dust cover) in 7.3 by 9.1 inch format (18.5 by 23 cm). A main selection of the Flying Book Club, it is replete with practical tips for the conscientious aviator, and it shows how to use today's aviation calculators and computers effectively, thus making any flight more efficient and saving considerable fuel in the process. Although there are many aviation computers and calculators described in *Cockpit Computers*, readers of KEY NOTES will be interested in the fact that the HP-41 is included. However, most of the actual programs are for other types of calculators. We recommend that you look before you buy.

On the plus side, the author *thoroughly* discusses such specific aviation problems as the time, fuel, and distance to climb; true airspeed; estimate time en route; area navigation; range and endurance; weight and balance; and so on.

Paul Garrison, the author of this new book, is not a new writer. He is not only a professional pilot but also a well-known freelance writer who has written over 300 articles for such national flight magazines as *Air Progress Aero*, *Private Pilot*, and *Plane and Pilot*. He is the author of 20 prior books, including *Inside Private Aviation*, *Cross Country Flying*, and *Gliders—How to Build and Fly Them*.

Although this book has just been released, it does not include the new Hewlett-Packard Interface Loop or its peripherals, or even the new Time Module. So if you combine these new products with the information in the book, the book takes on a whole new flavor.

Because it is a new book, you might not see it in your local bookstore for a month or more, especially if you live in faraway places. The price of the book is $24.95 (U.S. dollars) and the publisher is McGraw-Hill Book Company of New York. They also have offices in St. Louis and San Francisco, and in Aukland, Bogata, Hamburg, Johannesburg, London, Madrid, Mexico City, Montreal, New Delhi, Panama, Paris, Sao Paulo, Singapore, Sydney, Tokyo, and Toronto. Check at your local dealer or call the publisher's agent to locate or order this book. As a last resort, write to the publisher:

McGraw-Hill Book Company
1221 Avenue of the Americas
New York, NY 10020 U.S.A.

FEEDBACK, by **Fred D. Waldhauer**, is another new book just off the press. It is hardbound (with dust cover), 651 pages, and in 6.3 by 9.1 inch format (16 by 23 cm). And because the programs in this book deal specifically with the HP-41, we know a lot of our readers will want to see it.

The author, Fred D. Waldhauer, is a supervisor in the Transmission Technology Laboratory at Bell Laboratories in Holmdel, New Jersey. His work has concentrated on feedback processes and digital communications. He is the author of papers on circuits, feedback, and high-speed digital transmission, and holds 14 patents in these fields. Mr. Waldhauer is also a Fellow of the IEEE (1977), a professional engineer (New Jersey), a member of the Audio Engineering Society, and he received his M.S. in electrical engineering from Columbia University.

This book describes the first new, original approach to feedback in over 50 years, with important applications for electronic circuit design. Mr. Waldhauer greatly simplifies feedback analysis and design by adopting a new pattern based on "anti-causal analysis"—the analysis of feedback from output to input. This approach makes the feedback analysis problem easy to trace from the initial rough approximation to the final exact analysis and design. It offers a neat solution to what has always been a difficult problem.

In the book there are many examples and calculator programs that enable the reader to apply its principles to all problems involving feedback structures. Most of the examples are derived from electrical circuits. These examples range from audio-frequency design to microwave integrated circuits. The Table Of Contents is:

1. Feedback Amplifiers: An Alternate Foundation
2. Polynomials of Loss: Various Descriptions of Polynomials
3. Elements of Feedback Synthesis: A Case Study
4. Signal Flow Graphs of Polynomials, Rational Functions, and Circuits
5. Signal Delay in Feedback Systems
6. Two-Port Analysis of Circuits and Devices
7. Feedback Analysis of the Bipolar Transistor
8. Two-Port Feedback Analysis
9. Analog Integrated Circuit Design: Feedback and Feedforward
10. Output-Stage Design
11. Noise and Input Stages
12. Differential and Operational Amplifiers

Appendix A. Programs For Manipulating Polynomials
Appendix B. Feedback Analysis and Synthesis Programs
Appendix C. Two-Ports, Transistors, and ABCD Matrices

For readers who want to apply the new methods directly to their own designs or to check the designs given in the book, 31 programs are provided, all written for the HP-41C or HP-41CV. These programs cover most aspects of the material in the book. One of these programs synthesizes feedback systems for a prescribed performance; another converts the HP-41 into a "two-port network calculator." Included are the four basic functions of addition, subtraction, multiplication, and the matrix inverse, as well as lead interchange operations. This "calculator within a calculator" is itself programmable, and can convert numerical results into network properties including loss, input and output impedances, and sensitivities as functions of frequency.

An outgrowth of courses taught by the author in Bell Laboratories, this book will serve as a text in courses and professional seminars. Electronics engineers, scientists, technicians, and upper-division electrical engineering students should find it a more direct approach to the design of feedback systems and circuits.

The price is $47.50 (U.S. dollars) and, before you rule it out, remember that it contains 31 programs for the HP-41, replete with example, listing, etc. Check your local bookstore or agent first, and remember that this is a new book that might take months to get to faraway places. As a last resort, in Europe and Asia contact:

John Wiley & Sons, Ltd.
Baffins Lane, Chichester,
Sussex PO 19 1UD
England

In Australia and nearby areas, contact:
Jacaranda-Wiley, Ltd.
GPO Box 859
Brisbane, Queensland 4001
Australia

In the U.S., Canada, and Mexico, contact:
John Wiley & Sons, Inc.
Wiley Interscience Division
605 Third Avenue
New York, NY 10158

Something For Nothing

In the V5N3 issue of KEY NOTES, **John Loux** discussed inoperative functions in the "In The Key of HP" column under the subtitle "Indices." This discussion brought in a lot of mail dealing with "no-ops," and indices in general, and we thought you would enjoy seeing some of the suggestions that we received.

First, **Robert Whipple** of Washington, D.C., has this to say.

My candidate for filling the inoperative line following an ISG or DSE is a single, solitary decimal point. As I understand it, this takes only one byte (the same as CLD) and it has the documentary advantage that it is visually as close to a completely blank line as you can get

(Continued)

in a program listing. (This advantage is less pronounced, of course, in a program listing done in TRACE mode.) Instead of initializing a nontested index with a value like 1.999 and incrementing it with a simple ISG, I initialize it with the value 1 and follow the ISG with a single decimal point, thus saving 3 bytes of memory.

And, from **John Allen** in Nashua, New Hampshire, we have this.

KEY NOTES is great and I appreciate your suggestions for a "NOP" following DSE or ISG instruction. In the program I was writing, I couldn't use CLD because I wanted to preserve an AVIEW report. And X<>X, as you point out, takes one more byte, and also takes too long to write (I'm lazy). So why not DEG? It works fine and only takes one byte! So do RAD or GRAD, if you're in one of those modes.

Now, **Miles Abernathy** of Austin, Texas, has this to say.

A handy "do nothing" step is LBL nn. If nn is 00 to 15, it's only one byte, and it doesn't change any register.

Thanks, to you three, and to everyone who sent us setters dealing with this subject.

In the Key of HP

Most of the information you find in KEY NOTES is contributed by HP calculator users, just like yourself, who live in all parts of the world. This ongoing column is, for the most part, written by **John Loux** who is a Technical Advisor in the Corvallis Users' Library. His articles contain information, tips, and techniques that will help you to write more useful and efficient programs. Here's his latest.

THE DIGITAL CASSETTE DRIVE

The HP-41 has been recently endowed with the ability to access and manipulate large amounts of data and program information. I am speaking, of course, of the introduction of the HP 82161A Digital Cassette Drive. But, as with all technological innovations, the advantages gained are coupled with increased responsibility. The Users' Library is most concerned with the programmatic manipulation of the cassette drive. We know that program authors need to be made aware of the potential advantages and disadvantages inherent to the device before they can effectively generate application programs that use it. Hence, this article.

MANUAL MANIPULATION OF CASSETTE FUNCTIONS

To a computer user, the computer's access to mass storage is seldom a concern and is often transparent. This is as it should be. The user should not be burdened with how the currently used application program stores and retrieves data or chains program sections. It is for just this reason that all functions of the cassette drive (except NEWM) are programmable. The combination of the necessary functions being programmatically accessible with the powerful decision-making ability of the HP-41 makes it possible for a well-written program to handle virtually all aspects of cassette drive manipulation.

The program author should keep in mind that one of the main reasons that programmable calculators exist is for user convenience. The Library therefore feels that it is generally unacceptable for a submitted program to require manual intervention to deal with programmatically manipulable features of a device.

DATA SAVING

Because the HP-41 has a limit to the amount of read/write memory that it supports, programs that deal with (comparatively) large volumes of data often find themselves critically limited in the amount of information that can be stored. Even if room can be found, many times data must be destroyed by manipulations in the program. In the latter case, further calculation on the same data must be preceded by re-storing the data. For this reason, perhaps the greatest advantage that the cassette drive affords is the ability to retain this data for further and future uses. Cassette data files make it possible to access data without loading all of it into calculator memory, thereby reducing the need for a large number of calculator data registers. The cassette data file also can be easily reused without being modified.

Programs that generate and/or manipulate data files should take advantage of the cassette drive's ability to preserve initial data and not force the destruction of any accumulated data. In submitted programs, the purging or zeroing (clearing) of any data file should be well justified and well documented.

SINGLE VS. MULTIPLE DATA FILES

What appears to be a convenient way to segregate groups of data turns out to yield the slowest rate of data access. Multiple data files are an excellent way to segregate unrelated information but one must weigh the benefits of segregation against the speed and efficiency of access.

Movement through a single data file is relatively fast. Multiple blocks of information can be stored in the same data file if "pointers" to the beginning register of each block are stored in calculator memory. These pointers can be recalled and used with the SEEKR and READRX functions to access the desired block. Also, short routines can be devised to calculate the necessary pointer value. In contrast, addressing a new data file each time a different block of data is needed requires the same seek function, and, in most cases, the added delay of searching the cassette directory for the file location and the delay of moving to the file.

There is no set rule to follow in judging between the two methods of data block access. Both have their merits. All that the Library asks is that the authors of programs that use large or multiple data files consider their options.

DOCUMENTATION

A point that again must be stressed is that a main reason that calculators and software exist is for user convenience. Program authors should realize that much of a useful program's friendliness is lost if it is not documented well enough for the user to learn how to use it in a reasonable amount of time.

Documentation of the content and use of data files can be critical. The most obvious time that documentation of data file contents becomes a concern is when a file of constant values is destroyed. The user must be supplied with enough information to be able to resurrect the file. Another hopefully more common situation is where the user desires to write his/her own routines that access the author's data files. To be able to do this, it is obviously important that the user understands the internal organization of the file.

The Library expects that the documentation that accompanies each program dealing with data files completely describes the contents of any constant file and completely describes the internal organization of any variable file. The reasons for and methods in which the program manipulates the file(s) also should be well described.

ERRONEOUS ASSUMPTIONS

Most critical problems with programs stem from assumptions that the author has made about the configuration and/or state of the calculator. Some invalid assumptions that may arise in dealings with data files on the cassette drive are:

1. **The interface is in MANIO or AUTOIO mode.** The assumption of one or the other mode is invalid unless the program itself has set either mode. This is not to say that the program must set one or the other mode. Many application programs may run equally well in either mode, so long as the selected device in MANIO is of the appropriate type. If either mode is required, the program should set it. In certain instances, a program may work differently (intentionally) in either mode. In this type of program, calculator flag 32 should be tested in order to determine the current mode.
2. **The file pointer is currently pointing to the desired file.** This assumption is also invalid unless the program itself has performed the required SEEKR function.
3. **The file to be output to exists or doesn't exist.** It is dangerous to assume that if a file exists whose name is the same as that which the program uses that it is necessarily the file the user wants to be accessed. If the named file exists, the user should be warned either by the program or by the documentation that the program may alter its contents. Unconditional creation of

data files can also cause problems if another file exists that has the same name or if the program destroys any file of the same name before creation. Creation, destruction, and modification of file information must be well documented. In each of these cases, the program could query the user before the action is taken, thus making certain that it is desirable.

PROGRAM FILES

The reader's first thought about the usefulness of the cassette drive in conjunction with program storage is probably along the lines of structuring a personal program library. Although this is a reasonable pursuit, it may not be the most important program application of the cassette drive.

One way to optimize the use of calculator memory is to split the program into functional segments that can be loaded as necessary from mass storage into calculator memory. This technique may require more logical planning in order to maintain program integrity and consistency, but the advantages gained by calculators with a limit to their memory are in most cases well worth the effort. The program's sections may "chain" each other into memory, each calling the next as its usefulness is exhausted. Alternatively, one master program may call several slave routines that replace each other in memory.

The Library recommends that program files and subprogram files be documented in the same complete fashion as that expected of all submitted programs. Subprograms require more documentation than "stand alone" programs in order to aid the user who desires to write his/her own routines that access the subprograms.

The cassette drive supports status, write-all, and key-assignment files. Because the Library is a *user-oriented* organization, we recommend that you follow certain guidelines whenever you consider submitting one of these types of files. First of all, many users feel inconvenienced by programs (or documentation) that *require* key-assignments or that *require* a certain calculator status. For this reason, the Library recommends that submitted programs do not call key-assignment or status files without first giving the user a choice. Defining the keyboard or status of the calculator is best done in optional routines within the program, and these definitions should be well described in the documentation. Secondly, write-all files cause difficulties both for the Library and for program users. Plus, programs recorded as PRIVATE present the Library with obvious reproduction problems. For these reasons, the Library discourages the submittal of write-all files and PRIVATE programs.

DEVICE DEPENDENCY

It is not always obvious when a program requires a peripheral. For example, a program may use peripheral functions in such a way that if the peripheral is not in the system the functions are not encountered. Therefore, a program does not require a peripheral simply because it uses functions found only in that device. Neither can a program claim to require a mass storage device simply because the Library requires some sort of mass storage media with each submittal. A program requires a peripheral only when it cannot perform its function without the device.

In order to make programs accessible to the largest possible number of users authors should not write programs that require peripherals unless they feel that the requirement is a definite enhancement to the program.

New Special Service Offered

We now offer at Corvallis, HP Dealer location, product information, prices, Users' Library information, and service prices and status for TDD (telecommunications devices for the deaf) and TTY (teletypewriter) users with hearing or speech impairments. Please dial 503-758-5566 (not toll-free).

Addendum for 82180-90001 Manual

If you own the *HP 82180A Extended Functions/Memory Module Owner's Manual*, part number 82180-90001, dated November 1981 or April 1982, you should mark the following changes in your copy of the manual. Later versions of the manual will have either the Addendum with them or the corrections will be incorporated in the manual.

Page 8, under Configurations. If you have the HP 82104A Card Reader plugged into the calculator and an HP 82181A Extended Memory Module plugged into port 2, and you execute the card reader function [VER], some information in that extended memory module may be changed. Therefore, you should avoid using the [VER] function if you are also using an extended memory module in port 2.

Page 17, under Clearing Programs. If you execute [PCLPS] from the keyboard, be sure the calculator is positioned in program memory. You can position the calculator in program memory in any of the following ways:

- Press [CATALOG] [1] followed by [R/S] (as described under "Using [CATALOG] for Positioning" in your calculator owner's manual).
- Press [GTO] [ALPHA] *label* [ALPHA] using a label in program memory (one that is listed in CATALOG 1).
- Press [GTO] [.] [.].

If the calculator is positioned to a program in a plug-in application module or device when you execute [PCLPS], the information in the calculator's memory will be lost and the calculator will display **MEMORY LOST**.

Page 24, under [PURFL]. After a file in extended memory is purged, there is no working file. Therefore, before subsequently executing functions that operate on the working file, you should execute a function (such as [SEEKPTA] that defines the working file (that is, makes the specified file the working file-refer to "Working Files," page 23). For example, after executing [PURFL], write the name of an existing file in the ALPHA register, then execute [FLSIZE]—that file now becomes the working file. After executing [PURFL], you should *always* define a working file before executing functions that operate on it; otherwise, all files in extended memory will be lost.

Page 25, before Program File Operations. If a register in a file contains a string of seven characters all having character code 255, and if another file closer to the beginning of extended memory is purged, then all information from that register to the end of extended memory may be lost. To ensure that this doesn't occur, avoid appending, inserting, or adding to a file more than six consecutive characters having the character code 255.

Page 25, under Program File Operations. If you execute [SAVEP] from the keyboard, be sure the calculator is positioned in program memory (as described above). If the calculator is positioned to a program in a plug-in application module or device when you execute [SAVEP], the information in the calculator's memory and in extended memory may be changed or lost.

New HP-IL Video Interface Available

The new HP 82163A Video Interface will be available at your local HP Dealer sometime in June—probably by the time you are reading this KEY NOTES. With this new device, the HP-41 will be capable of displaying on a monitor or TV, 96 Standard ASCII or inverse video characters, with up to 32 characters per line and 16 lines per display. The HP-41 also will be capable of screen control and cursor control. The HP 82180A Extended Functions Module will help to create the necessary commands to enable screen and cursor control.

The new HP-IL Video Interface is *all* that is needed if you want just a convenient way to output data and programs in video. The HP-41 printer commands in this new module will allow all normal character output to occur in a very friendly manner. Flags 15 and 16 in the HP-41 will control the PRINT (output) mode to the video display. For example, when flag 15 is SET, it will make the video display act like a printer in TRACE mode; and when flag 16 is SET, it will make the video display act like a printer in NORMAL mode. And, when both flags are SET, they will put the

(Continued)

video display into a new "TRACE WITH STACK OPTION" mode that will cause stack registers X, Y, Z, and T to be "printed" (displayed) after each operation. Now you'll be able to "see" how you left the stack and what is in it! (Very nice, yes?) Finally, with both flags CLEAR, the video display will be in MANUAL mode.

Best of all, this new video interface is an HP-IL device! Therefore, it will be compatible with future HP-IL devices. And, although you may be excited about "seeing" your HP-41's innermost "secrets" on TV, be sure you call your local HP Dealer before you rush down to the store. This is going to be a popular peripheral, so make sure the Dealer has one and can demonstrate it for you.

Add "Racing Stripes" to Your Bar Code

In KEY NOTES V5N3 we proposed a method to increase security for programs reproduced in bar code. The method consisted of printing the bars over red background. After some further testing (including some tests volunteered by Richard Nelson and Noel Brinkley of PPC), we are back with the promised update.

It was found that some copiers checked relative contrast between printed information and a background color. These copiers could easily produce duplicates that a wand will read.

Introducing the racing-stripe approach! Several patterns and narrower stripes were tested. The stripes proved the most successful, especially when printed to overlay the leading and trailing edges of a row of bar code. These areas appear to be the most critical for the wand to successfully read the code.

The second discovery was that a thermal copier would not reproduce the red (as gray) because there is no carbon content in red ink. This problem is easily overcome by printing the bars using a carbonless black ink. Now the thermal copier won't "see" the bars either!

A combination of the two techniques should produce bar code that cannot be reproduced successfully by over 95% of the copiers on the market.

In case you missed the specifications earlier, the color of inks recommended (all "reds") in order of preference are: PMS #199, PMS #485, PMS #185, PMS Warm Red.

A Change is in Order

Starting June 1, 1982, Corvallis Division will no longer accept direct-mail and telephone orders for calculators and accessories. To order calculators and accessories in the future, please contact your local authorized HP Dealer. They should have stock on hand for immediate purchase. To locate your nearest HP Dealer in the continental United States except Oregon, please call toll-free (800)547-3400. In Oregon, Alaska, and Hawaii, call (503)758-1010.

If you do not live near an HP Dealer, you can order calculators and accessories through the HP Corporate Parts Center. They will accept purchase orders ($20 minimum), checks, money orders, and cash. However, C.O.D., credit card, and telephone orders will not be accepted. Be sure to include a $3.50 handling charge, plus your state and local taxes. Mail your orders to:

Hewlett-Packard Company
Mail Order Department
P.O. Box 7220
Mountain View, CA 94042

This notice and the above changes apply only to domestic orders. Neither Corvallis Division nor the above Mail Order Department will handle orders from anywhere outside the U.S.

The Corvallis Users' Library will continue to accept and process orders for subscriptions (KEY NOTES and Library) and programs. **But remember that the Library does not accept purchase orders for orders under $20.**

KEY NOTES Corrections

Here is a letter from **Nai Chi Lee**, of Stony Brook, New York, concerning an error (our error) that appeared in the last issue of KEY NOTES.

With reference to my integer-packing routine printed in V6N1p11, please be informed of the following:

1. The program will not work as printed! The error is in line 16. Delete it and replace with -1, ABS (this is not equivalent to 1).
2. In the description, paragraph 2, first line: "number uvwxyz" should be "number abcdef."

Also, in V6N1, on page 13, column a, the second line in the routine MAG should be LBL A; and the name of the author of this routine is **Jeffrey Smith** not Jeffery Smith. Sorry, Mr. Smith. In V6N1p11b, the second paragraph of the description preceding **John Hendricks'** PRDMS routine should read: For HHhMM'SS, substitute 104, ACCHR for lines 08 through 19.

The description of the "Stock Plotting" routine on the back page of V6N1 fails to mention that the calculator must be set to a minimum size of 18 and that the printer mode switch must be set to MAN.

The "In the Key of HP" article in V5N3 on page 7 has a confusing sentence in the description of the modulo function. The second sentence of the second paragraph of this description should read: A number can be determined to be within a certain range if RANGE MOD X < RANGE.

In V5N2, there is a typographical error on page 8. In answer 22 of the answers to the HP-67/97 test, STO 1 should be STO I.

New Club Formed in Germany

If you can read the German language, you might be interested in a new Club that was founded in November of 1981. Called CCD, for Computerclub Deutschland, the Club had over 1,000 members by February 1982! And all of the members are HP-41 users.

The Club prints a monthly newsletter, PRISMA, that contains many HP-41 programs, technical notes, Synthetic Programming, self-produced hardware, and much more. The newsletter is entirely written in German. Nearly all of the programs are for professional use, but a few games are printed. And all printed programs include bar codes.

Regular Club meetings are being held in nearly every large city in Germany, and members come from nearly all West-European countries. In German law, the Club is a "gemeinutzig anerkannter eingetragener Verein," which, in English, means a nonprofit, legal club, controlled by public commissions. However, there is a fee for membership.

If you are interested in the CCD, write to the first chairman:

Oliver Rietschel, 1. Chairman
Computerclub Deutschlande. V.
P.O. Box 373
2420 Eutin, W. Germany

Be sure to include a self-addressed, stamped envelope. If you can't include stamps, you can include at least two magnetic cards.

(Note: CCD is not sponsored, nor in any way officially sanctioned, by Hewlett-Packard—Ed.)

Linear Interpolation On The HP-41

Here is a routine from the biggest state in the "lower 49." Austin is the capital of this state—Texas— and it is the home of **Philip Petersen**, who sent this routine.

(41) This short routine linearly interpolates multiple times after initial conditions are set, using only register 00 for safekeeping. Besides the two ALPHA prompts (that take up many bytes), the routine is at peak efficiency and optimum ease of use.

```
01♦LBL "IN"        10♦LBL 00
02 "KNOWN?"        11 ST- Y
03 PROMPT          12 X<>Y
04 ST- Z           13 RCL 00
05 -               14 /
06 /               15 +
07 STO 00          16 STOP
08 "UNKNOWN?"      17 GTO 00
09 PROMPT          18 .END.
```

Operating instructions: For tabular data such as,

a	x
r	(unknown value)
b	y

where a corresponds to x, b corresponds to y, and r corresponds to the unknown value.

At the prompt—KNOWN?—key in: a [ENTER] r [ENTER] b [R/S]

At the prompt—UNKNOWN?—key in: x [ENTER] y [R/S]

Example problem:

Given the following properties of steam at 1.5 MPa and 1.75 MP, estimate the values of T, Vf, and Vg at 1.58 Mpa.

P(MPa)	T(C)	Vf	Vg
1.50	198.32	0.001154	0.13177
1.58	?	?	?
1.75	205.76	0.001166	0.11349

Solution:

KEYSTROKES	DISPLAY
[XEQ] [ALPHA] IN [ALPHA]	KNOWN?
1.5 [ENTER]	
1.58 [ENTER]	
1.75 [R/S]	UNKNOWN?
198.32 [ENTER]	
205.76 [R/S]	200.700800 (T at 1.58MPa)
.001154 [ENTER]	
.001166 [R/S]	0.001158 (Vf at 1.58)
.13177 [ENTER]	
.11349 [R/S]	0.125920 (Vg at 1.58)

On Creating and Using Cassette Data Files

Data files must be "created" before they can be used. Unlike program files, a data file has to be created so that you can specify its size. With the HP-41, the HP 82160A HP-IL Interface Module, and the HP 82161A Digital Cassette Drive, a file is created by placing into the ALPHA register the name of the file (not more than 7 characters), and by placing in the X-register the number of required registers. For example, if we key in: [ALPHA] BILL [ALPHA] 3000 [XEQ] [ALPHA] CREATE [ALPHA], the cassette drive will "create" a file named "BILL" and the file will have 3000 registers.

Now, suppose we want to store 10 registers (R_{10}-R_{19}) of information into cassette file "BILL" at locations 100 through 109. All we need to do is key in: [ALPHA] BILL [ALPHA] 100 [SEEKR]* 10.019 [WRTRX]. The first part of this sequence will position the cassette drive at register 100 in file "BILL," and the second part will write HP-41 registers 10 through 19 into cassette drive registers 100 through 109.

But suppose we now want to recover those same 10 registers of information and place them into registers 00 through 09 in the HP-41. That's no problem. Just key in [ALPHA] BILL [ALPHA] 100 [SEEKR] 0.009 [READRX] and you can view those 10 registers in the HP-41 by recalling registers R_{00} through R_{09}. In essence, we have made a duplicate copy of the information in R_{10} to R_{19} and put it into R_{00} to R_{09}.

** You can assign these functions to keys or use the form: [XEQ] [ALPHA] SEEKR [ALPHA].*

Sending 8-Bit Data With the HP-41

(Note: This article presents the theory behind Library Program #01644C, Bar Code Generator, which appears in this issue of KEY NOTES—Ed.)

There are two methods currently available to the HP-41 user who wants to build 8-bit binary bytes of any value. (For example, in constructing bar code such as that used in Library Program #01644C.) The easiest method uses the XTOA function contained in the HP 82180A Extended Functions Module. The other, a more difficult method, uses the BLDSPEC function contained in the HP 82160A HP-IL Interface Module. It is this second method that forms the topic of the following discussion.

First, take a quick glance at figure 1. Now, if we were to look at the decimal equivalent value of each bit in a binary byte, we would find that the value of any byte with bit 7 set (the high order bit) would yield a value equal to or greater than 128.

128	64	32	16	8	4	2	1	Bit Value
x	x	x	x	x	x	x	x	A binary 8-bit byte
7	6	5	4	3	2	1	0	Bit number

Figure 1. A Binary 8-Bit Byte

The byte value is the sum of all of the bit values.

Bit Set	Bit Value	128	64	32	16	8	4	2	1	Bit Value
1	2	1	0	0	0	1	0	1	0	
3	8	7	6	5	4	3	2	1	0	Bit Number
7	128									
	138 ← BYTE VALUE									

Figure 2. Byte Value Example

Creating a byte (or bit pattern) using BLDSPEC is easy, and a simple matter, until we exceed 127 (decimal). But, first, take a look at figure 3 in order to create a byte with a value of 0 to 127.

Creating a binary byte with a decimal value of 32.	CLX ENTER↑ 32 BLDSPEC

Figure 3. Creating a Byte Value 0 to 127

This will leave in the X-register an alpha data byte—seen as a character—that is actually a bit pattern equivalent to the decimal value used prior to the BLDSPEC command. The CLX and ENTER commands are used to ensure that the Y-register is clear and to prevent the BLDSPEC command from making an unwanted combined pattern. However, being able to combine patterns *is* useful in making byte values of 128 or greater. Keying in 127, BLDSPEC would yield a bit pattern of 01111111. To set the 8th bit (which is actually bit 7; see figure 1), follow the example in figure 4.

Creating a binary byte with a decimal value of 160.	CLX ENTER↑ 1 BLDSPEC 32 BLDSPEC

Figure 4. Creating a Byte Value Greater Than 127

The 1, BLDSPEC followed immediately with a number 0 to 127 BLDSPEC will create byte values from 128 to 255.

As these special characters (byte values) are built in the X-register, they should be recalled into the ALPHA register. When the string is complete, the OUTA function is used to send it to the selected device.

This technique was used in the Bar Code Generator program (#01644C); however, it would be valuable to anyone wishing to send binary bytes, other than the standard ASCII set, to any HP-IL device.

(Because of the present data-handling routines in the HP-41, null or zero bytes cannot be used as the first byte in a data string. Additionally, the OUTA function ignores them even in the middle of a string. Life is never perfect, n'est-ce pas?*—Ed.)*

Generating Bar Code With the HP-41

Because many readers have expressed a desire to be able to "create" bar code at home or at the office—or anywhere, for that matter—we have included in this issue a program that will satisfy *most* of that desire. If that word "most" gives you the idea that this is not a *perfect* solution for home-generated bar code, you are right. But this program *will* enable you to print usable bar code.

There are some basic operating limits you need to know. First, you **must** use the new HP 82160A HP-IL Module and the new HP 82162A Thermal Printer. Second, you **must** use the new black HP 82175A Thermal Printing Paper. Third, you **should** read, in this issue, the article, "Sending 8-Bit Data With the HP-41," because it explains the theory behind the following program and a few software limitations. But make no mistake about this: With the HP-41 the HP-IL Module, the new printer, the new black paper, and the following program, **you can generate immediately usable bar code.** Here is an expanded abstract.

(41) Bar Code Generator #01644C (Price: $6*)

This program, when coupled with knowledge from the 82153-90019 *Creating Your Own HP-41 Bar Code* manual (pages are included in program), will allow the user to print up to 16 continuous bytes of bar-coded information. Required inputs are made as the decimal equivalent of the binary code. The program uses the BLDSPEC function to create the binary codes needed to print bar code. This entire operation could be

(Continued)

simplified by using the XTOA COMMAND IN THE HP 82180A Extended Functions Module. However, this program is designed for the users who do not have other modules but still want to make their own bar code. So the Extended Functions Module is NOT required for this program. *Required accessories: HP 82160A HP-IL Module and HP 82162A Thermal Printer.* (74 lines, 189 bytes, 13 pages)

Author: **Bill Schafer**
Corvallis, Oregon

Although this method does work, for really first-class long-term bar code that you might want for your day-to-day program use, you can't beat manufactured bar code. Look back to page 5 of V5N1 and review the article about our bar code supplier, **George Lithograph.** Their service is good, their product is excellent, and the price is right. They print all of the bar code that you see in the Library programs. For more information regarding this service and an order form/price list, contact:

Dan Riopel
George Lithograph
650 Second Street
P.O. Box 77085x
San Francisco, CA 94107 USA

What's So Great About the "New" HP-IL Printer?

If you have had reservations about buying the new HP 82162A Thermal Printer, here are some features you might not have thought about. When you examine these at the counter display in your local HP Dealer's store, you will get a better insight of what this new printer can do for your system.

FORMAT: The FORMAT function allows you to automatically center text, or you can even left- or right-justify two columns of text or numbers or even a combination of both. For example, make a simple grocery list as follows:

[ALPHA] GROCERY [ALPHA] [ACA] [FMT] [PRBUF] You will notice that GROCERY will be automatically centered on the printout. Now, list the items and prices.

[ALPHA] MILK [ALPHA] 1.05 [ACA] [FMT] [ACX] [PRBUF]
[ALPHA] EGGS [ALPHA] 1.10 [ACA] [FMT] [ACX] [PRBUF]
[ALPHA] BREAD [ALPHA] .85 [ACA] [FMT] [ACX] [PRBUF]

Actually, this could be made into a simple subroutine that would do this task for *any* listing.

PARSE: The PARSE mode, as demonstrated here, needs the Extended Functions Module to simplify its implementation. However, if you look at the Bar Code Generator program on page 9, you will get some ideas on how to use BLDSPEC to implement PARSE without using the Extended Functions Module. The PARSE mode will be supported in a much more "friendly" manner with a future "extension module" (we're still working on it); however, it *is* accessible.

The printer in this example is at device location 2 (see HP-IL Module Owner's Manual, page 43). Key in: 2, [SELECT] [CLA] SF 17 252 [XTOA] 27 [XTOA] 38 [XTOA] 107 [XTOA] 49 [XTOA] 72 [XTOA] [ALPHA] ASTO 01 [ALPHA] [OUTA]. This sequence sets your printer to PARSE mode. The ALPHA string used to set the printer to PARSE mode is now stored in register 01 and can be used again ([ARCL] 01 [OUTA]). The SF 17 suppresses the printer carriage return and line feed. Now, "type-in" the following:

[ALPHA]—I-WANT-TO-SEE-PARSE-[ALPHA] [ACA]
[ALPHA] MODE-OPERATE-WITHOUT-THE-
[ALPHA] [ACA]
[ALPHA] EXTENDED-I/O-ROM-INSTAL
[ALPHA] [ACA]
[ALPHA] LED. [ALPHA] [ACA] [PRBUF].

This might not be the best operating mode, but you have to admit it *does* work.

BARCODE: Refer to the Bar Code Generator (#01644C) program and the article "Sending 8-bit Data With the HP-41" that appear elsewhere in this issue.

STANDBY MODE: This one is easy! From the ON position, just set the power switch to STANDBY, then key in: [XEQ] [ALPHA] PWRDN [ALPHA]. You will notice that the POWER Light is now off, but the HP-41, as the controller in the HP-IL system, can turn on the printer and shut it off.

However, nothing is totally "free." We do not recommend leaving your printer continuously in STANDBY mode during normal operating conditions; that is, using your computer system at your desk, recharger plugged in, etc. The STANDBY mode (on both the printer and the cassette drive) is provided to conserve battery power during "remote" or automatic, unattended conditions. (In other words, when the calculator is set to run a program unattended.) The recharger/AC adaptor might not always be able to keep up with the recharge rate plus supply continuous power for operations, so use STANDBY Only for its intended purpose. However, you must admit that it is a powerful and extremely useful new mode.

Both of the HP-41 printers are excellent additions to your portable computer system, but if you own the HP-IL Module, you owe it to yourself to check the added features of the new HP 82162A Thermal Printer at your local HP Dealer. Don't forget to take along this article!

You, HP-IL, and Control Functions

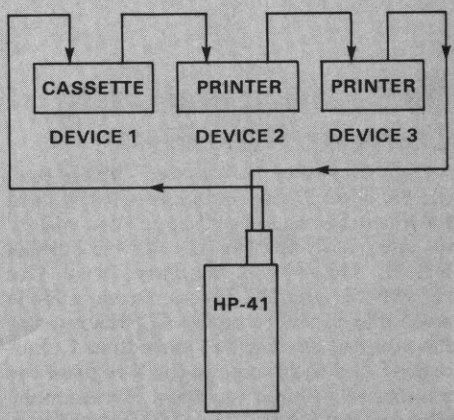
Have you asked yourself: "What good are the general I/O functions in the HP 82160A HP-IL Module?" You have? Well, then, let's take a look at some of them.

CONTROL FUNCTIONS

On page 43 of the *HP 82160A HP-IL Module Owner's Manual,* the general I/O functions are referred to as "Interface Control Operations," There are 15 of these functions and, as it so happens, for some applications they are the only method that can be used to make certain things happen on the Interface Loop. Although these functions are really very basic, they are a powerful set that allows a programmer to access information on just about any HP-IL peripheral.

Because it is not possible to discuss all of these Control Functions in this short column let's look at just a few.

AUTOIO: As you know, the HP-41 was designed to be a fully "friendly" device. The AUTOIO (automatic input/output) function also was designed to continue this trend in its I/O operation. In this mode, the HP-41 will automatically locate friendly printer and mass storage peripheral devices connected to it, then initiate printing when it's supposed to occur, and also interact with the mass memory (e.g., HP 82161A) when it is supposed to transfer data and/or programs. In either case, the HP-41 will automatically go to the first available printer or mass memory device in the Interface Loop and attempt its operation there. For example, if you have two printers in the Interface Loop and you want to cause an output on the *second* printer (see diagram), then you must "select" that device prior to sending the data to it. AUTOIO is, however, the normal "cold start" condition.



SELECT: If you want the second printer in the Interface Loop to be the active device, then you must select that device. As great as the HP-41 is, unfortunately it cannot read your mind. So here's how it does this. There are two different plug connectors for the HP-IL Module, a male (small) and a female (large) connector. The male connector indicates the direction of data flow around the Interface Loop and thus controls the location of devices in the loop. The Controller (the HP-41) is *always* device 0 and, counting around the loop in the direction of the male connectors (designated by the arrow) in the above diagram, the second printer is at device location 3. So, if you want that particular printer to be the primary printing device, key in the following:

3 [XEQ] [ALPHA] SELECT [ALPHA]

Now, all you need to do is execute a PRX or another print-command, and the second printer will happily start printing whatever you are sending to it.

MANIO: How many times have you wanted to do things with your HP-41 computer system that were not necessarily "the norm?" Well, MANIO allows for such times by giving you more direct control over what happens on the Interface Loop. An example of this can be found right here in this issue of KEY NOTES, wherein we wanted to create bar code on the HP 82162A Thermal Printer without using an "extension" module. Placing the printer in the MANIO mode allows the HP-41 to send 8-bit commands for bar code quite easily. (The 8-bit mode instruction set is shown in tabular form on page 14 of the owner's manual for the HP 82162A.) But, in order to print bar code, you first must understand what bar code really is. And a good way to learn is to look at program #01644C and the article, "Sending 8-Bit Data With the HP-41," appearing elsewhere in this issue. Included with the documentation for #01644C are some pages from the book, "Creating Your Own HP-41 Bar Code." (This HP book was reviewed in V5N1p5.)

To send 2 bytes of bar code, you first must put the printer in column mode, then tell it that you want to send 2 bytes of bar code. Using the HP 82180A Extended Functions Module and the following routine you will be able to do this.

Notice that, in this example, the printer is at location 2, which explains lines 02 and 03. And lines 10 through 13 specify the byte values of decimal 32 and 16.

```
01♦LBL "BCDE"     10 32
02 MANIO          11 XTOA
03 2              12 16
04 SELECT         13 XTOA
05 CLA            14 OUTA
06 210            15 AUTOIO
07 XTOA           16 RTN
08 129            17 .END.
09 XTOA
```

In the next issue (due in the mail by August 26), we will present more articles of this nature.

67/97 Routines and Tips

Pretoria is a capital city of South Africa located close to some major diamond mining areas. Also, Pretoria is the home of **J.M.E. Graindor**, the author of this next routine.

(67/97) Many programming tips referred to the use of the HP-97 as a timer. Although we fully realize the inaccuracy of the calculator as a timer, it nevertheless is possible to use it as a rather accurate digital watch.

Here is a surprisingly short routine to this effect. Key in the present time in format HH.MMSS, press [A], and presto, your digital watch is running. If you perceive that the watch is somewhat slow, key in, during a pause display, a small number (usually 1 or 2), and press [E] to adjust the watch. Repeat if necessary. Similarly, if the watch is somewhat fast, a negative 1 or 2 will do the trick. For a fine setting, a fractional number like 0.5 or 0.2 might be required.

It is very interesting to note that, similar to a real digital watch, this watch also switches from 24.0000 hours to 0 hours.

```
001  *LBL9      015    4
002   DSP4      016  RCL0
003   HMS→      017  X>Y?
004   STO0      018  ST-0
005     5       019  →HMS
006     4       020   PSE
007   EEX       021  GTO1
008   CHS       022 *LBLE
009     5       023   EEX
010   STO1      024     5
011  *LBL1      025     ÷
012   RCL1      026  ST+1
013   ST+0      027   RTN
014     2
```

And, back in the United States, **Carl Vancini** is busy working with his HP-97. Mr. Vancini lives in Stamford, Connecticut.

(67/97) I noticed that the short routine "Weighted Mean," developed for the HP-67/97 by D.L. King (V5N2p10b) does not produce the standard deviation. Here is a routine that produces the weighted mean with the corresponding standard deviation. The only limitation is that the weights shall be non-zero integers (1,2,3,...). To run, clear all registers, key-in the number, press ENTER, key-in the weight, press b. After all numbers and their weights are entered, press $\bar{x}$ to obtain the weighted average, then press s to obtain the standard deviation.

```
001  *LBLB      008  *LBL6
002     1       009    Σ+
003   ST+0      010  LSTX
004   X≷Y       011  DSZI
005   X≷I       012  GTO6
006   R↓        013  RCL0
007   X≷Y       014   RTN
```

Now, here is a short and interesting routine. **Mathew Tomczik** of St. Cloud, Minnesota contributed this.

(67/97) I have found a special routine on my HP-67 to use when trying to find the log, in any base, of any number. $\text{Log}_b\, x$: $x > 0$, $b > 1$.

```
001  *LBLA      005   R↑
002     LN      006    ÷
003   R↓        007  RTN
004     LN
```

Example	Input	Output
$\text{Log}_3\, 9 = 2$	9 [ENTER] 3 [A]	2
$\text{Log}_{81}\, 3 = ¼$	3 [ENTER] 81 [A]	.25
$\text{Log}_{1000}\, 10 = ⅓$	10 [ENTER] 1000 [A]	.333

Routines, Techniques, Tips, etc.

The routines and techniques furnished in this column are contributed by people from all walks of life and with various levels of mathematical and programming skills. While the routines might not be the ultimate in programming, they do present new ideas and solutions that others have found for their application. You might have to modify them to fit your personal application.

Now, from San Diego, California, we present **Jaroslaw Czajowski's** "song and dance" routine. How much is that goose in the window? The one that goes [BEEP] [BEEP] [TONE] 9...

(41) This subroutine is an improved version of the "Flying Goose" routine that appeared in V3N4p3, with a little spice of sound added to it. Run, see, and hear it!

```
01♦LBL "GOOSE"    18♦LBL 01
02 SF 01          19♦LBL 01
03♦LBL 01         20♦LBL 01
04 9              21 FS? 01
05 STO L          22 TONE 1
06 FC?C 01        23 FC? 01
07 SF 01          24 TONE IND L
08♦LBL 15         25 DSE L
09 FS? 01         26 GTO 15
10♦LBL 01         27 .009
11♦LBL 01         28 STO L
12♦LBL 01         29♦LBL 16
13♦LBL 01         30 TONE IND L
14♦LBL 01         31 ISG L
15♦LBL 01         32 GTO 16
16♦LBL 01         33 GTO 01
17♦LBL 01         34 .END.
```

The mention of Oak Ridge, Tennessee, brings to mind the sweet sound of a bluegrass mandolin echoing over the rolling green hills on a muggy, lazy, midsummer day. Thoughts turn to base-conversion on the HP-41, and we bring you this routine from **Richard Lyon**, who lives in Oak Ridge.

(41) I enclose a short program for converting an integer between 1 and 100,000,000 (base 10) to another integer base between 2 and 16. The result is recorded in the ALPHA-register, and the individual digits can be copied down as they are posted if the result contains more than 12 digits, or even if digits will be lost from the ALPHA-register, as in the conversion of numbers greater than 16,777,215 to base 2. Posting of the digits can be slowed down by squeezing-in a delay loop, such as:

```
30a 20
30b STO 18
30c ST+ Y
30d −
30e LBL A
30f DSE 18
30g GTO A.
```

(Continued)

With 20 as the input to register 18, the delay is about 4 seconds between postings of successive digits on my HP-41C, with my current battery condition.

Initially, ALPHA characters (0,1,...9,A,...F) for the sixteen digits must be stored in registers 00 to 15.

The principle behind the program is that after finding the first (lefthand) digit by successive divisions by the base number, the fractional part of the quotient will disgorge successively lower digits when multiplied successively by the base number.

Possible round-off error limits the initial integer to about 100,000,000 and also requires addition of a fractional increment to the initial integer. I use 0.5.

```
01♦LBL "F10"
02 1.1
03 STO 16
04 "NMBR"
05 PROMPT
06 STO 17
07 .5
08 +
09 "BASE"
10 PROMPT
11 STO Z
12 STO T
13♦LBL 00
14 X>Y?
15 GTO 01
16 /
17 X<>Y
18 ISG 16
19 GTO 00
20♦LBL 01
21 .1
22 ST- 16
23 CLA
24 RDN
25 X<>Y
26♦LBL 02
27 RCL X
28 INT
29 ARCL IND X
30 AVIEW
31 -
32 *
33 DSE 16
34 GTO 02
35 .END.
```

C. Lamar Williams, previously known for his elaborate work with letter-banner printing on the HP-41 (V5N3p4), is keeping up the good work in San Jose, California. Here is one result of his work.

(41) Budget planners frequently need to distribute an odd amount of money (Z) over some odd number of time periods (N). This distribution usually needs to be nearly uniform, i.e., approximately the same amount of money for each individual period. However, the total amount of money (Z) must be correct; i.e., it cannot suffer round-off errors when summing the "amount" for all of the periods. The following routine "DISTR" does the trick! Just key-in the total number of periods (N), press [ENTER], key-in the total amount (Z), [XEQ] "DISTR", and witness the answer. This routine works with or without the printer.

The problem is:

$Z = AX + BY,$

$N = A + B,$

where each variable is an integer; given N and Z, find A, X, B, Y.

The solution is:

$A = MOD(Z,N)$

$B = N - A$

$Y = INT(Z/N)$

$X = Y + 1$

Program lines 04-20 solve for A, B, X, and Y. The other program lines are for display with or without the printer. Have fun...!

```
01♦LBL "DISTR"
02 FIX 0
03 CF 29
04 STO 00
05 X<>Y
06 STO 01
07 MOD
08 STO 02
09 RCL 01
10 X<>Y
11 -
12 STO 03
13 RCL 00
14 RCL 01
15 /
16 INT
17 STO 04
18 1
19 +
20 STO 05
21 CLA
22 ARCL 00
23 "⊢ OVER"
24 XEQ 01
25 CLA
26 ARCL 01
27 "⊢ PERIODS="
28 XEQ 01
29 CLA
30 ARCL 03
31 XEQ 03
32 ARCL 04
33 "⊢ +"
34 XEQ 01
35 CLA
36 SF 01
37 ARCL 02
38 XEQ 03
39 ARCL 05
40♦LBL 01
41 AVIEW
42 FS? 55
43 GTO 02
44 PSE
45 PSE
46 PSE
47 PSE
48♦LBL 02
49 FC?C 01
50 RTN
51 ADV
52 ADV
53 ADV
54 ADV
55 ADV
56 ADV
57 RTN
58♦LBL 03
59 "⊢ AT "
60 RTN
```

Don MacDonald of Newport Beach, California, sent this next tip. And, in his letter, he asked that we give credit to PPC member **Tim Kay**.

(41, 67, 97, ...) Here is the quickest way to determine if x is even or odd:

```
-1
X≷Y
Y^X
```

You will see 1 if x is even and -1 if x is odd. These steps leave x in the LAST X register.

(On the HP-41, the sequence: 2; MOD, will work as well. However, x will not be preserved in the LAST X register—Ed.)

Now, from London, England, we have this tip. It was sent to us by **W.A.C. Mier-Jedrzejowicz**.

(41) If you want to prompt for a value in the middle of a program, and to store the value to use it, but you do not want to disturb the stack, you can do it neatly on the HP-41 as follows:

```
01 X≷nn
02 RDN
03 "message"
04 PROMPT
05 X≷nn
```

Line 01 saves the present X-value in register nn; line 02 rolls the stack down in preparation for line 04; line 03 is the ALPHA message requesting the new value. In line 04 the new value lifts the stack, restoring Y, Z, and T; line 05 saves the new value in register-nn and restores X.

This next contribution describes an interesting technique that many will find useful. It came to us from **Humbert Hans Suarez**, who is a student at the University of Geneva in Switzerland.

(41) There is one problem that often occurs while creating HP-41 programs: You have two programs, one that begins with, say, LBL "MAIN", and one that begins with LBL "USER" that has a LBL 01, a LBL 02, and a LBL 03. The program "MAIN" wishes to call as subroutines sometimes LBL 01, sometimes LBL 02, and sometimes LBL 03. But the calculator searches numeric labels solely in the program from which these labels were called.

One solution could be to put global ALPHA labels by the LBL 01, LBL 02, and LBL 03. However, this solution consumes lots of memory space and search time. A better solution is to put a GTO IND X at the beginning of the "USER" program and to call LBL 01 by—1; XEQ "USER", and LBL 02 by—2; XEQ "USER", and so on.

1st solution	2nd solution
LBL "MAIN"	LBL "MAIN"
⋮	⋮
XEQ "1"	1
⋮	XEQ 00
XEQ "2"	⋮
⋮	2
XEQ "3"	XEQ 00
⋮	⋮
END	3
⋮	XEQ 00
LBL "USER"	⋮
⋮	LBL 00
LBL "1"	XEQ "USER"
⋮	⋮
LBL "2"	END
⋮	⋮
LBL "3"	LBL "USER"
⋮	GTO IND X
END	⋮
	LBL 01
	⋮
	LBL 02
	⋮
	LBL 03
	⋮
	END

(This is a good demonstration of "the law of conservation of bytes through conservation of global-ALPHA labels," Mr. Suarez; however, your second solution does not conserve search time, as implied by your letter. The first solution requires one global label search, while the second solution requires a global label search, and then—because of the indirect GTO statement—a step-by-step local label search. So, our advice is: weigh the byte savings against the increased execution time before using this technique—Ed.)

This technique came to us in a letter titled, "How to Double Your Local (ALPHA) Labels From 15 to 30." Undoubtedly, this method has its drawbacks but, then again, it may be just what you're looking for. The letter was sent to us by **Vandale John**, who lives in Adinkerke, Belgium.

(41) This way of programming is very interesting in programs made for such conversions as feet<>meters, inch<>mm, gallons<>liters, Farenheit<>Celsius, and so on. For example:

```
01♦LBL "F<>M"     07 STOP
02♦LBL A          08♦LBL 01
03 X=Y?           09 RDN
04 GTO 01         10 .3048
05 .3048          11 /
06 *              12 .END.
```

How to use:

(a) Position the calculator to the program (GTO [ALPHA] F<>M [ALPHA]);
(b) Press [USER] on;
(c) To convert 2.25 feet to meters, press 2.25 [A], and see 0.6858;
(d) To converting 0.6858 meters to feet, press 0.6858 [ENTER] [A] and see 2.25 feet.

As you can see, the [ENTER] key becomes a shift function. This gives us 30 labels—A, A', B, B', ... a, a', ... e, and e'—good for 15 conversions.

Here's a routine to help us through the day. **Don Thayer** of San Leandro, California, sent in this one.

(41) While converting my HP-67 programs for use with my HP-41 and its printer, I discovered many of them required the conversion from a Gregorian Date to the equivalent Julian Day number. Not wanting to restructure the data register assignments, I developed the following program, titled "DAY," which uses only 72 bytes and does not require any data registers other than the stack. The routine is valid from March 1900 through February 2100 and was derived from the Standard Pack for the HP-67.

To use, enter the date in the form MM.DDYYYY and execute DAY. Example: July 4, 1980 (7.041980) = 2,444,425.

```
01♦LBL "DAY"      20 +
02 ENTER↑         21 1
03 INT            22 ST- Z
04 STO Z          23♦LBL 00
05 -              24 CLX
06 1 E2           25 1
07 *              26 +
08 ENTER↑         27 30.6001
09 INT            28 *
10 STO T          29 INT
11 -              30 X<>Y
12 1 E4           31 365.25
13 *              32 *
14 X<>Y           33 INT
15 3              34 +
16 X<=Y?          35 +
17 GTO 00         36 1720982
18 CLX            37 +
19 12             38 .END.
```

Next, we have two nice inputs from **Robert Swanson** of Portland, Oregon. This is the man who brought us the extraordinary game program "Flipo" (V6N1p5a), and both of the methods that he is discussing here were used to create that program.

(41) If a program does not otherwise use the trigonometric mode flags (flag 42, GRAD mode; flag 43, RAD mode) for trigonometric instructions, then one or both of them may be used as general purpose flags. Some advantages are: (1) display annunciators; (2) setting (GRAD, RAD) and clearing (DEG) are one byte instructions; (3) the execution time to set and clear these flags is only 20 msec (compared to 30 msec for other user flags); and (4) the status is maintained when the calculator is off. Remember, however, that DEG clears either of the flags and that only one of the two flags can be in set mode at a given time. Flags 42 and 43 are particularly useful in programs with three-way branch points. The following routine, "LOGIC," is an implementation of the truth table for the three types of AND logic, and also illustrates the use of the trigonometric mode flags at a three-way branch point. In this example, the user determines the branch (lines 11 through 14) by pressing key 11 (XEQ A), 12 (XEQ B), or 13 (XEQ C) in USER mode.

```
01♦LBL "LOGIC"    19 GTO 04
02♦LBL B          20♦LBL 02
03 GRAD           21 X<Y?
04 GTO 01         22 X<=0?
05♦LBL C          23 FS? 30
06 RAD            24 GTO 05
07 GTO 01         25 GTO 04
08♦LBL A          26♦LBL 03
09♦LBL 01         27 FS? 01
10 "DO"           28 X=0?
11 FS? 42         29 FS? 30
12 GTO 02         30 GTO 05
13 FS? 43         31♦LBL 04
14 GTO 03         32 "SKIP"
15 FS? 01         33♦LBL 05
16 FC? 02         34 DEG
17 FS? 30         35 PROMPT
18 GTO 05         36 .END.
```

Instructions: Have your calculator in DEG mode. To test flag:flag logic, set or clear flag 01 and flag 02 as desired, then XEQ A. If both flags are set, the routine halts with the message DO in the display; otherwise, SKIP appears. Arrange the stack at will and XEQ B to test stack:stack logic. See DO whenever $x < y$ AND $x > 0$. Finally, XEQ C and observe that DO is displayed only when flag 01 is set AND $x \neq 0$ (flag:stack logic). It is easy to prove that if another pair of flags is substituted for flags 42 and 43, this simple routine would require at least one additional byte. By the way, when you key in this routine, remember to use the top row of keys for the postfix instructions, like GTO 01, LBL 05, FC? 02, to reduce the number of keystrokes.

Here is a routine that will toggle a flag (e.g., flag 02) on alternate executions of a section of a program (such as a loop). It requires that another flag (here, flag 01) be toggled every time.

```
FC? 01
FS?C 02
FS? 30
SF 02
FC?C 01
SF 01
```

The always-false filler, FS? 30 (Catalog flag), for the AND Logic used to toggle flag 02, alternatively may be replaced by an always-false stack conditional whenever the relative status of the stack is known.

Feedback

This new column contains reader feedback about articles or routines that appeared in previous issues. Though much of the information presented here is useful on its own, you will find that it is a good idea to have your library of KEY NOTES handy while reading this column.

Here's a notable comment from the land of windmills and wooden shoes. **W.L.C. Brunings** from Bilthoven, Holland, read the "In the Key of HP" column in the last issue with a critical eye.

(V6N1p7b) KEY NOTES gives many useful tips. But, one of the tips can be a pitfall—the shortest way to divide by 100 using: 1 [%]!

Look at this program fragment.

```
RCL 00
RCL 01
1
%
+
```

Store 100 in register 00 and store 12 in register 01. You wish to compute 100 + (12/100) = 100.12! But, with the above routine you get 12.12

Of course, you can change the routine to:

```
RCL 01
1
%
RCL 00
+
```

But, that is not always possible; i.e., when, instead of RCL 00, the X-register contains the result of a previous computation.

(You're very right, Mr. Brunings. It would not be wise to go through all your programs and indiscriminately change every 100; / to a 1; [%]—Ed.)

Palmdale, California, is the home of **Paul Burke**. Though Mr. Burke claims that he isn't interested in stocks, he is interested in making the "STOCK" routine that we printed on the back page of the last issue easier for us to use. Here's his letter.

(V6N1p16) (41) The routine for plotting the weekly stock chart, "STOCK," does not exploit the capabilities of the HP-41 and HP 82143A combination to their most useful extent. Specifically, I prefer program initiation to start at the front of the program, and then branch to subroutines only after the input data has been handled. The inputs for plotting itself could be entered from PROMPT's asking

(Continued)

for the specific item as known to the user, instead of the input name used by PRPLOT, which may not make too much sense in the context of the program. This method then uses PRPLOTP for the actual plotting.

I have found that this approach leads to faster use of any program, especially if you haven't used it for some time. The input prompts tailored to the particular requirements of the program prevent confusion as to what is meant by the input request. I am sure that only a few stock analysts could interpret "AXIS?" and "X MIN?" correctly if they did not use the program frequently enough to become familiar with the basic inputs required by PRPLOT, but they would need no refresher to understand "PURCHASE PRICE?" or "START DAY?." The listings enclosed indicate the way I would initialize the program, and request the necessary plotting inputs, in terms the user could identify without reference to a coding sheet.

I use this approach a lot. It takes up some room in the machine for "gingerbread," but the CV has room to spare for most programs. Also, having the program keep track of its own inputs prevents the occasional "oops!" when a prompt is misinterpreted due to ignorance of the proper program functioning internally, which should not have to be reviewed every time a program is run.

Anyway, here is an amended listing and output of what my "STOCK" routine would look like... if I was interested in stocks.

```
01↑LBL "BEGIN"
02 "START DATE,"
03 "↑MMDD.YYYY"
04 PROMPT
05 GTO "DATE"

92↑LBL "IN"
93 "STOCK"
94 ASTO 11
95 "MIN PRICE?"
96 PROMPT
97 STO 00
98 "MAX PRICE?"
99 PROMPT
100 STO 01
101 "PURCHASE PRICE?"
102 PROMPT
103 STO 04
104 "START DAY?"
105 PROMPT
106 STO 08
107 "TIME SPAN,DAYS?"
108 PROMPT
109 STO 09
110 "WEEKLY PTS."
111 AVIEW
112 7
113 STO 10
114 XROM "PRPLOT"
115 .END.
```

(This does make the Stock Plotting routine easier to use. Just add lines 92 to 115 to the end of the listing from V6N1p16, and add lines 01 to 05 to the beginning of that listing, then execute "BEGIN." The HP-41 will ask the necessary questions and then prompt you for the weekly data to be plotted. Be sure to set a minimum size of 18 and to have your printer switch set to MAN—Ed.)

If you have ever had to wait one second or longer while your HP-41 is searching for a local-ALPHA label, you will appreciate this next subject. **Joseph Senecal** of Roseville, California, contributed this elaboration of a previous suggestion.

(V5N3p13a) (41) I have found Julius Zechmeister's idea, on how to find local labels quicker, very useful. My most commonly used program has many local labels and some large routines. The search for a label often took as long as the routine itself. With the example in KEY NOTES as an inspiration, I developed a routine to do the same from any spot in a program for any label. Here is how I implemented it.

```
01 LBL 99          31 LBL I
02 STOP            :
03 RTN             38 XEQ 99
04 GTO 99          39 GTO I
05 LBL F           40 LBL A
06 X<>Y            : {INITIALIZATION}
07 GTO 99          51 "INPUT?"
08 LBL G           52 AVIEW
09 RDN             53 XEQ 99
10 GTO 99          : {COMPUTATION}
11 LBL A           75 GTO 99
12 GTO A           :
13 LBL B           899 END
14 GTO B
:
21 LBL H
: {SHORT ROUTINE}
30 GTO 99
```

This procedure allows for fast execution of X<>Y and RDN as well as faster local label location after any pause in execution. It allows a user to continue a routine or choose a new routine with a minimum of label search time. The key to this procedure is lines 01 to 04. Line 02 stops to show results or prompt for input. Line 03 returns if LBL 99 was XEQ'ed but not if GTO'ed. Line 04 stops extra R/S's from doing anything. Lines 05 through 10 do the X<>Y and RDN functions rather than letting the HP-41 search for them (always the longest search). Unfortunately, this prevents them from being used in response to a prompt (leaves the HP-41 at line 03). The following lines are branches to long routines followed by short routines. An XEQ 99 will stop, then return when R/S is pressed. A GTO 99 will stop and not continue.

(The concept used here is well worth studying. The slow response of many lengthy programs can be eliminated by taking advantage of the "jump-distance storage" characteristics of the local-label GTO and XEQ statements. We do have a suggestion, though. Rather than using LBL F for X<>Y and LBL G for RDN, try assigning these functions to their respective keys. Then, when you press one of these keys at a halt in the program, the local-label search is bypassed and the function X<>Y or RDN is performed immediately. Using the local-ALPHA labels F and G will both move the program pointer, as mentioned in the letter, and clear the subroutine return stack. Of course, if you can work around these two drawbacks, the method of using local-ALPHA labels has the advantage of not requiring you to make key-assignments—Ed.)

Next, we have a valuable suggestion for those with a card reader. This information came from **Tom Flegal** of Berkeley, California.

(V5N3p14) (41) Regarding **Vally Lambrecht's** routine, this and other routines with card reader prompts should have flag 21 clear before the AVIEW statement. Otherwise, the program may halt at the AVIEW and a normal program card-read will result. Incidently, RSUB (XROM 30,04) will not clear the same program that contains it. I find it will work correctly without an END.

(By the way, Vally Lambrechts' routine can be adapted to call subroutines from the HP-IL Digital Cassette Drive or to call routines from Extended Memory—Ed.)

Routines to solve the quadratic equation are a popular subject in KEY NOTES, and these routines keep getting shorter and shorter. The quadratic equation is $ax^2 + bx + c = 0$ and the roots of this equation are

$$x_1 = \frac{b + \sqrt{b^2 - 4ac}}{2a} \text{ and } x_2 = \frac{b - \sqrt{b^2 - 4ac}}{2a}$$

This routine was contributed by **Paul Baker** of Stillwater, Oklahoma. The program works entirely in the stack, it handles both real and complex roots, and the body of this routine—excluding the LBL and the END—is only 26 bytes. Can you beat that?

(V5N3p13c) (41) When I saw **Brent Tranberg's** quadratic solution routine, I became curious as to how it could be modified to solve for complex roots. Here's my simple suggestion:

```
01↑LBL "SLV"       12 X>Y?
02 X<> Z           13 SF 01
03 ST/ Z           14 -
04 /               15 ABS
05 -2              16 SQRT
06 /               17 ST- Z
07 ENTER↑          18 X<>Y
08 ENTER↑          19 FC? 01
09 X↑2             20 +
10 R↑              21 .END.
11 CF 01
```

The user executes the program just as before; however, flag 01 will remain activated in the display to indicate complex roots of the form '$x_{1,2} = U \pm iV$' where the real component U is in the X-register and the imaginary component V is in the Y-register.

Thank you for a most informative newsletter. I read KEY NOTES cover to cover!

(Thank you, Mr. Baker, for a fine example of efficient programming, and for the nice words about KEY NOTES—Ed.)

Here's an input from the northeast corner of the U.S., by a dedicated, prolific HP-41 *aficionado*, **Alan Zeichick**, who lives in Bangor, Maine.

(V6N1p13b) (41) I was reading the "Routines, Techniques, Tips, etc." column in the new KEY NOTES and saw HP's condensed version of Mr. Price's Hyperbolic Trig program. Well, I thought, the HP version is more aesthetically pleasing, but less functional; it really should have the inverse hyperbolic functions, too. So, I wrote them. I kept the same format as the "HYP" routine, assigning ASINH, ACOSH, and ATANH to local-ALPHA keys [C], [D], and [E], respectively—the most logical locations, since there are no local-ALPHAs in the same place as the inverse trigonometric functions. I've set up my routines to be physically placed after the "HYP" routine.

```
23♦LBL E        35 SF 01
24 1            36♦LBL C
25 X<>Y         37 ENTER↑
26 +            38 X↑2
27 1            39 1
28 LASTX        40 FS?C 01
29 -            41 CHS
30 /            42 +
31 SQRT         43 SQRT
32 LN           44 +
33 RTN          45 LN
34♦LBL D        46 RTN
```

I hope you find this worthy of attention. By the way, if ATANH is not needed, lines 23-33 can be deleted,the same as mentioned in your "HYP routine. I like to be consistent.

(Definitely worthy of attention, Mr. Zeichick. One thing you might want to try, though; replace LBL E, LBL D, and LBL C, with LBL 10, LBL 09, and LBL 08. Then, use the XEQ key as an "inverse" key. For ATANH press [XEQ] and then [TAN] (which the calculator understands as [XEQ] 10). Now, you have the hyperbolics and their inverses all "assigned" to the most logical keys—Ed.)

Now, let's travel to Sollentuna, Sweden, where **Gerhard Rombach** is keeping busy by reading KEY NOTES and programming his HP-41. Here is his contribution.

(V5N3p14c) (41) Fred Lipshultz's program "GEN2" was very clever—but you have to do the conversion the other way (converting a number in any base to a number in base 10.)

Here is my solution: Change the last line (line 42) in "GEN2" to STOP and continue as follows. Otherwise you have to store the base in register 04 and the number in register 01.

```
43 CLX          58 X>0?
44 STO 03       59 SF 01
45 STO 05       60 RCL 04
46♦LBL 03       61 RCL 03
47 RCL 01       62 Y↑X
48 X=0?         63 *
49 GTO 04       64 1
50 1 E-1        65 ST+ 03
51 *            66 X<>Y
52 INT          67 FS?C 01
53 STO 01       68 ST+ 05
54 LASTX        69 GTO 03
55 FRC          70♦LBL 04
56 1 E1         71 VIEW 05
57 *            72 .END.
```

New Club Formed in England

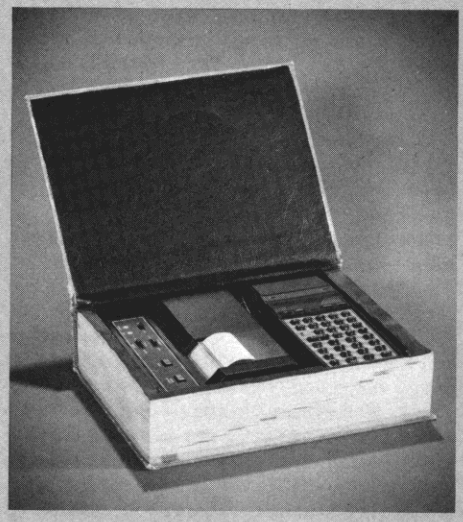
Elsewhere in KEY NOTES you will find a notice about a new calculator club in Germany. Just before we went to press, we received a letter about one more new club. Here's the letter.

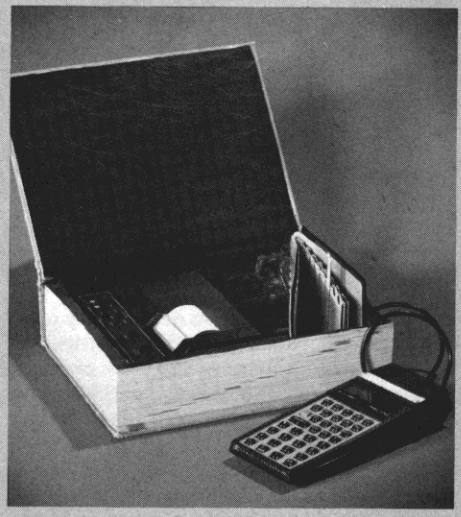
"I have recently formed a Users' Group for HP Personal Programmable Calculators in Britain and would be grateful if you could give the Group a mention in the next issue of KEY NOTES. The Group is called "PPC-GB" (Personal Programming Center-Great Britain) and is a chapter of PPC in the USA, but membership is open to *all* users of HP programmable calculators.

(Continued)

(Continued from page 1)

usable carrying case for it. So he carved-out the interior of a large book and lined it with vinyl. The HP-41 equipment is always together, very inconspicuous, and can be used without having to take it out of its *book*. He had these pictures taken, but never got around to sending them to you. So here they are!"





(Continued)

Back Issue and Subscription Information

Back issues of KEY NOTES are available back to V3N3, which introduced the HP-41. An index of these will be furnished on request. Available issues are:

V3N3 August 1979 (12 pages)
V3N4 November 1979 (12 pages)
V4N1 March 1980 (12 pages)
V4N2 Jun-Aug 1980 (12 pages)
V4N3 Sep-Dec 1980 (12 pages)
V5N1 Jan-Apr 1981 (16 pages)
V5N2 May-Aug 1981 (16 pages)
V5N3 Sep-Dec 1981 (16 pages)
V6N1 Jan-Feb 1982 (16 pages)

Prices for KEY NOTES back issues are as follows. All prices include first-class or air mail. Payment must accompany your order and must be a check or money order in U.S. dollars drawn on a U.S. bank. Or you may use your American Express, VISA, or MasterCard account; be sure to include your account number and card expiration date. Your order will be promptly mailed in an envelope.

NO. OF ISSUES	U.S., MEXICO CANADA	ALL OTHER COUNTRIES
1	$2.00	$3.50
2	$3.50	$5.00
3	$5.00	$6.50
4	$6.00	$8.00
5	$7.00	$9.00
6	$8.00	$10.00
7	$9.00	$11.00
8	$10.00	$12.00
9	$11.00	$13.00

KEY NOTES is published quarterly in February, May, August, and November. A one-year subscription in the U.S. and Canada is $5* a year. It is free (worldwide) in 1982 if you are a member of the Corvallis Users' Library ($20* U.S. and Canada; $35* elsewhere). Send your payment and complete name and address to the Corvallis address on back cover.

To get KEY NOTES in Europe, contact the UPLE (Geneva address on back cover). To get KEY NOTES elsewhere, contact your nearest HP Sales Office or send your name, address, and calculator serial number to the Corvallis Users' Library.

** U.S. dollars. See note at bottom of page 4.*

"I hope that the Group will compliment KEY NOTES, UPLE, and PPC by becoming a *local* forum for exchange of information, programs, and applications; for personal contact between members; for arranging meetings; and for the very important provision of a local "expert" to aid new users (or even a shoulder to cry upon when a program doesn't work).

"At present I am working on the prospect of a summer conference (half or one day meeting) and am trying to arrange with HP-UK and Dealers to provide demonstration equipment, speakers, and perhaps even a visit to HP here in the UK.

"I shall keep you informed of progress as the Group becomes more established and send you a copy of our newsletter when it is published. In the meanwhile, I would be grateful for any advice or help that you can offer—especially regarding user groups and producing newletters."

With many thanks, **David M. Burch**

We are happy to see our valued customers get together and form little groups, because it is a very good way to spread information, ideas, and comradeship. If you are interested in this new Group or want further information, contact Mr. Burch at the following address. It would be a nice gesture to include a self-addressed, stamped envelope.

David M. Burch/PPC-GB
Astage,
Rectory Lane,
WINDLESHAM, Surrey,
GU20 6BW England

(Note: PPC-GM is not sponsored, nor in any way officially sanctioned, by Hewlett-Packard.)

Columbia... *(Continued)*

the HP-41 helped them to keep on top of all of their daily "housekeeping" activities.

In addition to helping the crew organize its time, the second HP-41 computer was kept ready for flight-critical, deorbit-burn calculations. Once during each orbit around the Earth, the shuttle has an opportunity to land at one of six contingency locations. During a routine flight, Mission Control supplies the shuttle crew with deorbit-burn information. Should the shuttle encounter an emergency, however, the astronauts would rely on the HP-41 for these calculations.

Two other programs—one to help balance the *Columbia* prior to re-entry, and another to pin-point Earth observation sites—also are available to the crew and would be run on the HP-41's.

The HP-41's do not take the place of the shuttle's larger, general-purpose computers. However, they do complement the shuttle's larger systems and provide the crew with personal-computer convenience. Also, new and different HP-41 programs can be written between flights—quickly enough to keep up with many of the astronauts' changing computational needs.

We are very proud of the HP-41, and we are happy that NASA chose this handheld marvel for use on the space shuttle. Already the new Time Module is an asset to this mission and, in the future, the new HP-IL Module and the various HP-IL peripherals will surely prove their usefulness. We'll keep you informed as NASA makes more use of the HP-41 system.

Letters... *(Continued)*

Well...you can imagine how many HP-41 system "cases" we see every month, but we have to admit this one is, as they say, "far out." But we also have to admit that it's very clever and that it certainly does fulfill the security requirements that were intended. We congratulate **Keith Olson** and nominate him the "HP-41 Owner of the Quarter."

Mr Olson works for Sperry/Univac in the Bay Area and uses his HP-41 system extensively at the office to make his job easier. If he is as clever at his job as he is in contriving unique cases for his calculator, his employer should hold onto him! And, before we forget it, thank you, Mrs. Olson for the letter and photos.

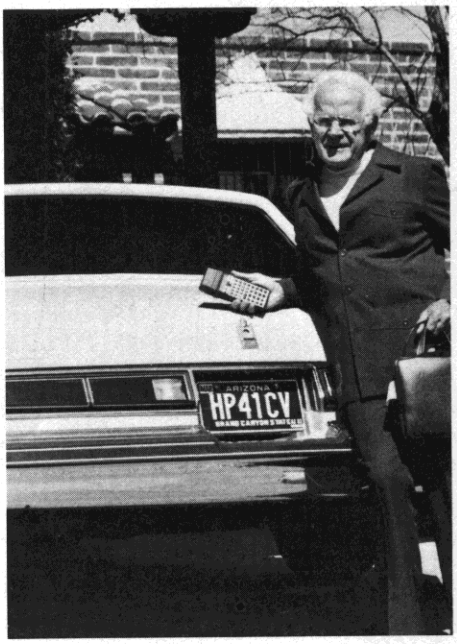
Now, here's another letter you will enjoy. How many people do you know who have "HP-41CV" on their automobile license plate? Well, *now* you know at least *one*! Here's his letter.

"Thought you might like to see my portable office! I am just finishing teaching a graduate course, "Hydrologic Analysis With Programmable Calculators," with Professor Lloyd W. Gay, at the University of Arizona. The course was accepted, like my license plate, with much enthusiasm.

"Being Assistant City Engineer, I could let the little wonders [*HP-41 system*] evaluate some of the city's turnoff problems. We plan to present the HP-41 programs to practitioners in a 3-day short-course this November."

The letter was signed, "Waiting for my new tape drive, Very truly yours, **Dr. Brian M. Reich.**" We thank you, Dr. Reich for the photograph and for taking the time to share your thoughts and your truly unique license plate with our readers. KEY NOTES is *always* interested in hearing about unique applications, situations, or what-have-you.

We might add that Dr. Reich is a Consulting Engineer and Hydrologist who lives in Tucson, Arizona.



HP KEY NOTES

March-May 1982 Vol. 6 No. 2

Programming and operating tips, answers to questions, and information about new programs and developments concerning Hewlett-Packard handheld computers. Published quarterly. See page 15. *Reader comments or contributions are welcomed. Please send them to one of the following addresses.*

Hewlett-Packard Company
Users' Library
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 USA

Hewlett-Packard SA
Users Program Library Europe
7, Rue du Bois-du-Lan
P.O. Box, CH-1217 Meyrin 2
Geneva-Switzerland

Hewlett-Packard Company
Corvallis Division
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 U.S.A.

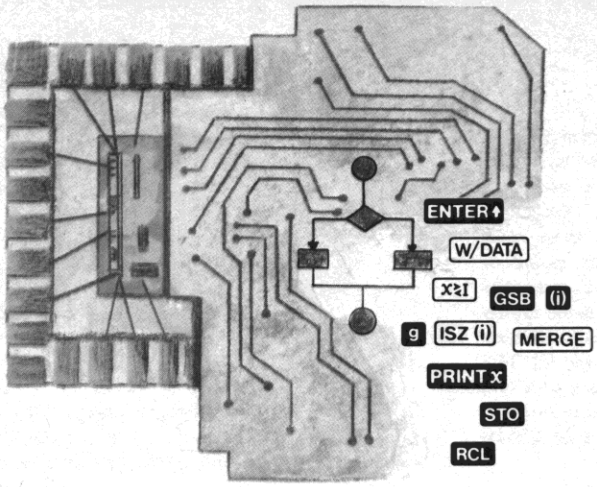
Address Correction Requested
Return Postage Guaranteed

FIRST-CLASS MAIL
U.S. POSTAGE
PAID
PERMIT NO. 814
PORTLAND, OR

775988SEAC487DEZEM 821110

MFCHAEL B DEZEARN
487 SEACLIFF DR
WEBSTER TX 77598

Printed in U.S.A.



Featuring, this issue:

HEWLETT PACKARD

June-August 1982 Vol. 6 No. 3
ISSN 0730-2037

HP Key Notes

Published Quarterly **By Subscription: $5/yr.** **$2.00**

NEW HP-75 Portable Computing System

On August 23, 1982, Hewlett-Packard announced a new portable, battery-operable personal computer, the HP-75C. (The C is for *Continuous Memory*.)

The new HP-75C features extensive BASIC language programmability as well as time and appointment functions. Its small size (10 by 5 by 1.25 inches; or 25.4 by 12.7 by 3.2cm); its ability to run on batteries; and its *Continuous Memory* make it a true portable computer for virtually any application anywhere in the world — on land, at sea, or in the air — or in outer space, for that matter.

The HP-75's uncluttered keyboard even permits touch-typing, yet it is extremely versatile. Every key can be redefined by the owner, and the redefined keys can be given new labels by snapping on keyboard overlays, thereby customizing the HP-75C for specific applications. A "hidden" numeric pad is built-in to facilitate lengthy numeric entries.

The HP-75C features 16k bytes of user memory (random-access memory, or RAM) built in. User memory can be increased with the plug-in 8k Memory Module to 24k bytes. Three software module plug-in ports in the HP-75C accept 8k- or 16k-byte read-only Memory (ROM) Modules. The ROM-based modules let the computer be customized for specific applications and also free all the user memory for data. With three 16k-byte plug-in modules, the 48k-byte built-in operating system and 24k bytes of user memory, the HP-75C's maximum memory is 120k bytes.

The ROM-based operating system means the computer is ready to use when turned on, and user memory is not diminished by operating system or plug-in software. The large operating system features 169 instructions, of which 147 are BASIC commands, statements, or functions.

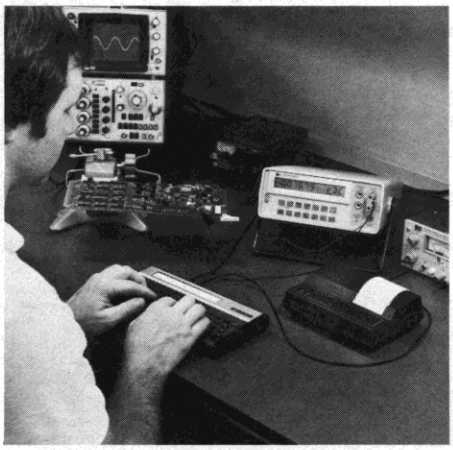
A 32-character, liquid-crystal display serves as a movable window on a 96-character line and features character descenders.

(Continued on page 2.)

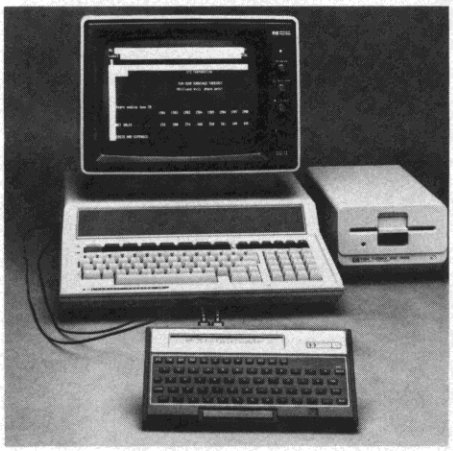
All prices in this newsletter are suggested retail prices excluding applicable state and local taxes—Continental U.S.A., Alaska, and Hawaii.

One type of off-line mass storage is integrated into the HP-75C. A hand-pulled magnetic-card reader, through which cards containing programs or data are pulled, reads or writes up to 1.3k bytes per card. A peripheral mass-storage device is available, but the card reader gives the HP-75C integrated mass-storage *even in the completely portable mode.*

Multiple files, which can interact, are an important part of the computer's capabilities. Program files, data files and appointment files can be named, saved in RAM, and programmed to interact with each other.



The HP-75C comes with built-in Hewlett-Packard Interface Loop (HP-IL). HP-IL lets battery-operable computers communicate with a range of peripherals and other computers, and is designed to be inexpensive and easy to use.

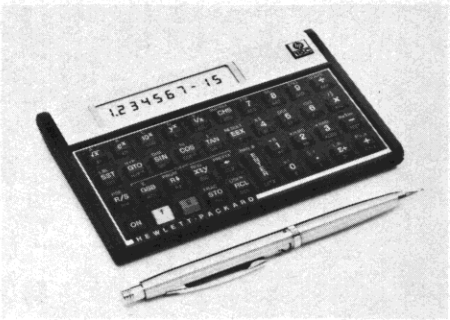


There is no end to what this marvelous creation can accomplish, with its complete range of peripherals; its ability to interface to instruments; its current and future software; and its total dedication to **portable, personal computing — anywhere, anytime.** No amount of words, here, could fully describe the total capabilities of the HP-75C. You simply have to see one and hold it in your hands. So check with your local HP Dealer or your HP Sales Office. It will be in great demand, so be patient.

By the way . . . the HP-75C weighs only 26 ounces (0.74 kilogram). But, if its *capabilities* could be "weighed," you would need to measure them in "tons!"

More Power at Your Fingertips

In June, at the Consumer Electronics Show in Chicago, Hewlett-Packard introduced two more significant contributions to the world of pocket keystroke programmable calculators. Designated the HP-15C and the HP-16C, these new calculators command an enormous range of computational power that, because of their petite size, is not readily apparent. In the same family of rugged, slim-line, liquid-crystal-display calculators initiated by the HP-11C and HP-12C — which were introduced last year (see V5N3p1) — these two new models solve an unprecedented number of engineering and computer-science problems.



The HP-15C Keystroke-Programmable Scientific Calculator is the most powerful "stand-alone" handheld calculator on the market today;

→

HP-86 Introduced

The modular design is the most visible feature of the new **HP-86 Personal Computer,** introduced by Hewlett-Packard on July 12. And, modularity means that you can build a flexible, low-priced system around the HP-86 to meet your changing needs.

The HP-86 consists of a powerful microprocessor and a keyboard in one package. The keyboard features a standard alphabetical pad and a numeric pad, plus, there are seven shiftable user keys for a total of fourteen user-definable functions. There are also six display editing keys to greatly simplify changes, corrections, additions, etc.

In addition to the operating system and the display RAM, the HP-86 has 64k bytes of user RAM. And, like the HP-87, this user RAM is expandable to half a million bytes with the addition of plug-in memory modules in the four ports in the back of the computer. These ports also let the user expand the capability of the HP-86. The HP 82900 CP/M* system, the HP 82950 modem, additional ROMs, as well as HP Memory Modules can be plugged into the ports.

In designing your HP-86 system, you can choose between a 9-inch and a 12-inch display monitor, and the HP-86 easily interfaces with the HP 82905B Dot-Matrix Impact Printer. A new disc drive, the HP 9130, with a 270k byte capacity, provides your system with plenty of mass storage. The new machine is also compatible with HP's personal-computer modem and a variety of printers and graphics plotters.

But, software is the key to the appeal of this new computer. All of the HP Series 80 BASIC software will run on the new HP-86 and, with the addition of HP's plug-in CP/M System, the CP/M operating system and associated software will run on the HP-86. New software for accounting, data-base management, tax planning, and word processing is available and this new software complements HP's existing graphics, electronic worksheets, and data-communication programs.

Two data-communications software solutions let the HP-86 communicate with other computers and with information services such as THE SOURCE** and Dow Jones News/Retrieval***. In fact, a free hour of connect time on the Dow Jones News/Retrieval and THE SOURCE is included with the purchase of the HP modem or HP's Data Communications Pac, giving the new user an introduction to information services.

This short article can only begin to touch on the wonders of this new computing marvel, the HP-86. See your local HP Dealer for a more complete picture of why the HP-86 is the solution to your computing needs; also for a very pleasant surprise: the HP-86 is the lowest priced personal computer ever offered by Hewlett-Packard.

**CP/M is a registered trademark of Digital Research, Inc.*

***THE SOURCE is a service mark of Source Telecomputing Corp., a subsidiary of The Readers Digest Association, Inc.*

****Dow Jones News/Retrieval is a registered trademark of Dow Jones & Company, Inc.*

Hewlett-Packard's Series 80 family of personal computers now includes (from left) the portable HP-85; the new, low-price HP-86; and the new HP-87XM, with its 128k bytes of built-in user memory. They are at your HP Dealer now; don't miss them!

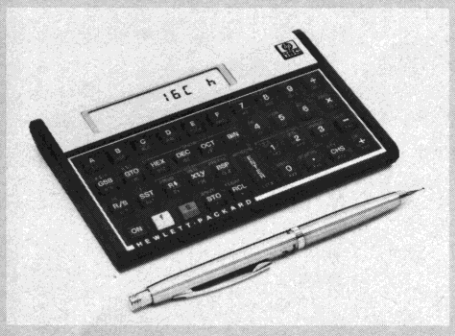
it even includes such novel, highpowered mathematical and computational tools as matrices and complex numbers, besides the already known "Solve" and "Integrate" functions. The HP-15C also offers 67 storage registers that can be allocated to a maximum of 448 program lines as you need them.

For the first time in a handheld pocket calculator, systems of linear equations are easy to handle. This marvelous "genius" can invert an 8 by 8 matrix in 85 seconds or solve a 7 by 7 system in 28 seconds. Or you can have five matrices — adding to a total of 64 elements (maximum) — stored in its memory. Furthermore, the operations involving matrices are executed by simply calling the descriptor of the matrices to the X- and Y- registers. Also, the descriptor can be any letter from A to E. But, best of all, the display provides alphanumeric annunciators to tell you, at any time, what matrix you are using, its dimensions, and what element you are in.

The ease of use of the Complex Functions — the Real and the Imaginary — will delight anyone who, because of job title or field of study (navigators, civil engineers, aviators, surveyors, students, teachers, etc.), has to express as a single mathematical entity, a magnitude and an angle, a distance and a direction, etc. To handle this function, the HP-15C has two parallel stacks, one for the real part, the other for the imaginary part of a complex number; you perform operations on the contents of both stacks *simultaneously*.

In particular, electrical engineers will love the *complex matrices* capability of the HP-15C. Just think: such complex, lengthy, error-prone computations as ladder networks and impedance calculations will be a pleasure, not a chore, with an HP-15C, because it solves such problems with consummate ease.

There are numerous matrix operations, including multiplication, "division" (multiplying times the inverse), addition, subtraction, inversion, determinants, norms, etc. In fact, the HP-15C has so many powerful features and functions — even a built-in random-number generator — you simply have to go to your HP Dealer and see for yourself.



The brochure for the HP-16C begins: "The most powerful calculator ever designed for computer science and digital electronics applications," so it's no mystery why we named it the "Computer Scientist." It has four number bases: binary, octal, decimal, and hexadecimal; four Boolean operators (AND, OR, EXCLUSIVE OR, NOT), extensive bit-manipulation capability, and variable word size (up to 64 bits), plus one's and two's complements. These features, combined with the HP-16C's programmability, give it the absolutely unique, highly desirable capability of emulating or "decoding" the single numerical instructions of any commercially available processor. As an example of this, the HP-16C Owner's Manual includes a program that will enable you to convert between IEEE standard floating-point binary format and the floating-point decimal format used in the HP-16C itself.

So, if you are a technical professional, here is the tool you once wished: "If only they would make one . . .!" It's here, and its programmability adds *more* power: you can now perform such tasks as simulating processor instructions. For example, the HP-16C can hold a maximum of 203 program lines, or 101 16-bit data registers. And you can use the handy insert/delete editing tools to debug your programs. Plus, you get 16 program labels, 4 levels of subroutines, 6 flags, and 8 conditional branching tests . . . and a lot more.

The HP-16C has virtually no competition. The only other model in this market offers no programmability, no binary base capability, no variable word size (maximum number of digits displayed: 8), and very limited bit-manipulation functions.

As we said about the HP-15C, seeing is believing. But you won't *really* believe what this extraordinary calculator can do — and how easily — until you visit your HP Dealer and see for yourself. And don't wait too long! They are in great demand . . .

NEW HP-41 Module Announced

Hewlett-Packard recently announced another module for your HP-41, the 00041-15042 Automatic Start and Duplication Module (AS & CD). It will be available at your local HP Dealer *after* September 1982.

Although this is a specialized module, it does provide two new important capabilities:

- Automatic start of program execution when the HP-41 is turned on, and
- Duplication of cassettes used in the HP 82161A Digital Cassette Drive.

The automatic start feature provides a means of writing "foolproof" HP-41 programs. With the Automatic Start Module installed, the HP-41 goes through a special sequence when it is turned on, and this sequence allows you to write programs that automatically set status, configure memory, access peripherals, or prompt the user. The automatic start feature looks for one of the following three conditions in an attempt to start program execution when you turn on the HP-41.

- A program labelled RECOVER in HP-41 main memory.
- A program labelled RECOVER in an HP-41 port with a lower number than that of the port containing the AS & CD Module.
- A "write-all" file named AUTOST on an HP-IL mass storage device.

If a program labelled RECOVER is found, the HP-41 will start running from RECOVER. If RECOVER is not found, the HP-41 will attempt to load the AUTOST file from a mass storage device in the Interface Loop.

Note that you do NOT have to have the HP 82160A HP-IL Module in place in order to use the automatic start feature. However, *it is necessary* to use it in order to use the cassette duplication feature of the AS & CD Module.

The cassette duplication (or mass copy) feature lets you easily duplicate programs and data. All the information on one HP 82161A Digital Cassette Drive can be copied to as many as 29 other cassettes (as you can see in the accompanying photo). Now, this can allow software suppliers to "manufacture" their own software, provide a means for individual users to "back-up" their cassettes, and enable volume end-users to copy and distribute data in cassette form.

If your question is, "Will it permit copying cassettes with private files?" the answer is no. If your question is, "Are there any shortcomings?" the answer is yes. But only this: This module has XROM numbers that are incompatible with the Machine Design Application Module. If you plug both of these modules into the same HP-41, operation will be unpredictable. Other than that, this is a really convenient addition to the HP-41 system, as you can see here in graphic detail!

Here are 30 HP 82161A Digital Cassette Drives placed in a continuous loop that is controlled by only one HP-41 and an HP-IL Module and an AS&CD Module. This setup was used in the Corvallis Users' Library to make 29 identical copies of a master cassette. Cassette duplication, as you know, is one of the Library's special services ($12 includes cassette). This is not only an impressive sight but also an impressive sound. The cassette drive has an intriguing sound — like no other device — and 30 of them whirring away attracts many curious ears!

IS YOUR ADDRESS BEING UPDATED?
(See page 7c.)

NEW HP-IL/GPIO Interface

The new HP 82165A HP-IL General Purpose Input/Output Interface now allows you to connect to HP-IL systems, other devices that contain parallel-interface ports. In other words, this interface connects the internal electronics of such a device to the loop, permitting communication between it and other HP-IL system components. Now, you will find it fairly easy to connect to *your* HP-IL system, devices produced by manufactures other than Hewlett-Packard.

For example, suppose your company uses a computer *not* made by Hewlett-Packard, and you own an HP-41/HP-IL system that you want to use as a portable data collector for more timely accounting of shipments and inventory. If your computer has parallel ports, you are in business. (If not, we'll gladly sell your boss a new HP-86 or HP-87!) With the new GPIO interface, your information could be transferred each hour from an HP-41, acting as a portable data collector, to your company's permanently installed computer. By this easy and efficient method, your company would be confident that their shipments and inventory summaries were no more than one hour old. Neat, eh?

The new HP 82165A HP-IL/GPIO Interface is at your local HP Dealer right now. Check it out. It's probably just the "tool" you need for *your* application.

NEW HP-41 Impact Printer

The new HP 82905B Printer, first shown at the Consumer Electronics Show (CES) early in July, is another HP-IL peripheral for the amazing HP-41 Handheld Computer.

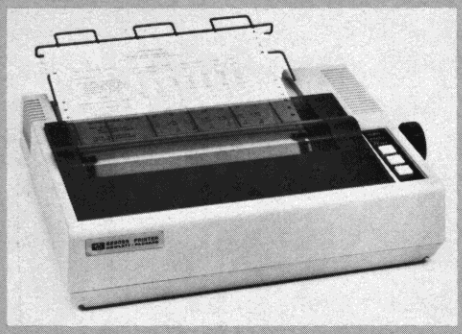
This new peripheral is a full-size, 80-column impact printer that produces permanent, high-quality copy and dot-mode graphics. Principal features are:

- 80-characters-per-second bidirectional operation.
- Variable column width (40 to 132 per 8.5-inch page).
- Five user-selectable print modes.
- Self diagnostics.
- Will accept single- or multi-part paper forms.

With user-developed software, individuals or companies now can assemble low-cost systems that not only compute solutions, but also print them quickly and efficiently on custom, multi-part forms. The end result could be anything — tax returns, paychecks, invoices, forecasts, inventory reports, etc. Other uses include fully documented, permanent, HP-41 program listings and automated test and measurement records.

The new printer also works equally well with HP-IL-equipped Series 80 Personal Computers, providing added versatility when these desk-top models are used in conjunction with HP hand-held computers.

By the time you read this, the new HP 82905B Printer should be at your local HP Dealer. Don't miss a demonstration!

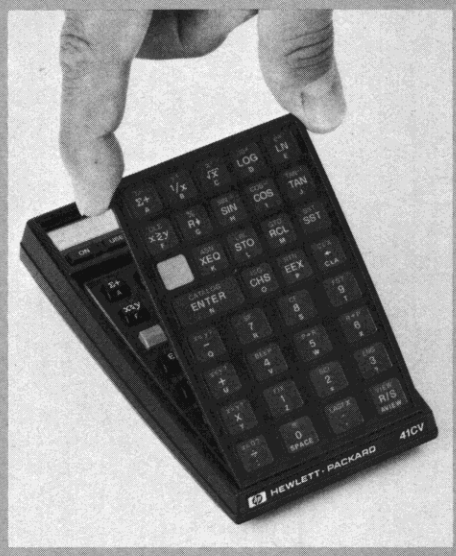


NEW HP-41 Touchpad

In V6N1p6a we published a photo and an article about a "Custom Keyboard" that was available to Corvallis Library Members, and it nearly started a stampede from all over the world.

So, to prove that we do listen to you, that "special-order" device is now an official HP-41 accessory. This new accessory is a membrane Touchpad with the standard HP-41 keyboard nomenclature. It snaps in place over your keyboard in the same manner as the usual overlays. It not only protects your keyboard from dust and spills but also enhances the aesthetics of your HP-41 Handheld Computer.

This new HP-41 Touchpad (HP 82200A) should be at your local HP Dealer by the time you read this. Call first for price and availability. And remember: this accessory is still available from the Corvallis Users' Library for 1 "point." DO NOT ORDER IT FROM THE FACTORY OR FROM GENEVA! (See V6N2p8a "A Change is in Order," or read *"Potpourri,"* on page 9.)



AND . . . /OR?

It all began with a letter that appeared in the V5N2 issue of KEY NOTES on page 12, column c. The letter was from **Claude Roeltgen** of Mondercange, Luxembourg, and it dealt with the subject of constructing the complex conditionals "OR" and "AND" on the HP-41. The information contained in this letter was valuable and very correct.

Everything remained quiet on the subject until V6N1p12c. There, we published a letter from our good friend, **Frank Wales,** of Glasgow (not Glaskow), Scotland. Unfortunately, because of a few of our typographical errors and our subsequently sleepy eyes, plus a misinterpretation by Mr. Wales, that letter can be a source of confusion. In his letter, Mr. Wales claims that there is an error in Mr. Roeltgen's solution for the "IF A AND B THEN instruction" conditional. But a careful inspection reveals no error. (See the second paragraph of Mr. Wales' letter and the second paragraph of Mr. Roeltgen's letter.) Nevertheless, being enamored of what we read and being slightly gullible, we proceeded to edit Mr. Wales' letter, and we even added some truth tables to the end to clarify his argument. This brings us to the final error in the article: in the first truth table, the sequence under "instruction performed?" should be 1,1,1,0.

Now for the good news . . . In the editorial comment at the end of Frank Wales' letter, we wrote a short routine to do the "IF A AND B THEN instruction" relationship, and we wrote that we were anxious to see a shorter version. (See KEY NOTES V6N1p12c.) Since that comment appeared, we have been swamped with letters from people all over the world. All of these letters included a shorter version of that routine (about two bytes shorter) and thus our anxieties have been satisfied.

```
01 A
02 inverse of B
03 FS? 30
04 instruction
```

The key to this routine is in line 03. This line can be replaced by any conditional that will always test false. A stack test (i.e., X=0) will work, if you know the condition of the stack, and it will save one more byte.

There is no possible way to individually credit everyone who sent in this routine. However, the first one we received was from **George Snyder** of Stoneham, Massachusetts. Thanks to everyone else who contributed. Please don't send any more; we already have enough to fill a book.

CORVALLIS LIBRARY NEWS

If you are not a Users' Library member and would like to know more about the Library and its many benefits, drop us a line (address on back cover). The fee is not only reasonable but also probably tax-deductible at income tax time. Also, you get coupons for 4 free programs, so how can you lose? Plus, Library members get KEY NOTES free, anywhere in the world. This is one of the few bargains left in this "inflated" world!

ORDERING PROGRAMS

HP-67/97 and HP-41 programs featured in KEY NOTES are available from both the Library in Corvallis and the Library in Geneva (except where stated otherwise). **Readers in Europe should order from Geneva (address on back cover) to get quicker service.** Readers elsewhere should order from Corvallis. Each program includes documentation and prerecorded magnetic cards; HP-41 programs include bar code.

Mail your order and a check or money order to the Corvallis or GENEVA address on the back cover of KEY NOTES. Don't forget to include your State or local taxes. Or, in the U.S., you can place your order by calling toll-free: 800-547-3400, except from Alaska and Hawaii. (In Oregon call 503-758-1010, NOT TOLL FREE.)

When ordering from outside the U.S., **attach your payment to your order.** Much time is wasted and orders are held up trying to match checks and orders that are sent in separately. Your payment can be in the form of an International Money order, a Foreign Draft, or the equivalent. *Any payment must be in U.S. dollars, drawn on a U.S. bank, otherwise it will be returned to you.* Another option for payment is to use such major credit cards as American Express, VISA, or MasterCard.

Orders are usually shipped within 2 working days after they are received in Corvallis. However, if you need a program yesterday, call us today at 503-757-2000, extension 3371. Although we can't get it to you yesterday, if you call before 12:00 noon, we'll get it in the mail today.

FUN AND GAMES . . . AND OTHER BOOKS

Amid all of the technical literature being developed by the Corvallis Users' Library is a new *HP-41 Games II* Solutions Book (00041-90443). It will be available by the time you read this, and the price is still $12.50* This collection of old and new best-sellers is destined to be just as popular as its predecessor, *HP-41 Games* Solutions book (00041-90099).

Another new book, available September 1, is the *HP-41 Structural Design* Solutions Book (00041-90441). Because this Solutions Book is larger (over 140 pages) and requires a cassette, it is priced at $30* (which includes a recorded cassette). But we are certain that structural engineers will call it a bargain when they see a section properties data base of 226 files, and a routine that allows creation of special section property files.

Also under development, and scheduled for an introduction later this year, is an *HP-41 Matrices and Complex Numbers* "Math" Solutions Book (00041-90442). We'll probably announce it in the next KEY NOTES (Nov. V6N4).

By the way, if there is a subject area that you would like to see covered by an HP-41 Solutions Book, write directly to the Corvallis Users' Library (address on back cover).

BEFORE YOU ORDER any of the Solutions Books mentioned above, make sure you first check with your local HP Dealer. Remember: All direct orders (whether to Corvallis or to the Corporate Parts Center — see page 9) include a $3.50* handling charge per order. These books should be on the dealer's shelves by the time you read this, no matter where you live.

WINNING PROGRAMS

The 1982 Users' Library Submittal Contest was announced in KEY NOTES (V6N1). This contest began in March and it will run through August. Each month, ten winning programs will be chosen on merit by our review panel. And, the authors of these winning programs get to choose a fabulous HP product as their prize! The top two winners every month will be awarded either an HP-IL Digital Cassette Drive or an HP-IL Thermal Printer/Plotter (both include HP-IL modules). The other eight winners will be awarded either a Time Module or an Extended Functions Module.

In the last issue, we featured the 10 winning programs for the month of March. What follows are the 10 winning programs for April and the 10 winning programs for May.

APRIL WINNERS

(41) Windmill Design #01719C (Price $10*)
This program evaluates a horizontal axis windmill at specified conditions of wind velocity and rotor RPM. Inputs to the program are the blade parameters of chord and twist angles at sequential values of radius R. The program evaluates performance at each input point and numerically integrates to determine output power and blade thrust forces.

A second mode of operation (Flag 1 set) allows the user to determine optimum blade twist angles, chord and RPM. A complete description of equations and program flow diagrams are given in the documentation. *Required accessories: 2 Memory Modules.* (449 lines, 939 bytes, 45 pages)
Author: **Emery Gasparek**
Camillus, New York

(41) Dose #01731C (Price $10*)
The ICRP multi-compartment respiratory tract model is used to predict doses to organs from radionuclide depositions in the nasopharyngeal, tracheobronchial, and pulmonary regions. For gastrointestinal tract doses, the four-segment model is used. Ingestion is included for the sake of completeness. Infant, child, and adult doses are displayed for arbitrary uptake and exposure time episodes. The program is not designed to address noble gases. *Required accessories: Three Memory Modules.* (668 lines, 1199 bytes, 29 pages)
Author: **John Ferman**
Minneapolis, Minnesota

(67/97) Propulsion Gas Turbine Annual Fuel Consumption for a Ship #04801D (Price $6*)
Employing an empirical formula to estimate specific fuel consumption for gas turbine main engines, this program calculates the annual propulsion fuel consumption for a ship operating in an assumed (user specified) operating profile. Required input includes: ship speed versus power relationship, assumed operating profile, shaft RPM as a function of ship speed, and reduction gear ratio. (224 lines, 17 pages)
Author: **James Baskerville**
McLean, Virginia

(41) Geodesic Arc Length, Azimuth #01660C (Price $6*)
Input the latitude and longitude of two points and this program will calculate the distance between those points, the azimuth, and the back azimuth, based on the reference ellipsoid of your choice. Uses the Andoyer-Lambert formulas with a second order flattening term as derived by Thomas. *Required accessories: One Memory Module.* (308 lines, 357 bytes, 12 pages)
Author: **Brian Iwatake**
Aiea, Hawaii

(41) Superelevated Roadway Design with Elevations #01768C (Price $10*)
Calculates roadway centerline and shoulder elevations for superelevated roadway curves rotated about the centerline. The program can be used in conjunction with the HP Surveying Module; however, the required subroutines are supplied with this program. In addition to solving for various combinations of horizontal and vertical alignments, the program provides checks of several design parameters. *Required accessories: 3 Memory Modules (2 Memory Modules if used with the HP Surveying Module).* (684 lines, 1265 bytes, 25 pages)
Author: **Greg Laragan**
Lewiston, Idaho

(67/97) Saturated Steam Flow Pressure Drop #04788D (Price $6*)
This program calculates pressure drop in lines for turbulent saturated steam flow by dividing the line into sections to account for the changing specific volume of the steam. The user provides the inside pipe diameter, saturated steam mass flow rate, equivalent pipe length, and inlet steam pressure. No physical property data input is required. The program includes routines to calculate the specific volume of saturated steam and saturation temperature as a function of the absolute pressure. (204 lines, 10 pages)
Author: **Randall Soo-Hoo**
San Francisco, California

(41) Symbol Selection And/Or Permutation #01753C (Price $6*)
This program will systematically select and/or permute a non-null subset of a symbolic set of up to ten members. A SCAN/POINTER algorithm is utilized for both selection and permutation. The definition of "selection" used in this program is "a combination in the natural order of the set." The program checks all values for range and validity. *Required accessories: One Memory Module.* (239 lines, 520 bytes, 18 pages)
Author: **Randy Cooper**
Cypress, Texas

(Continued)

(41) Cylinder Radar Cross Section #01796C (Prize $6*)

This program calculates the bistatic radar cross section of a totally reflecting, nonabsorbing circular cylinder for a plane wave incident perpendicular to the cylinder axis. The polarization can be at any specified angle to the cylinder axis. Restrictions on the cylinder size and radiation wavelength are minimal. *Required accessories: One Memory Module.* (379 lines, 607 bytes, 18 pages)

Author: **Glenn Stumpff II**
Dayton, Ohio

(41) Cantilever/Suspended Span Beams #01741C (Price $10*)

This program calculates all positive and negative moments, joint shears, and reactions for lines of cantilever/suspended beams. The user chooses the live-load variation factor, spacing of supports (columns) and of point-load(s). One to five point loads can be located between supports. Output is complete and self-contained. The program is useful for service load or limit states design. Use S.I. or Imperial units. *Required accessories: Printer, Quad Memory Module.* (1119 lines, 1659 bytes, 20 pages)

Author: **Frank Anrep**
Toronto, Canada

(41) Interactive Flight Information Manager #01804C (Price $12*)

This program is used for preflight planning and enroute flight plan verification, time and distance estimation and recall, fuel management and data storage. Features include: extensive alphanumeric prompting, functional keyboard, clock function, extensive data storage/recall capability, look ahead/look back for distances, times, and fuel reserves, plus automatic printing with the HP 82162A or 82143A Printer. *Required accessories: Four Memory Modules.* (807 lines, 1719 bytes, 25 pages)

Authors: **John Shaughnessy, Ph.D.**
Newport News, Virginia
Jeffery Farmer
Morganfield, Kentucky

(Mr. Gasparek and Mr. Ferman are the "big winners" for April, and they chose a new Digital Cassette Drive or a new Printer. Congratulations to all of you. It is especially nice to see that those "old" HP-67/97 machines are still solving esoteric problems — Ed.)

MAY WINNERS

(41) Fluid Flow Through Square-Edge Orfice #01787C (Price $22*)

This program computes the flow of a fluid (from a gas to a viscous liquid) through an orifice in a pipe. Required data are the fluid viscosity and density and, if a gas, the pressure, molecular weight, and heat capacity ratio. The size of the pipe and orifice must be known, as well as the pressure drop across the orifice. Results are valid for turbulent and laminar flows, and for diameter ratios (orifice to pipe) as high as 0.8. *Required accessories: Quad Memory Module, Extended Functions/Memory Module, Extended Memory Module. Card Reader and Printer will be helpful.* ("ORFC" consists of a main program and 17 auxiliary programs all loaded on 37 tracks, 19 cards; 29 pages of documentation)

Author: **Norman Samish**
Houston, Texas

(41) Pipe-Sizing for Compressible Flow #01859C (Price $18*)

This program is designed to solve a broad range of pipe-sizing problems for compressible flow. The program assumes that flow is isothermal, and that either the upstream pressure or the downstream pressure is known. The Mach number can be found at the inlet and, more importantly, at the outlet, where sonic velocity may limit the flow. The documentation of this program is complete and explicit. *Required accessories: Quad Memory Module, Card Reader or Wand, Extended Functions/Memory Module.* (778 lines, 1447 bytes plus data, 56 pages)

Author: **Jeffrey Huang**
Corvallis, Oregon

(41) Financial Institutions Analysis #01857C (Price $10*)

With the data of averaged balances and Profit-and-Loss Statements of two or more (infinite) periods, this program analyzes and/or simulates with 50 outputs for each two periods the key variables determining the performance of a financial institution. These key variables include: return, net income, losses, debt, and others. *Required accessories: Four Memory Modules, Printer optional.* (535 lines, 1351 bytes, 22 pages)

Author: **Raul E. Barragan-N**
Bogota, Colombia

(41) Electric Rate Analysis #01858C (Price $8*)

This program is used to compare the monthly charges for electric energy usage under two different rate schedules. Each rate schedule can have up to five energy blocks, a monthly customer charge, and a monthly minimum charge. The program will calculate the total monthly charge under both rates for a given kilowatthour (kWh) usage, and the percent increase (decrease) of the second rate over the first rate. Output to the printer consists of printing each rate schedule, a table heading, and a comparison of charges at specified kWh usages. Either a single kWh value can be entered or the user can define a range of kWh values to be compared. *Required accessories: 3 Memory Modules, Printer.* (669 lines, 1513 bytes, 18 pages)

Author: **John Brown**
Boise, Idaho

(41) The Wire Calculator #01777C (Price $6*)

This program simplifies wire sizing problems associated with aerospace electrical subsystem design and implementation. It solves for wire gauge, length, current, voltage drop, temperature, resistance, and provides for either English or metric input/output. The basis is orbiter specification MB0150-048 (MIL-W-81381A) for nickel-coated wire in sizes **#26 thru #0 gauge.** *Required accessories: 2 Memory Modules.* (325 lines, 792 bytes, 17 pages)

Author: **James Fuhrman**
Anaheim, California

** U.S. dollars. Orders from anywhere outside the U.S. must include a negotiable check (or money order), in U.S. dollars, drawn on a U.S. bank. All orders from anywhere outside the U.S. and Canada must include an additional 10 percent fee for special handling and air mail postage. (For example, an order for two programs = $6 × 2 = $12 + $1.20 = $13.20 total.) If you live in Europe, you should order KEY NOTES Programs directly from the Geneva UPLE, but make certain you make payment as required by Users' Program Library Europe; the above $6 fee is good only for orders to the Corvallis Library.*

(41) Shortcut Multicomponent Distillation Design #01778C (Price $6*)

If the compositions of the feed, overhead, and bottoms streams are known and the relative volatilities are known, this program will calculate the minimum reflux ratio and the minimum number of stages. Then, it will calculate the number of stages or reflux ratio given one of the two. The program uses Underwood and Fenske equations with the Gilliland correlation. *Required accessories: 2 Memory Modules.* (326 lines, 572 bytes, 21 pages)

Author: **Robert Green**
Starkville, Missouri

(41) Oxygen Deficit in a Polluted Stream With Time #01806C (Price $10*)

This program determines the oxygen deficit in a polluted stream with time. Oxygen deficit will be printed out in 1/10-day intervals until it is at a maximum. Input data may be entered either manually or by a data card. The computed maximum oxygen deficit and the original input data can be saved on a magnetic card. The calculation of the variation with time of the dissolved oxygen in a polluted stream is important in water-resources engineering. Along with the program is a brief discussion of this topic and a summary of the methods used to calculate the oxygen deficit in a polluted stream. *Required accessories: 3 Memory Modules, Printer.* (367 lines, 1039 bytes, 21 pages)

Author: **Choo-Heun Jonas Lee**
Seoul, Korea

(41) Eigenvalues/Vectors for Nth Order Systems #01814C (Price $10*)

This program will compute eigenvalues/vectors for systems up to and including 7th order. It uses the standard format BX = [gamma] AX, where A is diagonal and B is symmetric. Options include finding the square roots of the eigenvalues and normalizing the eigenvectors. SIZE check, data correction, and data review subroutines are included. A plotting program for mode shapes is also included. *Required accessories: Minimum of 3 Memory Modules; Quad Memory Module required for 6th and 7th order systems, Printer is optional; Printer and Card Reader necessary to merge and run plotting routine.* (775 lines, 1372 bytes, 24 pages)

Author: **Wm. Brewster Davis**
La Jolla, California

(67/97) Design of Crossflow Finned Tubular Heat Exchangers #04809D (Price $8*)

This program calculates the principal dimensions of a crossflow finned tubular heat exchanger. Temperatures, pressures, and heat transfer requirements for the specific application are required input, as well as geometric friction and heat transfer data for the particular finned surface being considered. Derivation of technique and a practical example are provided. (224 lines, 45 pages)

Author: **James Baskerville**
McLean, Virginia

(67/97) Performance Ranking of Finned Tubular Heat Exchanger Surfaces #04810D (Price $16*)

The objective of this program is to determine, from a large field of candidate surfaces and geometric arrangements, which surface is optimum for a particular heat exchanger application. The traditional method of presenting friction factor and heat transfer data for finned

tubes is transformed to permit performance comparisons. A design example is provided. (448 lines, 126 pages)

Author: **James Baskerville**
McLean, Virginia

(The "big winners" for May are Mr. Samish and Mr. Huang, but our congratulations go to all ten of you. This Contest sure is bringing some excellent programs into the Library — Ed.)

HP-41 Software Update

If you own any of the HP-41 Application Pacs listed below, be sure to carefully read this article.

HP-41 FINANCIAL DECISIONS PAC

The *Financial Decisions Pac* has had two revisions since it was originally introduced. If you have an early copy of this pac, you have read about these corrections either in KEY NOTES or via the addendum card provided with your pac. In general, the errors can be easily corrected with simple keystrokes. For your convenience, the programs involved and the corrections to them are reprinted below.

All errors have been corrected and these corrections have been incorporated in the finished product currently available. If you execute these programs regularly and have found your *Financial Decisions Pac* inconvenient to use because of the extra keystrokes required, you may exchange your current module for the most recent version. Use the following procedure to determine which version you have.

Turn off your HP-41 and remove all peripherals. You do not need to remove Memory Modules. Insert your "Finance" module. Press [ ] [CATALOG] 2. The first entry in the catalog specifies the module name (FINANCE) followed by the version code (1B, 1C, 1D). If the revision code on your module is 1B, there are two errors in your module, and they are explained below under **Financial Decisions Pac 1B.** If the revision code is 1C, there are three errors in your module, and they also are explained below, but under **Financial Decisions Pac 1C.** Revision code 1D is the latest (correct) version of the module.

Financial Decisions Pac 1B

1. In the FINANCE 1B module, the prompts for the "Modified Internal Rate of Return (MIRR)" program were inadvertently reversed. When SAFE RATE = ? is displayed, key in the risk rate. Likewise, when RISK RATE = ? is displayed, key in the safe rate. In the example on page 25 of the applications book, the risk rate is 8% and the safe rate is 10%.

2. The "Bonds" program does not correctly calculate the yield of a short-term bond. When you wish to determine the yield of a bond held for one full coupon period or less, you must do the following:

a. Store either the call price or 100 in register 19.

b. CF 22

c. Press [E] to calculate the yield.

Financial Decisions Pac 1C

1. The 360 calendar mode in DAYS does not calculate the correct number of days between dates when the first date occurs on the 31st day of the month and the 2nd date occurs any day except the 31st. The result is 1 day less than the correct answer.

2. After initialization of the "Compound Interest" program ([XEQ] MONEY) or after clearing the financial registers ([ ], E), if I is the variable solved for when N, PV, and FV are known (PMT = O), you cannot solve for another variable until a value has been stored.

3. When I = O in the "Compound Interest" program, you cannot solve for N, PV, PMT, or FV (gives DATA ERROR or an incorrect result).

To receive your replacement module, send your current *Financial Decisions Pac* Module **(versions 1B or 1C only)** to your nearest HP Service Center along with a request for a ROM update. DO NOT RETURN ANY 1D modules! A list of Service Centers is printed on the Service Card you received with your HP-41. If you can't determine your nearest Service Center, ask your local HP Dealer for the address.

HP-41 STRESS ANALYSIS AND STRUCTURAL ANALYSIS PACS

The early copies of the *HP-41 Stress Analysis Pac and Structural Analysis Pac* have an error in the "Simply Supported Continuous Beams" program (version 1A*). This error was originally reported in the March 1980 issue of KEY NOTES (V4N1), along with the keystrokes that could be used to correctly execute the program. Currently available copies of these pacs are correct. An Addendum Card that contains a detailed description of any errors, plus ways in which to "get around them," may be obtained by writing to:

Hewlett-Packard Co.
Customer Support
1000 N.E. Circle Blvd.
Corvallis, OR 97330 USA

In your request, be sure to include the name of your pac and the version of your module.

While the early copies of these two pacs are still very usable, those of you who regularly use this program may want to exchange your current module for a corrected version. To receive your replacement module, send your incorrect version 1A module to your nearest HP Service Center along with a request for a ROM update. A list of Service Centers is printed on the Service Card you received with your HP-41. If you can't determine your nearest Service Center, ask your local HP Dealer or HP Sales Office for the address.

**You can distinguish the "1A" version of the module (which contains the error) from the "1B" version (which is correct), with the following procedure.*

Turn off the HP-41C and remove all peripherals. You do not need to remove Memory Modules. Insert the "Stress" or the "Structures" module. Press [ ] [CATALOG] 2. The first entry in the catalog specifies the module names (STRESS or STRCTA), followed by the revision code (1A, 1B, 1C, . . . etc.). If the revision code on your module is 1A, follow the above procedure. For later revisions, ignore the above procedure.

WARNING

Do not send any application modules to the United States from a country outside the U.S. Your module could get tangled-up in Customs and you might lose it or cause an enormous delay.

KEY NOTES Correction

This letter from **George Killian,** of Stamford, Connecticut, explains a small problem that appeared in the last issue of KEY NOTES (V6N2P13a).

Don Thayer gave a routine for computing Julian Day Numbers and, as an example, he stated that the JD# for July 4, 1980 was 2,444,425. This answer is both right and wrong! The JD# is the number of successive days starting at January 1, 4713 B.C. (on the Julian calendar). However, by convention, the JD# changes at noon, not at midnight. This is so all astronomical observations of a given night will occur on a given Julian Day Number. Thus, 2,444,425 is the JD# for July 4, 1980 before noon, while 2,444,426 is the JD# for the same date after noon and until noon of July 5, 1980.

Software Corrections

Chris Verhoef, of Velsen, Netherlands, recently wrote to remind us that the 14 X 14 Math Pac matrix dimension was based on the HP-41C with three Memory Modules. Now, with the Quad RAM or the HP-41CV, a 16 X 16 matrix is possible.

We also have a correction to the Users' Library *High-Level Math* Solutions Book. The problem is located in the program "4 X 4 Matrix Operations" (the last program in the book). The indices of the output inverted matrix do not increment properly. To make the program run correctly, change line 427 to XEQ 08, omit line 426, and add FC? 09 after line 41.

Is Your Address Correct?

Rarely does a day go by that doesn't bring a big stack of KEY NOTES marked: "Not Deliverable — Address Unknown." Not only is it frustrating for us to see you not receive something you enjoy, it is also quite expensive for both HP and you. Every time your copy of KEY NOTES **doesn't** reach you, we have to pay *at least* $0.25 to get it back, so we can *know* that your address is incorrect. Then it costs about $1.00 to get it *back* to you. Needless to say, this is a total waste of funds and resources that can be better used to bring you more information, more pages, or more issues per year.

But *we* cannot keep *your* address current unless *you* inform us when *you* move. And because it is growing increasingly expensive to pay for "returned, undeliverable" KEY NOTES, we are faced with increasing costs that eventually must be passed on to you in order to keep KEY NOTES alive and growing bigger and better — and more often. Since you often mention in your letters that you'd like to receive KEY NOTES more often, here is your golden opportunity to bring that dream a lot closer. If you faithfully report your address changes, we can save a lot of totally wasted money, which in turn can be channelled into providing more benefits to you.

It's up to you. Less needless costs mean more helpful benefits. If everyone conscientiously helps to eliminate this expense, everyone will reap the benefits.

Entrepreneurs, Where Are You?

If software is not your *forte* or if you have not learned to program like a *virtuoso* or *virtuosa* and cannot create, literally, *glissandos* of creative code and cause the HP-41 to pour out *crescendos* of data or solutions, you are not playing our tune of the moment and need read no further.

If, however, you compose in a higher key or register, and if you have some *really good* programs that you think can make it into the **HP PLUS** program, we're finally in tune, and we can play some beautiful music together. (Don't laugh; it got *your* attention . . .)

Okay, what does the HP+ program do for you? First, *your* software is listed in a special section of the *Catalog* that is mailed to all members of the Users' Library and distributed through the HP Dealers; Second, we give *you* access to over 1500 HP Dealer outlets. That, my dear "composers," is distribution power that can put money in your pocket, a very welcome thought in these "uneconomic" times.

What does HP require from you? Easy — just good quality software. We want to hear those HP-41's "singing" all over the world. But we also want you to be ready to commit resources to selling and supporting your programs.

In what areas do we see opportunities? How about data management (with the cassette drive and extended memory)? How about time management, agriculture, remote data collection, medical applications, dental applications, navigation, sales? The list could go on and on, but it doesn't need to. What makes the *entrepreneur* special is that he or she goes into areas that no one ever dreamed of.

What do *you* see as the new areas? Are you playing our song? Can you "tickle that keyboard" and make the HP-41 "sing"? For more information on how **HP PLUS** can work for you, write:

HEWLETT-PACKARD CO.
1000 N.E. Circle Blvd.
Dept. 5360K
Corvallis, OR 97330 USA

The X<>F Function

Michel Walch of Ville D'Avray, France (though not in our atlas, we hear it's very close to Paris), requested that we elaborate a bit on the explanation of the X<>F function that appears on page 13 of the *HP-82180A Extended Functions/Memory Module Owner's Manual.* The following hypothetical example should clarify the general application of X<>F.

Joan Jones is creating an extensive program that will require the use of 20 flags. She is planning to submit this program to the Users' Library, and she knows that by using the peripheral flags (flags 11-20) as general purpose flags, she would be limiting the program's usefulness. Also, Joan's printer uses flag 12 and 13, her Video Interface uses flags 15 and 16, and she occasionally uses flag 17 to suppress the carriage return output to her HP-IL loop.

However, having recently purchased an Extended Functions/Memory Module (X functions), Joan is elated by the fact that the X<>F function is going to make the manipulation of these 20 flags very easy.

Joan's first move is to establish three blocks of eight flags each. She labels each flag with two numbers, as follows:

Block 0	Block 1	Block 2
(0,0)	(1,0)	(2,0)
(0,1)	(1,1)	(2,1)
(0,2)	(1,2)	(2,2)
(0,3)	(1,3)	(2,3)
(0,4)	(1,4)	(2,4)
(0,5)	(1,5)	(2,5)
(0,6)	(1,6)	(2,6)
(0,7)	(1,7)	(2,7)

This gives her a total of 24 flags, of which, 4 will not be used.

Initially, in her program, Joan would like flags (0,3), (0,5), and (0,6) to be set and all of the rest of the flags to be clear. This means, according to page 12 of the manual, that the initial numeric value for block 0 will be 8 + 32 + 64 = 104. And, since all the flags in block 1 and block 2 are clear, the initial numeric value for block 1 and block 2 is zero.

Joan decides that register 00 will be used to store the numeric value for block 0 throughout the program. Likewise, register 01 will contain the numeric value of block 1, and register 02 will contain the numeric value of block 2. Initially, in the program, 104 is stored in register 00 and 0 is stored in both registers 01 and 02.

Early in the program, Joan wishes to test and perhaps change the status of a few of the flags in block 0. She uses the program lines RCL 00; X<>F. This places block 0 into the first 8 flags of her HP-41. Now, if she wishes to set flag (0,2) and clear flag (0,6) she uses the program lines SF 02; CF 06.

A little later in the program, Joan wishes to save block 0 and manipulate the flags in block 1. She uses the program lines RCL 01; X<>F; STO 00. Now, Joan's HP-41 flags 00 to 07 correspond to block 1 flags (1,0) to (1,7), and register 00 contains the updated numeric value of block 0. Joan's program can now easily manipulate flags in block 1. To set flags (1,0), (1,1), (1,5), and (1,7), she uses SF 00; SF 01; SF 05; SF 07. To set flag (0,2) if flag (1,4) is set, she uses CF 09; FS? 04; SF 09; RCL 00; X<>F; FS?C 09; SF 02; X<>F; STO 00.

Later in the program, when Joan wishes to access block 2, she uses RCL 02; X<>F; STO 01. This places block 2 in HP-41 flags 00 to 07 and updates the numeric value of block 01.

You can see from this simple example that the X<>F function provides the HP-41 user with an almost unlimited number of general-purpose flags. By sacrificing a few registers for blocks of flags (at 8 per) and a few bytes for moving these blocks of flags in and out of HP-41 flags 00 to 07, the possibilities are . . .

```
LBL 00
GTO 00
```

or, in other "words," endless.

More About HP-IL Control Operations

In the last issue (V6N2P10b) we reviewed the use of general HP-IL I/O functions. This is a continuation of that discussion.

FINDID

The FINDID function enables the HP-41 to locate certain types of devices in the interface loop. Using this feature, a programmer can design software to eliminate the user's concern about the order of connecting devices in the loop.

For example: A programmer is developing software to be used by nontechnical personnel. An HP 82165A GPIO Converter, that is connected to a parallel printer, has been implemented for page-formatted output. The programmer wants the user to be able to connect the loop in any legal configuration, without regard to device location. The FINDID command will allow this to be done by simply using this program sequence:

```
01 "HP82165A"
02 FINDID
03 SELECT
```

The first line places the I.D. of the desired device in the ALPHA register. The second line locates the specified device and returns the loop location to the X-register. The third line SELECT's the device at the location specified by the value in X. After these three short steps, the programmer may begin to accumulate data in the printer buffer or print ALPHA data.

The capability of the FINDID function to determine a device location by its device I.D. is termed "Device Identity" in official HP-IL parlance. Many devices on HP-IL, such as the HP 82161A Digital Cassette Drive and the HP 82162A Thermal Printer, respond to another type of code called "Accessory Identity" but do not respond to Device I.D. These devices are automatically found for you, without additional programming. The devices available today that do require you to SELECT them prior to communication are: the HP 82165A GPIO Converter, the HP 82166A HP-IL Converter, the HP 3468A Digital Multimeter, and the HP 82905B HP-IL 80-Column Impact Printer.

The search that is initiated by the FINDID function begins at the currently selected device. For example, if device 2 is the currently selected device and there are two HP 82165A GPIO Converters in the loop, one at location #1 and one at location #4, then a FINDID in search of a GPIO converter would return a 4 to the X-register.

INSTAT

This command allows the programmer to query the selected device as to its current status. From previous discussions we know how to find certain devices, and to SELECT any device. Each device has a status word (or words) that consists of one or more 8-bit bytes that may be accessed by the HP-41.

For example: Assume that the HP-41 has just displayed the message "TRANSMIT ERROR" after attemtping a "READP." How can the exact problem be determined?

First, count around the loop from the HP-41 in the direction of the small (male) connector until you locate the cassette drive in question. Key-in, to the X-register, the location of the drive, then key: SELECT; INSTAT. This sequence selects the device specified in the X-register and brings the first, and only the first, status byte from that device into both the X-register and flags 0-7. Now, the X-register has the decimal representation of the first binary byte of status, and flags 0 through 7 are set or clear to the actual binary status of this byte. If, for instance, the X-register had the number 24, flags 3 and 4 would be set. An analysis of the flags shows:

128	64	32	16	8	4	2	1	Value
7	6	5	4	3	2	1	0	Flag
0	0	0	1	1	0	0	0	Status

Flag 3 = Value of 8

Flag 4 = Value of 16

Total of 24

Page 11 of the *HP 82161A Digital Cassette Drive Owner's Manual,* tells us that a status of 24 indicates a time-out error; no data detected on tape. That is, you may now conclude that your tape, or the data on the tape, is defective.

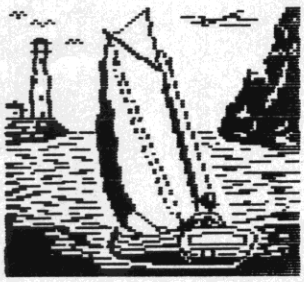
The duplication of the status message in flags 0-7 is an aid to the programmer. It provides an easy test for specific device status changes.

We Get Letters . . .

Someone recently asked me (your *Ed.*), "How many letters do you get? Do you really read *all* of them?" The answer to the first question is: I never counted them, but in 8 years, the total must be *very much more* than 10,000. And to the second question, the answer is: yes, every one of them — unless they are merely about a missing issue, just an order, or some such piece of clerical business. And I liked the wit, humor, and expertise evident in *this* one.

"Here is my little contribution to 'character building.' I thought possibly you folks at the Library and the readers of KEY NOTES might find it of some interest. I think it nicely illustrates the expanded graphics of the HP 82162A Thermal Printer.

NANTUCKET



"Perhaps I have opened a door through which others will follow: there may be a computer Picasso hiding somewhere, madly punching keys on his HP-41 and as yet undiscovered."

Robert L. (Keystroke) Gardner
Anamosa, Iowa

Potpourri

Here is a new column that you are going to find in all future issues. It will contain — as the name implies — a mixture of news and miscellaneous announcements to keep you up to date with what is happening in the world of personal calculators and computers.

PPC CONFERENCE IN UK

In the last issue (V6N2p15b), we reported that a users' group had been formed in England. The founder of that group, **David M. Burch,** recently informed us that he had a very gratifying response from the article. He also told us that this new group — now called PPC-UK — will hold a National Conference on Saturday, 9 October 1982, in central London for easy access for most areas of the country.

As an added attraction, **Richard J. Nelson,** founder of the PPC in Santa Ana, California, in 1974, will be on vacation in London and will attend this National Conference. If you can make it, don't miss any presentation he might make. He can make an HP-41 do things that are seemingly impossible!

Mr. Burch also now has his first Journal off the press and intends to publish eight times a year. The PPC-UK covers all of the United Kingdom, and overseas members are welcome.

For information about the National Conference or PPC-UK, contact:

David M. Burch/PPC-UK
Astage,
Rectory Lane,
WINDLESHAM, Surrey
GU20 6BW England
Phone: Home: 0276-75440
or
Chris Tossell
30 Kirkham Road,
Harrogate,
HG1 4EL England
Phone: Home: 0423-504148

NO MORE FACTORY ORDERS

If you didn't read V6N2p8a, "A Change is in Order," maybe you should. We are still rerouting factory orders or returning them. **As of June 1, 1982, Corvallis Division will no longer accept direct-mail and telephone orders for calculators, handheld computers, and accessories.** To order calculators and accessories in the future, please contact your local authorized HP Dealer. They should have stock on hand for immediate purchase. To locate your nearest HP Dealer in the continental United States except Oregon, please call toll-free (800)547-3400. In Oregon, Alaska, and Hawaii, call (503)758-1010.

If you do not live near an HP Dealer, you can order calculators and accessories through the HP Corporate Parts Center. They will accept purchase orders ($20 minimum), checks, money orders, and cash. However, C.O.D., credit card, and telephone orders will not be accepted. Be sure to include a $3.50 handling charge, plus your state and local taxes. Mail your orders to:

Hewlett-Packard Company
Mail Order Department
P.O. Box 7220
Mountain View, CA 94042

This notice and the above changes apply only to domestic orders. Neither Corvallis Division nor the above Mail Order Department will handle orders from anywhere outside the U.S.

The Corvallis Users' Library will continue to accept and process orders for subscriptions (KEY NOTES and Library) and programs. **But remember that the Library does not accept purchase orders for orders under $20.**

NEW CLUBS FORMED

If you live in or near Argentina and want to meet other HP-41 owners who enjoy programming, you can contact **Pablo Montoreano** at this address:

Beruti 2828 8° "B" (1425)
Cap. Fed. Argentina

If you live in or near France, you might be interested in a PPC Chapter that was recently started there by **Jean-Daniel Dodin,** who you can contact at:

77 rue du Cagire
31100 Toulouse, France

If you live in or near Norway, a new HP-41 users' group has been formed. This group is dedicated to HP-41 programming and intends to publish a member's journal every two months. For information, contact:

Norsk Data Samfunn
HP-41C(V) User Group
Tuengen Alle' 11
Olso 3, Norway

It would be helpful to include a large self-addressed and stamped envelope when you write to any of the above clubs.

"BLACK" THERMAL PAPER

Some HP-91/92/97 owners have been using the newer HP 82175A Thermal Printing Paper that was designed ONLY for the HP 82143A and 82162A Printer/Plotters that are HP-41 peripherals. Using the new black thermal paper in your HP-91/92/97 will cause premature printhead wear. Use the HP 82045A paper that is specifically for these older calculators.

PENS FOR CARDS

Quite a long time ago, we reported that the Sanford Sharpie® #3000 soft-tip pen would do a good job of marking magnetic cards. Some time ago, **George M. Langdon** of Stamford, Connecticut, wrote about another Stanford pen, but we mis-filed his letter and only recently found it. He has had good luck with the #3500 Sharpie® *extra-fine-point market with 0.4mm hard plastic tip. It's available in black, red, green, and blue ink [and so is the #3000, now — Ed.*] and works very nicely on Mr. Langdon's cards. Thanks for the tip, Mr. Langdon; sorry about the delay.

® *"Sharpie" is the registered trademark of Sanford Corporation, Bellwood, Illinois.*

NEW HP DEALER AGREEMENT

Earlier this year, Hewlett-Packard and Computerland Corporation entered into an agreement for Computerland stores to market HP's personal calculators, computers, and associated peripheral products. Computerland is the world's largest chain of computer stores, with over 250 stores in the U.S. and 300 worldwide.

Now, in addition to all our former HP Dealers, you have many places in which to see and actually use the HP Series 80 Personal Com-

puters, peripherals, and accessories. Computerland and other dealers that market the Series 80 products also will carry the new HP-75 Portable Computing System.

HOW TO SAVE MONEY

If you use your personal programmable calculator or personal computer — handheld otherwise — to advance yourself in your job, position, or business, make sure you check with your tax accountant about declaring it — and any software, books, or accessories you buy for it — as a tax deduction. It is very probable that you can save quite a bit of money, as this is a perfectly legal deduction if you qualify. Check your income tax instructions for both State and Federal rules.

ERSATZ PROGRAMMING!

Synthetische Programmierung auf dem HP-41C/CV is a translation in German of the book written by **Dr. William C. Wickes** (references: V6N1p7c; V5N3p10a; V4N3p8). So if you live in Germany or if you speak and read German better than English, you probably will be delighted to hear of this translation.

The translator was **Deutsch von Heinz Dalkowski,** and he did an absolutely marvelous job, making many good improvements and even fixing some errors. These 150 pages are essential if you are interested in this subject.

Look first for the book at your local bookstore. If that fails, or if you live outside of Germany, write to the following address for more information. The price in Germany is 30 DM.

Heldermann Verlag Berlin
International Bookshop
Herderstr. 6/7
D-1000 Berlin 41

HAPPY BIRTHDAYS!

Did you know that August marks the Eighth Anniversary of KEY NOTES (originally HP-65 KEY NOTE)? The first issue was 8 pages, and 16,000 copies were printed; however, we had "left-overs" for 7 years! Since then, we've had print-runs as high as 208,350 copies (V5N3). In 8 years, **nearly 2 million copies** have been distributed all over the world. (Back issues are currently available back to V3N3; see page 15.)

Also, the HP-41 was 3 years old on July 16. It is hard to believe that 37 months have come and gone since then. And, no, I'm not permitted to tell you how many there are in the world today. But the word "many" comes to mind!

APOLOGY & CORRECTION

In the last issue (V6N2p3c), the author of the beautiful program, "The Ultimate Calendar — A.D. and B.C. #01593C," was erroneously listed as **William** *Hutchins* of Los Angeles, California. Sorry for the error **Mr. Hitchins;** we apologize for misspelling your last name.

(Do you have any news for this column? Send it to us for consideration. We will print all information that may be of use to our readers — Ed.)

NOTE: The users' groups and clubs mentioned above are not sponsored, nor in any way officially sanctioned, by Hewlett-Packard.

Feedback

This new column contains reader feedback about articles or routines that appeared in previous issues. Though much of the information presented here is useful on its own, you will find that it is a good idea to have your library of KEY NOTES handy while reading this column.

Weimar, California is the birthplace of this comment from **G. Robert Harvey.** This is a timesaving, bytesaving tip.

(V5N2P11c) (41) Even shorter than the "GTO U" instruction (suggested by John O'Shannessy to return fast execution to the top two rows of keys) is simply CF 27. This can be placed as the last command before the RTN or END command in the program. The program global pointer is thus still within the last-run program so that restarting the last-run "user" program again requires only [RTN] [R/S].

As CF requires only 2 bytes and GTO "U" requires 3, 1 byte is saved per program. Also, the 8 bytes required for the short (LBL "U", END) program are not used.

The mention of Seattle, Washington, brings to mind the street musician playing clarinet against the harmonizing call of the seagulls that hover over the waterfront marketplace. Seattle is the home of **Chuck Dinsmore** and an HP-IL system. Here's his input.

(V6N2P9b) (41) The article regarding the sending of 8 data bits gives the clue on how to set the new printer to PARSE mode and how to create a byte value greater than 127. The sequence CLX; ENTER; 1; BLDSPEC: 124; BLDSPEC is used to create byte value 252 by those who don't own the Extended Functions/Memory Module. Those who do own the X-Functions Module simply use 252; XTOA. Values less than or equal to 127 can be created by using BLDSPEC directly. When the byte sequence is built and placed in the ALPHA register, SF 17; OUTA may be executed. Here is a listing of a small routine to set the new printer to PARSE mode. Many thanks for all the valuable information given to us in HP KEY NOTES.

```
01♦LBL "PARSE"     15 49
02 CLA             16 XEQ 01
03 CLX             17 72
04 ENTER↑          18 XEQ 01
05 1               19 SF 17
06 BLDSPEC         20 OUTA
07 124             21 CLA
08 XEQ 01          22 ADV
09 27              23 RTN
10 XEQ 01          24♦LBL 01
11 38              25 BLDSPEC
12 XEQ 01          26 ARCL X
13 107             27 CLX
14 XEQ 01          28 END
```

(If you have more than just the printer in your HP-IL system, be sure to place the device address of the printer in the X-register and execute SELECT before OUTA is encountered in the program. Mr. Dinsmore's letter came to us printed on HP-41 printer paper. None of the words were broken at the end of the line, so we assume he used — you guessed it! — PARSE mode — Ed.)

Jeffrey Smith, of La Palma, California, is a regular contributor to HP KEY NOTES. Here is his improvement on a routine that appeared in the last issue. His additions add to the utility of the routine.

(V6N2P12b) (41) The routine submitted by **W. A. C. Mier-Jedrzejowicz** is very good but, just to be safe; flag 22 should be cleared before the PROMPT, and then tested immediately afterwards. If the flag is still clear, then you can assume that there was no entry and that the stack was not lifted. I have inserted lines 04, 05, 06, and 07; this has the added benefit of re-using the value that was in register nn when there is no entry.

```
01 X<>nn          ; Save X-register
02 RDN            ; Prepare for stack lift
03 "message"      ; User's prompt
04 CF 22          ; Assume no entry
05 PROMPT         ; Permit input
06 FC?C 22        ; Only [R/S] pressed?
07 R↑             ; Then lift the stack
08 X<>nn          ; Save new or old value
```

Denmark is divided into fourteen counties plus the communities of Copenhagen and Fredericksberg. Fyn is one of those counties and its capital city is Odense, the home of **Kim Grau,** who contributed this routine.

(V5N3P13a) (41) The dual temperature conversion routine by **Alan Marcus** uses 63 bytes of program memory plus 4 data registers (this makes a total of "91 bytes"). My version uses only 57 bytes of program memory and no data registers. It works in the same manner as the one of Alan Marcus: the program prompts for the temperature to be converted and displays the dual answers labeled with F and C. Additional data entries followed by R/S skip the "TEMP?" prompt.

```
01♦LBL "TEMP"      14 CLA
02 FS?C 22         15 ARCL X
03 GTO 01          16 "┣F  "
04 FIX 1           17 X<>Y
05 "TEMP?"         18 LASTX
06 PROMPT          19 -
07♦LBL 01          20 1.8
08 ENTER↑          21 /
09 ENTER↑          22 ARCL X
10 1.8             23 "┣C"
11 *               24 CF 22
12 32              25 AVIEW
13 +               26 END
```

(Note that, for both this routine and Alan Marcus' routine, only one of the resulting answers will be meaningful. For instance, if you wish to convert 56 degrees Farenheit to Celsius, input 56, run the routine, and the calculator will display: 132.8F 13.3C. Since the input was in Farenheit, the meaningful answer is in Celsius, or 13.3C. If the input was in Celsius to be converted to Farenheit, the meaningful answer would be the one labeled F. This method of disregarding input units is probably quite effi-

cient for such shorter conversions. Don't forget to key in the two spaces after the F in line 16 — Ed.)

Now, we have this comment from **Cicero Brilho,** of Sao Paulo, Brazil.

(V5N3P12a) I refer to Mr. Waldi's contribution to introduce a German word "Quersumme" and the small program to compute it with the HP-67/97.

I adopted that program to my HP-34C to find out which numbers are equal to the Quersumme of their square, cube, . . ., double, triple, etc. I found that Mr. Waldi forgot in the first case (square) the logic zero and in the second case (cube) he forgot the number 18 and also the logic numbers zero, -1, -8, -17, -26, and -27.

On a pair of dice, there are 36 separate combinations that can result from 1 toss. Of those combinations, 6 give a value of 7 [(6,1) (5,2) (4,3) (3,4) (2,5) (1,6)]. This means that the probability of getting a 7 for 1 roll of a pair of dice is 6 in 36 or 1 in 6; thus, p = 1/6. That was easy. However, what are the chances [P(k)] of getting exactly 5 (k = 5) 7's in 23 (n = 23) tosses of a pair of dice? **Robert Swanson,** of Portland, Oregon, may have the answer.

(V6N1P11a) (41) "BIPROB" is a stack solution for the Binomial Probability Distribution. Just key in n, [ENTER], p, [ENTER], k, XEQ "BIPROB." In about two seconds, see P(k) displayed. Warning: n and k must be non-negative integers. Ranges: 0 < n < 70; 0 < = k < = n; 0 < p < 1. (Excluding the label and RTN, this routine is six lines and 5 bytes shorter than Serge Drogi's similar routine.)

```
01♦LBL "BIPROB"     13 X<>Y
02 SIGN             14 ST- L
03 X<>Y             15 X<> T
04 ST- Y            16 LASTX
05 LASTX            17 Y↑X
06 Y↑X              18 LASTX
07 LASTX            19 FACT
08 FACT             20 /
09 ST/ Y            21 *
10 X<> L            22 *
11 R↑               23 END
12 FACT
```

Next, we offer this comment from **Frank Harrison,** of Paris (we assume that's France).

(V6N2P8c) (41) In the article, "Linear Interpolation on the HP-41", it should be noted that this routine can also be used for extrapolation.

Proof can be obtained from the example given, by substituting the position of 1.75 and 1.58, using the answer given for T(C) of 200.7008 at 1.58. The new answer will be 205.76.

Routines, Techniques, Tips, Etc . . .

The routines and techniques furnished in this column are contributed by people from all walks of life and with various levels of mathematical and programming skills. While the routines might not be the ultimate in programming, they do present new ideas and solutions that others have found for their applications. You might have to modify them to fit your personal application.

There are many prolific HP-41 fans in Australia. The following routine comes from one of those fans. The routine is an interesting concept and you will be the judge of its utility. It was contributed by **Peter Lichtenberg** of Darwin, Australia.

(41) This is a routine to detect input errors of data pairs (X, Y), by using a double entry technique. I am applying it in surveying where coordinates are the data. Because many of my programs use this routine, I use the global label "IN" and, for fast access, position the routine at the end of program memory. It works as follows:

```
01♦LBL "EXAMPLE"
02♦LBL 00           05 RDN
03 "P Y↑X ?"        06 RDN
04 PROMPT           07 X≠Y?
05 XEQ "IN"         08 GTO 01
06 FS?C 00          09 RDN
07 GTO 00           10 RTN
08 .                11♦LBL 01
09 .                12 "ERROR"
10 .                13 CF 21
11 END              14 AVIEW
01♦LBL "IN"         15 PSE
02 X<> T            16 SF 21
03 X≠Y?             17 SF 00
04 GTO 01           18 RTN
```

At line 04 of "EXAMPLE," input the two values twice, in this manner: Y [ENTER] X [ENTER] Y [ENTER] X [R/S].

By inserting, after line 11 of "IN" — 1; ST+ 01; ST+ 02 — and then, after line 08 — 1; ST+ 01 — you will find in register 01 the number of executions of "IN" and in register 02 the number of errors. With this record, I found that 3% of my inputs were wrong, even though I was using the correction key.

Next, we have a routine contributed by **James Glass,** of Sherman Oaks, California. Those with a Time Module will be able to make good use of this routine.

(41) This routine, AUTOCLK, is handy if you want your HP-41 to "wake up" displaying the clock (of course, this assumes you have the Time Module installed). The program lets you turn off the calculator, then turn it on again, and still have the clock active. To clear the clock display, just press the back-arrow key.

```
01♦LBL "AUTOCLK"    05 SF 11
02 SF 11            06 CLOCK
03 OFF              07 GTO 01
04♦LBL 01           08 END
```

With his letter, Mr. Glass included his version of the ADOW routine shown on page 74 of the Time Module manual. His routine made use of the Extended Functions Module to save 1 byte over the original version. Taking his idea, we were able to change a few lines to save a few more bytes. So, here is another, slightly modified, contribution from **James Glass.**

(41) This routine makes use of extended functions to shorten the ADOW routine. If the day number is not needed in the X-register, line 09 may be deleted, and if the display of the day name is not required, line 13 may be deleted.

```
01♦LBL "ADOW"       08 AROT
02 "SATFRITHUWEDTUE" 09 LASTX
03 "MONSUN"         10 ASHF
04 -3               11 ASHF
05 DATE             12 ASHF
06 DOW              13 AVIEW
07 *                14 END
```

In past issues of KEY NOTES we have published several routines to "guard" the HP-41 from use by people unfamiliar with its workings. This next contribution is along those same lines. It was contributed by **Walter Bican,** of Vienna, Austria.

(41) I found another method, that needs no program lines at all and scarcely uses any storage room, to prevent others from fooling around with my HP-41. It goes like this:

Switch to ALPHA, clear ALPHA, and ASTO the cleared ALPHA in the X-register.

If any flags are visible (for example, USER), clear them. Turn off the calculator.

Now, if anybody turns on the HP-41, there is nothing to see. In the case that this person presses the ON key several times, the result is the same. Eventually, this person will assume the calculator is ruined and will leave it alone. The worst that can happen is that the HP-41 will turn itself off after 10 minutes.

(Of course, we who know better, know that entering any number or pressing the correction key will put the HP-41 back in business — Ed.)

What are the advantages of the HP-35 over the HP-41? Well, **Richard Partridge,** of Princeton, New Jersey, can name one.

(41) There are times when I want my HP-41 to display all significant figures, yet suppress trailing zeroes — in other words, to act like an HP-35. There are several ways to do this, none of them simple or elegant. The enclosed routine is the best I have found.

```
01♦LBL "HP-35"      09 10
02 8 E-3            10 *
03 X<>Y             11 ISG Z
04 ENTER↑           12 GTO 01
05♦LBL 01           13♦LBL 02
06 FRC              14 FIX IND Z
07 X=0?             15 X<>Y
08 GTO 02           16 END
```

(Continued)

We recently received two letters that were somewhat complementary. The first letter was from **P. G. Glockner,** who lives in Calgary, Alberta, Canada. Mr. Glockner was interested in finding the most convenient method of accessing registers greater than 99 for the storage and retrieval of data. The second letter came from a Biology student by the name of **Christopher Lafoiet,** who lives in Buie's Creek, North Carolina. Maybe, Mr. Lafoiet's routine is just what Mr. Glockner is looking for.

(41) I am a student in Biology, and I find that in Biometry I have to use registers higher than 99 to store varied numbers from the keyboard. In response to this problem, I wrote a routine that I think would qualify for your "Routines, Techniques, Tips, Etc." column and could be used by those who have loads of data to manipulate. The STOL routine is assigned to the STO key, and RCLL to the RCL key.

```
01♦LBL "STOL"      10♦LBL "RCLL"
02 X<> L           11 X<> L
03 RDN             12 RDN
04 "STO---"        13 "RCL---"
05 AVIEW           14 AVIEW
06 PSE             15 PSE
07 X<> L           16 X<> L
08 STO IND L       17 RCL IND L
09 STOP            18 END
```

Milli, micro, nano, pico — if these prefixes have a familiar ring, then you will be interested in the following technique. It was submitted to KEY NOTES by **Ed Keefe,** of Ankeny, Iowa. The Extended Functions Module is required to implement this technique.

(41) To establish a FIX-ENG display format (displayed numbers overflow and underflow into ENG notation rather than SCI), do the following:

ENG n; RCLFLAG; FIX n; 41.043; STOFLAG.

This will result in both flags 40 and 41 being set. This display format is favored by some electronic technician students. To save this display (plus the status of all the rest of the flags), use RCLFLAG and then store the resulting ALPHA data. The FIX-ENG mode may then be restored by recalling the ALPHA data into the X-register and executing STOFLAG.

Michael D. Daniels recently wrote to us but didn't write his address on the letter. The envelope was separated from the letter and therefore we have to assume he lives in Anytown, U.S.A. Though we don't know where he lives, we do know that he owns an Extended Functions/Memory Module and that he uses this routine.

(41) While using my new Extended Functions/Memory Module, I came across the need to read a program from cards into extended memory. I have found this routine to be very helpful.

To use this routine, one must first press GTO . ., then key in "REX". This makes it the last routine in memory. XEQ "REX," enter the name of the routine to be put into Extended Memory, and that's it! The routine first stores the named program in main memory so there must be room. This routine also destroys the T-register but leaves X, Y, and Z alone.

```
01♦LBL "REX"       09 CF 21
02 "NAME?"         10 AVIEW
03 AON             11 RSUB
04 STOP            12 ASHF
05 AOFF            13 SAVEP
06 ALENG           14 PCLPS
07 "┣ENTER "       15 RDN
08 AROT            16 END
```

(Be sure to key that blank space in line 07 — Ed.)

Over in Germany, **Walter Holl** is making good use of his HP-41 and Printer. Mr. Holl lives in the small town of Nauheim. Here's his routine.

(41 with Printer) The PRPLOT routine does not distinguish between y-values that are outside the selected plotting range (YMIN . . . YMAX) and those y-values that are equal to YMIN or YMAX, respectively. This can result in a misinterpretation of a graph. The following routine avoids this problem by printing an "invalid-symbol" if y exceeds the limits of the plot. To use the routine, simply add XEQ " <>" after the y-value is calculated in the function to be plotted. The sample printout shows a plot of the function y = sin(x).

```
01♦LBL "<>"        09 STO 03
02 RCL 00          10 RDN
03 X>Y?            11 RTN
04 GTO 00          12♦LBL 00
05 CLX             13 "DDDDDD"
06 RCL 01          14 ASTO 03
07 X<Y?            15 END
08 GTO 00
```

```
01♦LBL "SIN(X)"
02 SIN
03 XEQ "<>"
04 END
```

```
  PLOT OF SIN(X)
X (UNITS= 1) →
Y (UNITS= E-1) ↑
 -9.00          9.00
        0.00
      ┣--------┣--------┫
 0            x
 30           ¦    x
 60           ¦         x
 90           ¦         x
120           ¦         x
150           ¦    x
180           x
210     x     ¦
240 x         ¦
270 x         ¦
300 x         ¦
330     x     ¦
360           x
```

Now, let's journey to the "Big Apple," where the bright lights and the beautiful sounds offer round-the-clock entertainment to Miss Liberty. Ah, New York City, the home of **Dr. Keith Bernstein,** who owns an HP-41, a Printer, and an Extended Functions Module.

(41? I have been an HP-41 user for the past ten months. I think the system is great, but I have been unhappy with one feature: the inability to specify formatted fields in the accumulation of fixed numbers with the 82143A Printer. With the addition of the 82180A Extended Functions/Memory Module, I have solved this problem with a simple subroutine titled "FA" (Formatted Accumulation). It requires an HP-41, the 82180A Module, and a Printer.

First, the user inputs the total number of places in the field, remembering to include the radix, possible separators, and any extra spaces to the left of the first printed digit to separate it from the left margin or a previously accumulated field. Then the user specifies the condition of flags 28 and 29 and fixes the number of decimal places. Finally, the number to be accumulated is placed in the X-register by a PROMPT or a register recall, then "FA" is executed.

The result of the use of this subroutine is that the right-most digit and/or decimal point will always lie in the same column, making clear and easy-to-read printouts. I enclose a copy of the subroutine, a printing routine ("DEMO") that uses "FA," and a sample printout from "DEMO" with 4-, 11-, and 5-place fields.

```
01♦LBL "FA"        09 XEQ "FA"
02 CLA             10 11
03 ARCL X          11 FIX 1
04 ALENG           12 SF 29
05 CHS             13 "11 PLACES"
06 RCL Z           14 PROMPT
07 +               15 XEQ "FA"
08 SKPCHR          16 5
09 ACA             17 FIX 0
10 END             18 CF 29
                   19 "5 PLACES"
01♦LBL "DEMO"      20 PROMPT
02♦LBL A           21 XEQ "FA"
03 4               22 PRBUF
04 FIX 0           23 END
05 SF 28           456     7,269.1    12
06 CF 29             2    95,216.6   748
07 "4 PLACES"     4029        12.0     6
08 PROMPT           63       464.9    15
```

Here's another routine for use with the Extended Functions/Memory Module. The author of this routine is **Michael Edwards,** who is a student of engineering computer science at the University of Illinois, in Urbana, Illinois.

(41) This is a routine that finds the amount of space available in an Extended Memory ASCII file. It could be easily converted to a subroutine by omitting lines 2 through 7 and 24 through 32, and by changing line 20 to RTN. The routine returns, to

the X-register, the number of available bytes in the file that is named in the ALPHA register.

```
01♦LBL "FLEFT"      17♦LBL 00
02 FIX 0            18 GETREC
03 CF 29            19 FC? 25
04 "FILE NAME?"     20 GTO 01
05 AON              21 ALENG
06 STOP             22 -
07 AOFF             23 GTO 00
08 0                24♦LBL 01
09 SEEKPTA          25 CLA
10 FLSIZE           26 ARCL X
11 7                27 "├ BYTES LEFT: "
12 *                28 ARCL Y
13 ASTO Y           29 ARCL Z
14 ASHF             30 AVIEW
15 ASTO Z           31 FIX 4
16 SF 25            32 SF 29
                    33 END
```

Lately, we have received a number of routines to round-up (in absolute value) any number (for example, -0.2 becomes -1.0 and 0.2 becomes 1.0). The shortest one that came through the mail had a body (excluding the LBL and END) of 13 bytes and it was from **Bulend Aktay,** of Stuttgart, Germany. Now, 13 bytes isn't bad. However, we were not wholly convinced that rounding-up a number was a function that should occupy more than 1 register of memory. So, three of us, **Chris Coffin** and **Harry Phinney,** of Customer Support, and your technical editor, put our heads together to come up with the shortest sequence of program lines to perform this function. INT: LAST X; FRC; X≠0?; SIGN; +; was our conclusion. That's two bytes apiece for a total of six bytes. The T and L stack registers are changed.

We are quite convinced that this is the most streamlined sequence for rounding-up a number in absolute value while maintaining the same sign (+ or −). However, we will welcome, and probably publish, any routine of 5 bytes or less that performs this same function.

Here's a tip from **John Ioannidis** of Athens, Greece.

(41) END functions much more identically to RTN than you have mentioned. In fact, executing END from the keyboard will result in the line pointer being positioned to the beginning of the program (line 00), just like RTN or GTO.000.

Next, we have a routine that was contributed by **Robert Greenberg,** of Tucson, Arizona.

The routine is designed for use with the printer, but the first portion of the routine (through line 18) can be used without the printer.

(41) I often have a need to see a plot of the values contained in a block of storage registers. For example, as a hydrologist I'm often confronted with having a block of storage registers containing a hydrograph's coordinates and a need to see the plotted hydrograph.

"SP" determines the extrema (the maximum and minimum values) of a specified block of data registers. Using these values as the YMAX and YMIN constraints, "SP" then plots the contents of the block of data registers. "SP" assumes a full plot width of 168 (line 22) and a constant X interval.

Lines 01 through 18 may be used alone (followed by an END or RTN) to determine the extrema without plotting. The maximum and minimum values will be contained in the X and Y registers, respectively.

When "SP" is used in its entirety, it will plot data with the axis at 0 if YMIN < 0, otherwise the axis is placed at YMIN. After execution, the maximum and minimum values are recoverable in the Y and Z registers, respectively. "SP" uses no data storage registers, only the stack so it will plot any block of data registers.

```
01♦LBL "SP"         16 X<>Y
02 STO L            17 ISG L
03 1 E99            18 GTO 00
04 ENTER↑           19 RCL Z
05 CHS              20 SIGN
06♦LBL 00           21 RDN
07 RCL IND L        22 168
08 X>Y?             23♦LBL 01
09 STO Y            24 RCL IND L
10 RDN              25 RDN
11 X<>Y             26 STKPLOT
12 R↑               27 ISG L
13 X<Y?             28 GTO 01
14 STO Y            29 END
15 RDN
```

To use, input a control number (iii.fffcc) similar to that used by the ISG and DSE functions where:

iii: is the lowest-numbered register in the data block

fff: is the highest-numbered register in the data block

cc: is the increment value (if cc is not specified, cc = 01 is assumed)

Example input: 7.01903, XEQ "SP", will find the extrema and plot the contents of registers 07, 10, 13, 16, and 19.

Firenze, Italy, is the home of **Curzio Rositani,** who contributed this next tip. Anyone who has struggled with the development of a program that manipulates a variable-size matrix will appreciate Mr. Rositani's effort.

(41) I am suggesting a simple algorithm, that could be useful to people faced with the problem I have been faced with: the matter of handling matrix indices with the HP-41. I had to find the relationships between the indices themselves. This is not an elementary problem when working with a variable dimension matrix A_{ij}.

The general condition that relates the data-register number of an element to the indices of that element as it would be stored (by columns!) in the matrix A_{ij} follows:

VARIABLE DEFINITIONS:

u: The order of the matrix (number of rows).

a: The data-register number where A_{ij} is stored.

i: Index of row.

j: Index of column.

P_{ij}: The data-register number where A_{ij} is stored.

$$P_{ij} = a - 1 + u(j-1) + i$$

$$i = [(P_{ij} - a) \text{ MOD } (u)] + 1$$

$$j = \text{INT}\left(\frac{P_{ij} - a}{u}\right) + 1$$

(Be sure to note that Mr. Rositani is storing his matrices in columns. That is, a 2 by 3 matrix would be stored in this manner:

$A_{1,1}$ — Register (a)
$A_{2,1}$ — Register (a + 1)
$A_{1,2}$ — Register (a + 2)
$A_{2,2}$ — Register (a + 3)
$A_{1,3}$ — Register (a + 4)
$A_{2,3}$ — Register (a + 5)

If you feel more comfortable storing your matrices in rows, that is:

$A_{1,1}$ — Register (a)
$A_{1,2}$ — Register (a + 1)
$A_{1,3}$ — Register (a + 2)
$A_{2,1}$ — Register (a + 3)
$A_{2,2}$ — Register (a + 4)
$A_{2,3}$ — Register (a + 5),

then you can still use Mr. Rositani's equations, but u becomes the number of columns, i becomes the index of the columns, and j becomes the index of the rows — Ed.)

Book Reviews

Books are reviewed or announced in KEY NOTES only as a service to our readers. A review here does **not** represent an endorsement by Hewlett-Packard. If you are unsure about the contents of a book, we suggest you first check with a local bookstore; if that fails, write to the publisher. Availability problems also should be addressed to the publisher, not to KEY NOTES.

HP-41 RESERVOIR ENGINEERING MANUAL, by **D. Nathan Meehan** and **Eric L. Vogel,** is a new book just released by PennWell Books, a division of PennWell Publishing Company and a producer of many fine books about the petroleum industry.

D. Nathan Meehan is the district engineer in Champlin Petroleum Company's South Texas District in Corpus Christi. He has held a variety of reservoir and production engineering positions in Denver, Oklahoma City, Houston, and Fort Worth, including his most recent assignment as a project supervisor in Champlin's corporate planning and economics and acquisitions groups. Meehan received his BS degree in physics from the Georgia Institute of Technology and his MS degree in petroleum engineering from the University of Oklahoma. While at O.U., he

studied as a Mining and Mineral Fuel Resource Conservation Fellow and received the Henry DeWitt Scholarship for graduate study in mineral fuels. He has published dozens of articles dealing with enhanced oil recovery, reserve evaluation, and fluid properties and is a registered professional engineer.

Eric L. Vogel is a research and development project leader for business application software at Hewlett-Packard's Corvallis Division, which manufactures the HP-41 calculator. He worked part-time for Hewlett-Packard as an undergraduate at Oregon State University, where he received his BS degree in mechanical engineering in 1978. He joined Hewlett-Packard on a permanent basis in 1980 after working as a technical writer for a Corvallis-based software house. His most recent project with Hewlett-Packard is the *HP-41 Petroleum Fluids Pac.*

The HP-41 Reservoir Engineering Manual is a 364-page hardbound book (with dust cover) that contains the most comprehensive set of reservoir engineering programs available for the unique capabilities of the HP-41 handheld computer. Thirty-two practical programs will solve complex reservoir engineering problems, including: decline curves; oil and gas material balance; water influx; static and flowing bottom-hole pressures; gas deliverability; pressure buildups and drawdowns; waterflood performance; well logging; and discounted cash flow analysis.

The book's programs provide accurate performance estimates and forecasts by calculating fluid properties at current reservoir conditions for each step of the analysis. This is made possible only by using the HP 00041-15039 *Petroleum Fluids Pac,* which is *required* for the programs in the book.

Each program includes an overview, equations, examples, user instructions, and program listings. All programs work equally well with and without a printer. The last section of the book contains bar code for all the programs.

The contents are: Oil, Gas, and Water PVT Properties; Decline Curve Analysis; Volumetric Calculations and Reserves; Material Balance; Natural Gas Engineering; Pressure Transient Analysis; Waterflooding; Well Log Analysis; Economics; and Bar Code.

Although this book is not everybody's cup of tea, it definitely is an invaluable tool for any practicing reservoir engineer. The price is $60.* and it is available as follows:

In Japan, Hong Kong, Korea, Southeast Asia, and Australia, write to:

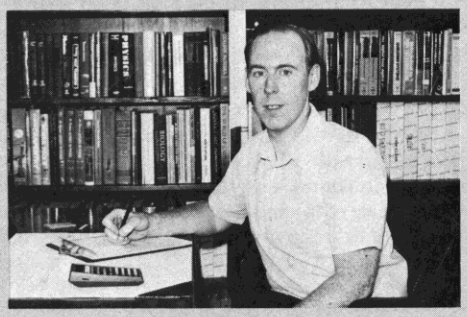
Warren Ball
Intercontinental Marketing Corp.
IPO 5056
Tokyo, Japan 100-31

In the U.S., Canada, South America, Europe, etc., write to:

PennWell Books
P.O. Box 21288
Tulsa, Oklahoma 74121

**U.S. dollars. Foreign orders and orders that do not include payment will have added postage and handling charges.*

CURVE FITTING FOR PROGRAMMABLE CALCULATORS, by **William M. Kolb,** is a 125-page, spiral-bound book in 8.5 by 11 inch format (21.6 by 28 cm), and just off the press.



And if you are even remotely interested in curve fitting, this is the book for you. Long-time readers of KEY NOTES also will recognize the author's name as a long-time contributor to these pages (most notably, his article, "On Understanding Flags, "V1N3 p4).

The book provides all of the essential information needed to fit data to the most common curves. It avoids the usual mathematics and presents, instead, straightforward solutions that can be used with most calculators.

Basically, the book is a collection of curve-fitting formulas intended to help anyone who must occasionally perform data analyses. These formulas can be used to find a specific model for *your* data, or as a guide in choosing among several *possible* models. While not exhaustive, the book does include a comprehensive collection of the most useful one- and two-variable models.

The equations for these models are for scientific calculators and, if your calculator has fewer than 100 registers available, you may need to change register assignments before using the formulas. A positive asset in this book is the fact that register numbers are used consistently throughout, so that you can easily go from one model to another. Furthermore, another nice touch is that graphs of the various equations are provided to help select an appropriate model, and sample problems are included to assist in debugging programs.

There are three major parts. The first is a general discussion of curve fitting, and it is intended as a primer for beginners. Part II contains various statistical models and also the calculations necessary for estimating the coefficients. Part III is a series of appendixes that will help you program these models and even develop new ones.

One particularly neat piece of work in this book is the HP-41 program and bar code in the appendix that automatically fits 19 of the curves. The program displays the equation being fitted, computes the coefficients and adjusted coefficient of determination, and calculates the fitted value of Y for a given X. Data can be entered or deleted at any time. Also, a user-friendly input routine will handle such problem cases as zeros and negative numbers without "crashing" or stopping the program.

(Very, very neat, Bill! — Ed.)

As a final touch, all figures that are used to illustrate the text were drawn on an HP-85 Personal Computer, using a modified version of the HP-85's Standard Pac. The cover design also was done on an HP-85, using a program developed by the author.

The price of this fine book is $14** postpaid anywhere in the U.S. or Canada, and $17** postpaid to all other countries. Mail your orders to:

IMTEC
P.O. Box 1402
Bowie, MD 20716 U.S.A.

THE HP-IL SYSTEM: An Introductory Guide to the Hewlett-Packard Interface Loop, by **Gerry Kane, Steve Harper,** and **David Ushijima,** is a new soft-cover (paperback), 106-page book in 7.3 by 9.1 inch format (18.5 by 23 cm).

Two of the authors of this book are professional writers assigned by the publisher to help the third author, **Steve Harper.** Mr. Harper has worked for Hewlett-Packard about 10 years and is presently serving as R&D project manager for HP-IL interfacing. He was involved in the final stages of the design of the HP-IL integrated circuit and the HP-IL protocol, and is named as a co-inventor on one of the HP-IL patent applications. He graduated from Brigham Young University in 1972 with an MS degree in electrical engineering.

Chapter 1 is an introduction of what the Interface Loop is, how it works, and what it does. In Chapter 2, you will learn how to configure a loop, from the simplest system using present peripherals to the more complex systems that include devices that do not have an HP-IL interface. Included are many, many diagrams and information on how to build a device interface.

From here on, the book gets into more detail, because more detailed knowledge of the loop system is a very necessary requirement if you plan to configure complex or custom-designed systems. Chapters 3, 4, and 5 introduce and cover the topic of HP-IL messages, and how they are used in a typical loop system. These chapters enable you to familiarize yourself with the HP-IL message structure. Messages are also listed in a glossary at the end of the book, plus, a brief summary of the way messages are organized and classified appears in appendix A.

Those interested in the design of HP-IL compatible equipment will find a basic hardware interface explained in chapter 4. Then, the details of the HP-IL functional specification necessary to begin designing an interface are introduced in chapters 6 and 7.

In all, a lot of information about the Interface Loop, but it is well segregated for the novice and the professional. If you intend to *really* apply the HP-IL system, you surely should read this book. It is priced at $16.99** and can be obtained as follows: In the U.S., call (800)227-2895 for charge-orders, or write to:

OSBORNE/McGraw-Hill
630 Bancroft Way
Berkeley, CA 94710

In areas outside the U.S., write to the above address or check at your local bookseller's store. Orders from outside the U.S. will probably have added postage and handling charges. ➔

EduCALC Mail Store at 27963 Cabot Road; South Laguna, CA 92677 U.S.A. is another excellent source for calculator-related books, including those reviewed above. For an order form and a complete list of the well-over-50 different books they carry, write to the above address, and you will receive prompt service. Because they distribute books all over the world, those of our readers who have had trouble obtaining calculator-related books in the past now have a dedicated dealer who will respond to their orders.

PI A La Mode!

You might not find gold or silver anymore in the hills around Jerome, Arizona, but you will find a goldmine of ideas from **Stephen H. Lohr.** Here is a neat trick — or a puzzle — he sent to us for HP-41 owners. It will also work on HP-65/67/97 calculators.

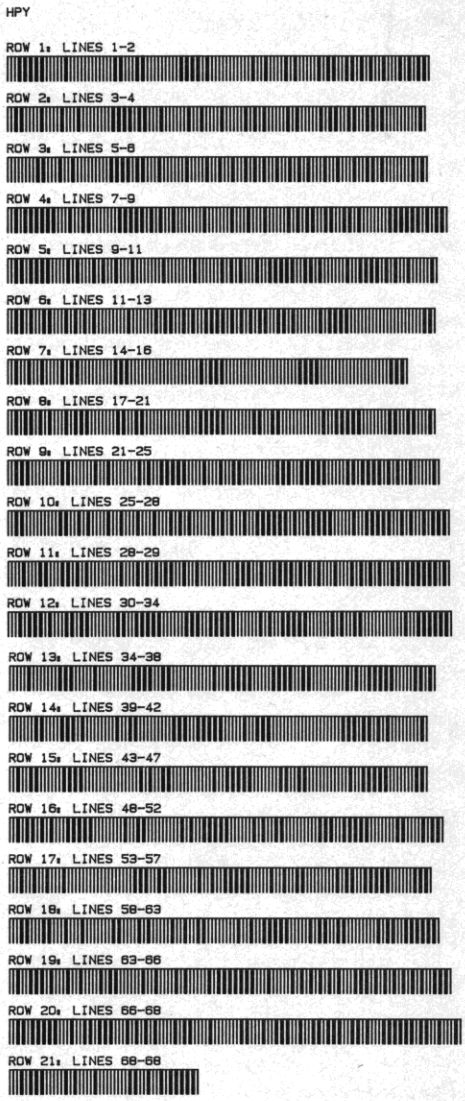
From a Master Clear state, and without using the 0 and 9 keys or the [II] key, and with the fewest possible keystrokes, how do you get pi in the display?

Because it is too easy to just look at the answer before you try to figure it out, we've "hidden" the answer somewhere in this issue. And — we've added *our* answer, which uses *less* keystrokes than Mr. Lohr's answer.

A Key Birth, a Notable Birth

The first HP-65 KEY NOTE (the predecessor to KEY NOTES) was printed near the end of July, 1974. This means that, as of this issue, KEY NOTES is officially 8 years old. So, let's light 8 candles and program the following routine into our HP-41's. You may NOTE that the song is a bit off-KEY. All the credit for the routine goes to it's contributor, **George Flushing,** of Boyton Beach, Florida. You can customize this routine to liven up any birthday party by changing line 28 to the celebrator's name and line 66 to the respective birthdate.

```
01♦LBL "HPY"      35 VIEW 00
02 CF 21          36 TONE 7
03 " HAPPY"       37 VIEW 02
04 ASTO 00        38 TONE 6
05 " BIRTH"       39 VIEW 03
06 ASTO 02        40 TONE 4
07 "  DAY"        41 XEQ 03
08 ASTO 03        42 GTO 04
09 "   TO"        43♦LBL 01
10 ASTO 04        44 VIEW 00
11 "  YOU"        45 TONE 1
12 ASTO 05        46 VIEW 00
13 XEQ 01         47 TONE 1
14 XEQ 02         48 VIEW 02
15 XEQ 01         49 TONE 2
16 XEQ 03         50 VIEW 03
17 VIEW 00        51 TONE 1
18 TONE 1         52 RTN
19 VIEW 00        53♦LBL 02
20 TONE 1         54 VIEW 04
21 VIEW 02        55 TONE 4
22 TONE 8         56 VIEW 05
23 VIEW 03        57 TONE 3
24 TONE 6         58 RTN
25 " DEAR"        59♦LBL 03
26 AVIEW          60 VIEW 04
27 TONE 4         61 TONE 5
28 "KEY NOTES"    62 VIEW 05
29 AVIEW          63 TONE 4
30 TONE 3         64 RTN
31 TONE 2         65♦LBL 04
32 TONE 2         66 "AUGUST, 1974"
33 VIEW 00        67 AVIEW
34 TONE 7         68 .END.
```



Back Issue and Subscription Information

Back issues of KEY NOTES are available back to V3N3, which introduced the HP-41. An index of these will be furnished on request. Available issues are:

V3N3 August 1979 (12 pages)
V3N4 November 1979 (12 pages)
V4N1 March 1980 (12 pages)
V4N2 Jun-Aug 1980 (12 pages)
V4N3 Sep-Dec 1980 (12 pages)
V5N1 Jan-Apr 1981 (16 pages)
V5N2 May-Aug 1981 (16 pages)
V5N3 Sep-Dec 1981 (16 pages)
V6N1 Jan-Feb 1982 (16 pages)
V6N2 Mar-May 1982 (16 pages)

Prices for KEY NOTES back issues are as follows. All prices include first-class or air mail. Payment must accompany your order and must be a check or money order in U.S. dollars drawn on a U.S. bank. Or you may use your American Express, VISA, or MasterCard account; be sure to include your account number and card expiration date. Your order will be promptly mailed in an envelope.

NO. OF ISSUES	U.S., MEXICO CANADA	ALL OTHER COUNTRIES
1	$2.00	$3.50
2	$3.50	$5.00
3	$5.00	$6.50
4	$6.00	$8.00
5	$7.00	$9.00
6	$8.00	$10.00
7	$9.00	$11.00
8	$10.00	$12.00
9	$11.00	$13.00
10	$12.00	$14.00

KEY NOTES is published quarterly in February, May, August, and November. A one-year subscription in the U.S. and Canada is $5* a year. It is free (worldwide) in 1982 if you are a member of the Corvallis Users' Library ($20* U.S. and Canada; $35* elsewhere). Send your payment and complete name and address to the Corvallis address on back cover.

To get KEY NOTES in Europe, contact the UPLE (Geneva address on back cover). To get KEY NOTES elsewhere, contact your nearest HP Sales Office or send your name, address, and calculator serial number to the Corvallis Users' Library.

**U.S. dollars. See note at bottom of page 6.*

What's in a Name?

For everyone who *really* reads KEY NOTES, here is an opportunity to become famous. And the individual who *does* become famous will owe such luck to **Dr. Roger D. Metcalf** (Arlington, Texas), who is responsible for this article.

Several weeks ago we received a letter from Dr. Metcalf. Here is an excerpt from it.

"Lately I have been having an identity crisis with my HP-41CV; that is, is it a super sophisticated calculator or true computer? Now, it seems as if some of the publications have solved my problem by considering the HP-41 a true handheld computer (See *Popular Electronics,* July 1982, "PE Compares New Handheld Computers.")

"So, now that I *know* the HP-41 is really a computer, my new worry is, what language does it speak? Certainly not BASIC, FORTRAN, etc. Sort of a FORTH, maybe? And RPN doesn't even come close to describing *all* the capabilities.

"How about a contest to 'Name That Language!'? Perhaps even a Custom Keyboard could be sent to the person who suggested the contest? (My address below!) I truly believe that a formal name for our 'Language' would add credibility in the popular press ..."

Well, Dr. Metcalf, *you* might have been jesting, but your editor isn't, as you already know, because *he knows* you now *have* that Custom Keyboard (HP-41 Touchpad) that no amount of dentistry could ever get for you before now. See how we really *do* listen? And *read* your letters?

So, honored KEY NOTES readers, Dr. Metcalf has *his* reward; now which one of you is going to win our contest to "Name That Language"? The rules are very simple, and everyone except Hewlett-Packard employees and their families may enter.

1. The "Language" name must be a clever acronym such as: radar, BASIC, scuba, FORTRAN, sonar, COBOL.
2. The acronym must relate to the HP-41.
3. The acronym must be original (new).
4. Only one entry per person is permitted.

And if you don't know the definition of acronym, here is the dictionary copy: "A word formed from the initial letter or letters of each of the successive parts or major parts of a compound term." For example: BASIC = **B**eginner's **A**ll-purpose **S**ymbolic **I**nstruction **C**ode; or: COBOL = **CO**mmon **B**usiness **O**riented **L**anguage. As you can see, we are giving you lots of latitude in your choice.

The contest ends at midnight (our time) on October 22, 1982, and the winner will be chosen by your editor and your technical editor, and their decision will be final. The winner will be announced in the November issue (V6N4); and the winner will receive as prizes:

1. An HP-41 Touchpad;
2. Four Library programs (of choice);
3. A personal, original, gift from the editor;
4. Worldwide fame and the envy of his/her peers!

Who knows? You just *might* get in future history books ... .

Lohr: [XEQ] RAD [COS⁻¹] [ST] [+] [·] [×]
Ed: [XEQ] RAD [COS⁻¹] [ENTER] [+]

The Long and Short of Multiplication

Here's a routine that should catch the eye of any elementary or high school mathematics teacher. **Jonathan Moffett,** who is a high school teacher in Pittsworth (Queensland), Australia, contributed this jewel.

(41) Long multiplication is still taught in schools. Checking the answer is easy, but how does a teacher pinpoint where an error occurred? I find the following routine useful. The routine multiplies any two positive integers provided that the result is not greater than 1 E10. On the prompt "N M" key in: number, [ENTER], multiplier, [R/S]. If a printer is attached, the whole problem is printed out. Otherwise, since flag 21 is set, press [R/S] to display each line. Since the product is left in the X-register, it is possible to chain problems. Flags 28 and 29 should be clear.

```
01♦LBL "LM"
02 FIX 0
03 CF 01
04 SF 12
05 SF 21
06♦LBL 01
07 "N↑M"
08 PROMPT
09 STO 02
10 X<>Y
11 STO 01
12 VIEW X
13 *
14 STO 00
15 "X "
16 ARCL 02
17 FS? 55
18 ACA
19 ADV
20 FC? 55
21 AVIEW
22 XEQ 02
23 RCL 02
24 LOG
25 INT
26 X=0?
27 SF 01
28 .1
29 %
30 STO 03
31♦LBL 03
32 RCL 02
33 10
34 /
35 INT
36 STO 02
37 LASTX
38 FRC
39 10
40 *
41 RCL 03
42 INT
43 10↑X
44 *
45 RCL 01
46 *
47 FC? 01
48 VIEW X
49 ISG 03
50 GTO 03
51 FC?C 01
52 XEQ 02
53 RCL 00
54 VIEW X
55 FS? 55
56 XEQ 02
57 CF 12
58 ADV
59 RTN
60♦LBL 02
61 "------"
62 ASTO T
63 ARCL T
64 AVIEW
65 END
```


HP KEY NOTES

June-August 1982 Vol. 6 No. 3

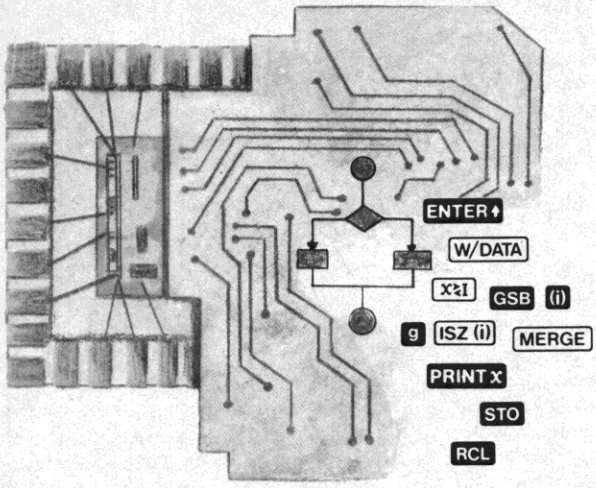
Programming and operating tips, answers to questions, and information about new programs and developments concerning Hewlett-Packard handheld computers. Published quarterly. See page 15. *Reader comments or contributions are welcomed. Please send them to one of the following addresses.*

Hewlett-Packard Company
Users' Library
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 USA

Hewlett-Packard SA
Users Program Library Europe
7, Rue du Bois-du-Lan
P.O. Box, CH-1217 Meyrin 2
Geneva-Switzerland

Hewlett-Packard Company
Corvallis Division
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 U.S.A.

Address Correction Requested



Featuring, this issue:

HEWLETT PACKARD

September-December 1982 Vol. 6 No. 4
ISSN 0730-2037

HP Key Notes

Published Quarterly | By Subscription: $5/yr. | $2.00

HP Has the Right Tool for the Job

On November 3, Hewlett-Packard began a major media campaign to promote its full range of personal computation products. In such media as *The Wall Street Journal, Business Week, U.S. News & World Report, Time, Fortune,* and *Newsweek,* you can see advertisements of the new products now available for every facet and level of personal computation.

Because the Corvallis Facility is the "home" of the Series 10, 40, 70, and 80 families of personal computation products, we are proud to present a photo, used in the above-mentioned campaign, of HP's *complete* range of personal computation products to show you how prominently *your* personal machines figure in this splendid line-up.

In the foregound are two brand-new HP products: on the left is the HP 120 Personal Office Computer and on the right is the HP Series 200 Model 16 Personal Technical Computer, both produced by other HP Divisions. Information about these and other Series 100 and 200 computers can be obtained from your local HP Sales Office.

In the background, left to right, are the Series 80 Personal Computers from Corvallis: the *portable* HP-85, the *powerful* HP-87, and the *personal-priced* HP-86. And if you don't recognize it, that's the new and wonderful HP-75 at right, center; it's in a class by itself. If you haven't seen it yet, you are missing a real contribution to the world of personal computation.

Last, but by no means least — at left, center — are the super-portable, personal Series 10 calculators and the forerunner of handheld personal computers, the venerable HP-41.

If you can't locate an HP Sales Office near you, call our toll-free number 800-547-3400 (U.S. only). (In Oregon, Alaska, and Hawaii, call 503-758-1010.) TTY users with hearing or speech impairments, please dial 503-758-5566.

NEW...For You, Inside...

It seems fitting that we end this year with a lot of new things for you, just in time for your Christmas shopping. And you will notice right away that this issue is thicker and heavier than usual!

NEW HP-10C CALCULATOR

Turn to page 3 to see the latest in HP low-cost, high-quality, thin-line calculators. Yet, despite its low price, it is programmable, plus lots more.

NEW HP-41 GRAPHICS CAPABILITY

Turn to page 4 for another surprise from Hewlett-Packard. You'll find the new HP 7470A Plotter and the new HP 82184A Plotter Module. Combine these with your HP-IL system and you have a low-cost graphics system that is very, very hard to beat!

NEW HP-12C TRAINING GUIDE

Turn to page 9 to see the newest in aids for anyone who cannot unravel the mysteries and intricacies of financial functions and transactions. And with the present state of interest rates, mortgages, and investments, this new book would appear to be the answer to many present-day problems.

NEW HP-12C REAL ESTATE HANDBOOK

Turn to page 20 and you will find the new *HP-12C Real Estate Applications Handbook* that we ran off the presses just in time for Christmas. This handy handbook is not a dissertation in jargon and unintelligible formulas and algorithms. It *shows* you by example, how you can make decisions that will make your career profitable instead of profitless. If you are in any facet of real estate, don't miss this!

All prices in this newsletter are suggested retail prices excluding applicable state and local taxes—Continental U.S.A., Alaska, and Hawaii.

CORVALLIS USERS' LIBRARY NEWS

The new Corvallis Users' Library HP-41 "cassette-based" submittal contest is still underway. (And if you were a member of the Corvallis Users' Library, you would have heard about it before now!) The prospect of winning an HP-75C Portable Computer should convince even the most reluctant programmer to take pen in hand. Five Grand Prizes of HP-75C Portable Computers, plus Ten Series 10 (HP-11C/12C/15C/16C) calculators will be awarded to the authors of the fifteen programs judged most outstanding by our review staff. The contest rules are consistent with previous contests, except these programs must be HP-41/HP-IL cassette-based. This contest ends December 31, 1982, but don't wait until the last days to submit your entry. Send your entry early enough so that if the review staff should have to return your program for modification, you will have time to resubmit your program before the contest deadline.

You probably notice that only the June winners of our last Corvallis Library contest appear in this issue. Our review staff is working "day and night" but the response to our contest was tremendous; so . . . next issue will contain the July and August winners.

HP-41 SOLUTIONS BOOK UPDATE

Here's the rundown on what is new, soon to be in stock, and upcoming.

00041-90443 HP-41 Games II — in stock and shipping (and extremely popular!). $12.50*

00041-90441 Structural Design — available in December 1982. $35.00*

00041-90442 HP-41 Matrices and Complex Numbers — watch for announcement in next issue.

HP-75 USERS' LIBRARY NEWS (CORVALLIS)

With the introduction of the HP-75C also comes both an HP-75 Users' Library and eleven HP-75 Solutions Books. The Solutions Books are $35.00* each (including magnetic cards), and they are available in the following applications.

Math I	- 00075-13003
Math II	- 00075-13004
Math III	- 00075-13005
Games I	- 00075-13006
Games II	- 00075-13007
Electronics	- 00075-13008
Finance	- 00075-13009
Real Estate	- 00075-13010
Statistics	- 00075-13011
Test Statistics	- 00075-13012
I/O Utilities	- 00075-13013

FREE MEMBERSHIP — FOR A WHILE . . .

For a limited time, a complimentary HP-75 Corvallis Users' Library membership will be offered with the purchase of the HP-75. (At present, this offer is limited to only the U.S. It will probably be extended to include Europe and maybe elsewhere, but we will let you know in the next issue.) The membership includes a *Catalog*, documentation guide, submittal forms, and the program "EASYMEMO," which can be used to format your letters or memos. We encourage all new HP-75C owners to help launch our new Library by documenting any programs that may be of interest to others and by sending them to:

The HP-75 Corvallis Users' Library
1000 N.E. Circle Blvd.
Corvallis, OR 97330

***BEFORE YOU ORDER** any of the Solutions Books mentioned above, make sure you first check with your local authorized HP Dealer. Remember: All direct orders (whether to Corvallis or to the Corporate Parts Center) must include a $3.50 handling charge *per order*. If you live outside the U.S. and Canada, you should check with your local HP Sales Office or authorized HP Dealer for availability of these books. The above ordering and pricing information does NOT apply outside the U.S. and Canada. Also, all payments must be in U.S. dollars.

WINNING PROGRAMS

The June winners of the 1982 Users' Library Submittal Contest are featured below. Although the contest ended in August, as of the press date for this issue of KEY NOTES, the review staff has chosen the winners only through June. The contest brought in such a tremendous number of program submittals, we will have to feature the winners for July and August in the next issue of KEY NOTES.

The winning programs for June cover a broad range of interests. The authors of these programs each received a fine HP product for their fine contributions to the Users' Library.

JUNE WINNERS

(41) Electric Rate Analysis II #01983C (Price: $12*)

This is a general purpose electric rate analysis program to compare two rates. It supports Hopkinson and Wright type demand rates with up to five energy blocks, up to three demand blocks, a customer charge, and monthly minimum. The two rates need not be of the same type. Both rates and the results of comparisons are printed. *Required accessories: 3 Memory Modules and Printer.* (734 lines, 1515 bytes, 30 pages)

Author: **John Brown**
Boise, Idaho

(97) Prediction of Seasonal Heating Costs With a Heat Pump #04836D (Price $16*)

This set of programs uses weather data, heat pump capacity specifications, and building loss relations to calculate seasonal heating energy requirements and estimated costs. Comparison is made to similar installations heated with gas and oil. Analysis of life-cycle energy savings versus investment is included. *Required accessories: HP Library programs #00252D or #00564D.* (440 lines, 58 pages)

Author: **James Baskerville**
McLean, Virginia

(41) Design and Rating of Packed Columns #01962C (Price: $10*)

Using equations fit to pressure drop curves published by the Norton Company, the design diameter is calculated for a packed column at flooding or six other user-specified pressure drops. Given diameter, the program will rate the column (calculate the pressure drop). Properties required are: densities, liquid viscosity, flow rates, and packing type. *Required accessories: Two Memory Modules.* (447 lines, 1052 bytes, 22 pages)

Author: **Robert Wooley**
Midland, Michigan

(41) Pharmacokinetic Parameters From Serum Drug Concentrations #01878C (Price: $16*)

This program determines k and V from two drug levels, or k from an assumed V and one drug level, during maintenance dosing with any of four dosing schedules: intermittent, fixed interval; intermittent, two fixed intervals; intermittent, non-uniform does, non-uniform interval; continuous intravenous infusion/sustained release. A loading dose and/or one additional level taken before maintenance dosing begins can be taken into account for additional flexibility. Steady state concentrations are projected when dosage regimen and pharmacokinetic parameters are known. *Required accessories: Card Reader, Quad Memory Module.* (715 lines, 1585 bytes, 41 pages)

Author: **E. Maurice Jones**
Charleston, South Carolina

(41) Flow Computations For Various Open Channel Configurations #01957C (Price: $14*)

This program computes flow, given normal depth, or normal depth given flow,, and critical depth, for rectangular, triangular, trapeziodal, circular, or parabolic channels. Standard output is normal depth, top width, and flow. The user may select additional outputs including area, wetted perimeter, hydraulic radius, and/or average velocity. The program accepts S.I. or U.S. units. *Required accessories: 3 Memory Modules.* (554 lines, 1442 bytes, 35 pages)

Author: **Martin Hanson**
Park Falls, Wisconsin

(41) Time Domain Analysis for Nonlinear Networks #01909C (Price: $12)

This Calculator-Aided Design program can perform DC or Time Domain analysis for arbitrary networks. Components allowed are linear inductive, capacitive, resistance, independent current/voltage sources, current- or voltage-controlled current sources, and user-defined nonlinear devices. Modified companion models and Newton-Raphson algorithm are used. Outputs are all the node voltages. *Required accessories: One Memory Module.* (649 lines, 1106 bytes, 27 pages)

Author: **Nai Chi Lee**
Stony Brook, New York

(41) Queue #01896C (Price $18*)

"QUEUE" is a fully integrated program enabling the user to compute queueing statistics for 6 different queue types. The program has been written in a modular manner so the user may delete segments of the program that do not apply to his/her situation. This feature enables the user to run even the longest of the routines on an HP-41C with only 1 memory module! The types are as follows — type A: single server model with arbitrary service times; type B: multiserver model with Poissen arrivals and exponential service times; type C: basic single

server model; type E: models with a finite calling population; type F: single server model with arbitrary service times and a priority queue discipline. *Required accessories: 3 Memory Modules.* (908 lines, 1610 bytes, 55 pages)

Author: **Lauren Hansman**
Kitchener, Canada

(41) Crane/Outrigger Reactions and Stability #01952C (Price: $8*)

This program computes outrigger float reactions or wheel tandem reactions for a truck crane, and internally checks stability for specified lift loads, horizontal boom angle, and operating radius. A valuable program for those concerned with truck crane installations such as construction sites, piers, and docks. A warning "TIPPING" is displayed if instability is detected. *Required accessories: 2 Memory Modules.* (473 lines, 959 bytes, 18 pages)

Author: **Charles Dinsmore**
Seattle, Washington

(41) Joint Venture Financing #01894C (Price: $12*)

Investment is recovered by a two-part levy on sales. The first levy period runs until capital plus compound interest is recovered. The second stage usually equals the first levy period; alternatively, the user can specify first or second periods. Program calculates levies, first levy period, money and real internal rates of return. *Required accessories: Three Memory Modules.* (715 lines, 1351 bytes, 25 pages)

Author: **Clive Goldman**
London, England

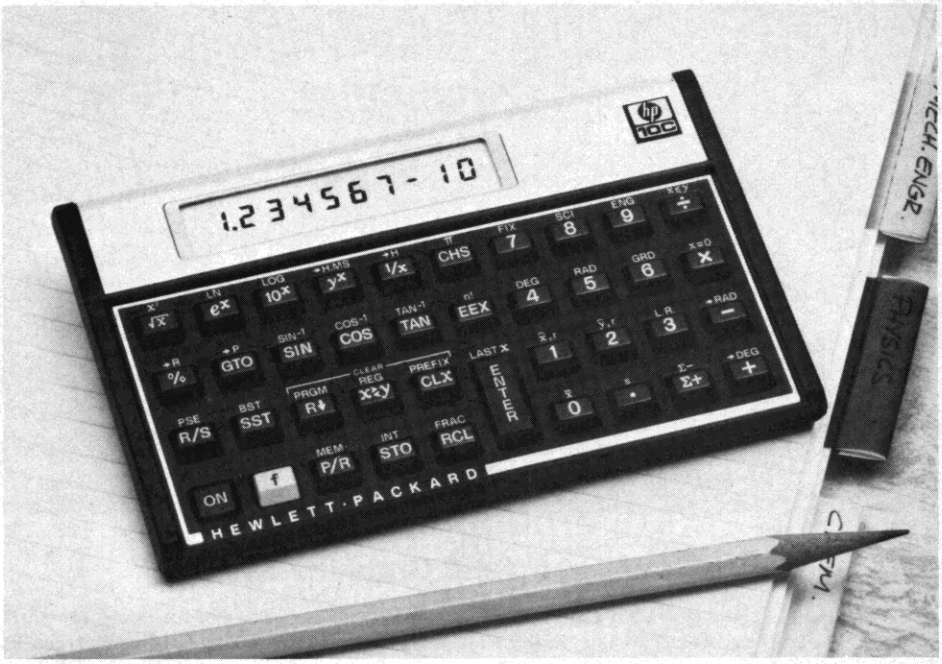
(41) Radar Plotting With Timer #01908C (Price: $8*)

This program plots, concurrently, five radar targets. It returns relative, truecourse, speed of targets, CPA times, distances, bearing. In the process, users ship may alter course or speed. In users ship simulated course, speed alterations, it forecasts plotted targets CPA times, distances, bearings for best course-of-action. *Required accessories: HP 82182A Time Module.* (339 lines, 563 bytes, 14 pages)

Author: **Matteo Kalcic**
Trieste, Italy

(The "big winners" for June were Mr. Wooley and Mr. Jones. They each chose a new Digital Cassette Drive, or a comparable prize. All of the other June winners received an Extended Functions/Memory Module or, again, a comparable prize — Ed.)

**U.S. dollars. Orders from anywhere outside the U.S. must include a negotiable check (or money order), in U.S. dollars, drawn on a U.S. bank. All orders from anywhere outside the U.S. and Canada must include an additional 10 percent fee for special handling and air mail postage. (For example, an order for two programs = $6 x 2 = $12 + $1.20 = $13.20 total.) If you live in Europe, you should order KEY NOTES Programs directly from the Geneva UPLE, but make certain you make payment as required by Users' Program Library Europe; the above $6 fee is good only for orders to the Corvallis Library.*



HP-10C: The New Low End, 10 Years After the HP-35A

Remember the HP-35A? A revolutionary concept in 1972. An "electronic slide rule" with one register, *no* programming, and — oh, yes — a nice, soft, black leather case. All for the "affordable" price of $395, factory list. And, it was as hard to find at dealers as it was to get delivery from HP.

Yet . . . had the electronic computing industry gone the way of inflation instead of each generation of calculator getting more powerful and cheaper, a calculator like the HP-35 would today cost $740.

But the programmable HP-10C — with 79 lines of program memory that can be traded-off for 10 registers — is *at least* 10 times more powerful than the HP-35. Just think: assuming the inflationary process mentioned above had become reality, and if the relationship between power and price were linear, the HP-10C would today cost over $7,400!

Instead of such a horrible fate, the HP-10C lists for a small percent of the HP-35's original price. So it is indeed the least expensive programmable *ever* produced by HP. Plus, it is a great *value*: it offers a virtually unbeatable combination of power and features, some of them unique in this price range.

One of the HP-10C's unique features is conditional branching. By virtue of this, the HP-10C is probably the least expensive "decision-making" machine in today's market. To this, the HP-10C adds other traditional HP features that enhance the power of a calculator and make programming and editing so much easier, such other features as merged keycodes, forward and backward program review, RPN, LAST X, pause, and the convenience of slim-line LCD design, long-life batteries, *Continuous Memory,* and automatic shut-off.

The HP-10C is the fifth model of the new Series 10 calculators. The HP-11C and HP-15C are also scientific programmables, with 203 and 448 lines of program memory, respectively. The HP-12C, also programmable, is designed for financial applications plus a large range of other special functions. The HP-16C is a virtually one-of-a-kind specialized calculator for computer science and digital electronics applications. This entire Series was created not only for maximum pocketability but also for durability. The HP-10C could very well be the most indestructible of the lot.

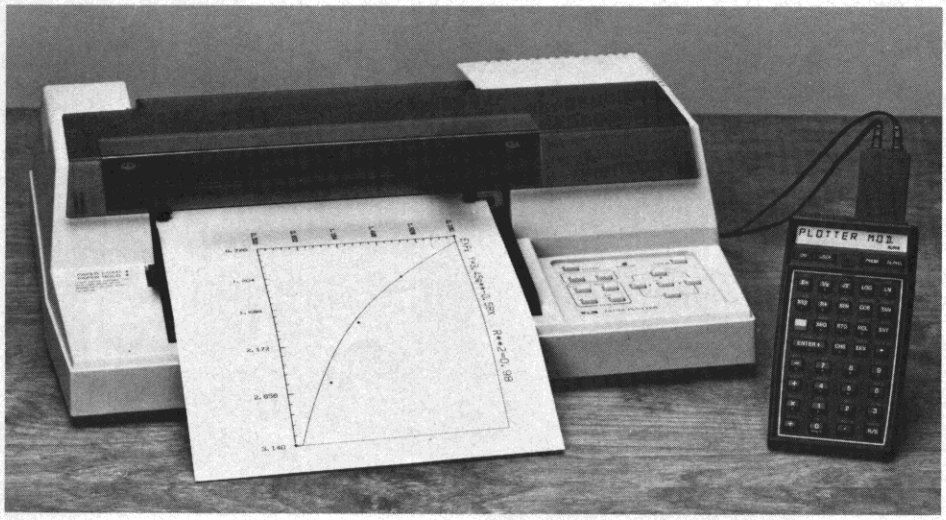
In addition to having a full complement of math, trig and log functions, the HP-10C keyboard includes rectangular/polar-coordinate conversions, degree/radian conversions, decimal hours/hours-minutes-seconds conversions, mean and standard deviation, linear regression and estimate, summations, correlation coefficient, and factorials. Also, numbers can be displayed in fixed-decimal, scientific, or engineering notation.

The HP-10C may be the least-expensive programmable calculator in the HP line, but you will find that it is way, way beyond the least-useful on the market. And with Christmas rapidly approaching, maybe now is the time to go see your local HP Dealer and get a head-start on what most of us refer to as "last-minute" Christmas buying.

"NANTUCKET" Coming Next Issue...

Because a lot of people are curious about how **Robert L. (Keystroke) Gardner** managed to get such a nice "painting" on his stock HP 82162A Thermal Printer, in the next issue we are going to tell you how he did it. Quite a few readers have written that they have driven themselves to distraction trying to emulate his "Nantucket painting."

Well, there's a lot more to the story than just the printer capability. We have discovered that Mr. Gardner is in a class all by himself, and we want you to read about him in the next issue. We'll bring you a story you will enjoy no end, and we'll let Mr. Gardner tell you in his own words how he managed to make that printer do some things that are seemingly not possible. So, until next February, don't waste your time trying to copy his "art," unless you've already discovered the "secret."



Announcing HP's Lowest- Cost Graphics System!

On November 1, Hewlett-Packard introduced two new products that, when combined with the HP-41 and HP-IL module, provide a complete system for generating graphics and bar code.

The **HP-41 Plotter Module (82184A)** is our newest software product for the HP-41. This 8KB plug-in module provides HP-41 language extensions that make plotting easy and fast. You can develop your own programs using the 52 plotter functions (including 10 bar code functions), or you can use the Utility Plotting Program to generate framed, labeled plots with a minimum of preparation. The Utility Plotting Program contains 5 plotting routines that use your inputs to automatically generate a plot of a math function or a series of points that you enter.

The Plotter Module enables an HP-41 to control HP plotting devices that operate in HP-IL and are compatible with the HP Graphics Language (HP-GL). HP's San Diego Division has recently announced a new version of their **HP 7470A Plotter** — with the HP-IL Interface. The HP 7470A (Opt. 003) will be available after November 1. The plotter accepts two standard media sizes: 8.5 x 11 inch (ANSI A) and 210 x 297 mm (ISO A4), and plots on paper and transparency film.

The HP 7470A uses low-inertia, microgrip technology. The medium is driven back and forth across the platen for plotting along the x-axis. Pen movement locates points along the y-axis. Movement of both paper and pens allows the HP 7470A to plot lines at speeds of up to 15 inches (38 cm) per second.

The HP-41 plotter system is easy to use and inexpensive to own. With the HP-41 Plotter Module and the HP 7470A Plotter, you can create multi-color line graphs, bar charts, and text pages to improve your presentations. Or use your HP-41 for data collection in engineering or statistical studies, then plot the results for better interpretation and analysis. Graphics makes complex data easier to understand and communicate.

An added feature of the Plotter Module is its bar code capability. Using the Plotter Module with the HP 7470A Plotter enables you to plot standard HP-41 bar code for the HP 82153A Optical Wand, as well as bar code for scanning devices used in other bar code systems. The module also enables you to print HP-41 bar code on the HP 82162A Thermal Printer.

The HP-41 Plotter Module and HP 7470A (Opt. 003) Plotter are available now at your local dealer.

PLOTTER EXAMPLE: EXPONENTIAL CURVE FIT

This program is a remake of the curve fit program in the "Standard Applications" PAC. It only fits data to an exponential curve but it could easily have been designed to fit any or all of the other types of curve fits. It uses "X?" and "Y?" from the plotter module to prompt the user for the X and Y coordinates of each point. When all data points have been entered the user simply presses [R/S] in response to the next request for an X coordinate.

The program automatically computes the correct scale for plotting and then plots a "*" at each data point. It then computes the equation for the exponential curve that best fits the data, and plots 20 points using this equation. Finally, it labels the graph with the computed equation in the form "EXP: Y=ae**bX" and the coefficient of determination in the form "R**2=".

The data points used in the example are the same as those used in the Standard PAC manual.

Xi: 0.72 1.31 1.95 2.58 3.14
Yi: 2.16 1.61 1.16 0.85 0.50

```
01♦LBL "EXP"
02 CLRG
03 "Y?"
04 ASTO 08
05 "X?"
06 ASTO 05
07 .0242
08 STO 02
09 505.00505
10 STO 03
11 STO 01
12 STO 07
13 XROM "PLINIT"
14 XROM "PLTUXY"
15 4221
16 STO 02
17 XROM "PLANOT"
18 XROM "PLTUXY"
19 ΣREG 12
20 CLΣ
21 RCL 08
22 1 E3
23 *
24 INT
25 1 E3
26 /
27 STO 05
28♦LBL 13
29 RCL IND 05
30 ISG 05
31 RCL IND 05
32 LN
33 X<>Y
34 Σ+
35 ISG 05
36 GTO 13
37 RCL 17
38 RCL 13
39 RCL 12
40 RCL 12
41 XEQ 09
42 STO 20
43 RCL 14
44 RCL 13
45 RCL 12
46 RCL 16
47 XEQ 09
48 RCL 20
49 /
50 STO 21
51 E↑X
52 STO 18
53 RCL 17
54 RCL 16
55 RCL 12
56 RCL 14
57 XEQ 09
58 RCL 20
59 /
60 STO 19
61 RCL 21
62 RCL 14
63 *
64 RCL 19
65 RCL 16
66 *
67 +
68 RCL 14
69 X↑2
70 RCL 17
71 /
72 STO 22
73 -
74 RCL 15
75 RCL 22
76 -
77 /
78 STO 22
79 "PLT"
80 FIX 2
81 ASTO 08
82 -19
83 STO 05
84 11
85 STO 02
86 XROM "PLTUXY"
87 1
88 PEN
89 LORG
90 5
91 CSIZE
92 SETGU
93 97
94 ENTER↑
95 30
96 MOVE
97 SF 17
98 "EXP: Y="
99 ARCL 18
100 "+e**"
101 ARCL 19
102 "+X"
103 LABEL
104 "   R**2="
105 ARCL 22
106 LABEL
107 CLX
108 PEN
109 STOP
110♦LBL 09
111 *
112 STO 23
113 RDN
114 *
115 RCL 23
116 -
117 RTN
118♦LBL "PLT"
119 RCL 19
120 *
121 E↑X
122 RCL 18
123 *
124 .END.
```

COMING IN THE NEXT ISSUE:

Richard J. Nelson, editor of the *PPC Calculator Journal* (see page 12 footnote) will contribute a technical article on the HP-41 system. Also, because he is just back from the PPC Conference in the United Kingdom, perhaps he will report on that and his experiences in other parts of Europe. Don't miss it!

Milliseconds Away

This letter, from **Patrick Schibli,** of Berneck, Switzerland, leads nicely into the subject of function timing with the HP 82182A Time Module.

As a matter of fact, computers add two numbers faster than they can multiply them. Therefore, I propose that everyone change their: 2; *; program sequences to: ST+ X (unless the LASTX register is used). Up to four "ST+ X" statements work faster than a simple multiplication by 2, 4, 8, or 16, respectively. However, using "ST+ X" to double a number more than once consumes extra bytes.

Mr. Shibli is right. Using the "ST+ X" function, rather than 2; *, to double a number, will save time. If you have a Time Module, it is easy for you to determine just how much time you do save. With only a Time Module connected to your HP-41, the time savings will be approximately 60 milliseconds. Now, the time it takes to execute a function is a "touchy" subject, because there are many variables that affect the operation speed of the HP-41 and some of those variables are beyond our control. Most functions have execution times that depend even upon the numbers involved. What follows, in this article, is an explanation of how you can use your Time Module to determine, for a given HP-41 system configuration, whether one programming method executes more quickly than another.

For a particular condition, timing a function, using the Time Module, is a relatively simple procedure. The general process requires that a time measurement be taken for multiple iterations of the function in question and, then, that the time measurement be translated into seconds per iteration, or whatever.

Take a look at the following routine.

```
01♦LBL "TFUNK"  11 RCLSW
02 .1           12 1 E2
03 STO 12       13 *
04 0            14 FRC
05 SETSW        15 LASTX
06 RUNSW        16 INT
07♦LBL 00       17 .6
08 ISG 12       18 *
09 GTO 00       19 +
10 STOPSW       20 END
```

This routine, "TFUNK," contains all of the fundamentals for timing a function. Lines 02 and 03 initialize register 12 (our loop control register) for 100 iterations. Lines 04, 05, and 06 initialize and start the time measurement. Lines 07, 08, and 09 are the iteration control loop. Line 10 stops the time measurement. Lines 11 through 19 display the measured time interval in seconds per iteration. This is just a skeleton routine. You still have to calibrate it and make some other adjustments (read on) in order to use it.

Now, key-in the routine, "TFUNK." Press [GTO] [.] [.] to pack it, and then execute TFUNK. FIX 4 and note the number in the display (it should be around .100 if nothing but the Time Module is connected to your HP-41). This is the average time, in seconds (100 ms), that it took each iteration of the first 100 iterations of the control loop. Press [R/S]. Note that the resulting average time for each of the second 100 iterations is less than that of the first 100 iterations. Succeeding runs result in a displayed time that varies by no more than a few tenths of a millisecond. The reason the first run gives a greater time is because of the somewhat lengthy compiling procedure that takes place during the first iteration of the loop. (See KEY NOTES V5N2p7c.) After making changes or PACKING a function-timing routine, the results of the first run should be disregarded.

Now, modify TFUNK so it gives a result that is close to zero. Just add, to the end, two lines that subtract the number that resulted when you ran the routine the second time in the last paragraph. They will be something like: .0995; −, or .0970; −. PACK the routine and run it twice to verify that the result you get is small (no bigger than a few tenths of a millisecond). With all this completed, we are ready to time some functions.

To time the function X<>Y, place the line X<>Y after line 07 in TFUNK. TFUNK will now look like this:

```
01♦LBL "TFUNK"  13 1 E2
02 .1           14 *
03 STO 12       15 FRC
04 0            16 LASTX
05 SETSW        17 INT
06 RUNSW        18 .6
07♦LBL 00       19 *
08 X<>Y         20 +
09 ISG 12       21 .0994
10 GTO 00       22 -
11 STOPSW       23 END
12 RCLSW
```

where the number in line 21 is the time that resulted when you ran the skeleton routine.

PACK the routine (XEQ "PACK") and run it (XEQ "TFUNK) twice. You will find that the X<>Y function requires about 10 ms. That is, "about 10 ms" with nothing but the Time Module connected to the HP-41, and with the batteries in fairly good condition, and at a temperature of about 295 degrees Kelvin*, etc.

You can now replace X<>Y with any function or sequence of functions whose execution times are not number-dependent. Such functions as R↑, RDN, ENTER, LASTX, etc. fall into this category.

Such functions as SIN, LN, FACT, etc. have execution times that depend on the parameters. There are tests and traps in the microcode of these functions that may cause the result for one parameter to be resolved quicker than the result for another parameter.

To time a function whose execution time is number-dependent, like SIN or LN, you must have a skeleton routine that sets-up the stack so that the function always operates on the same number. To time the SIN function, start with a routine like this:

```
01♦LBL "TFUNK"  13 1 E2
02 .1           14 *
03 STO 12       15 FRC
04 0            16 LASTX
05 SETSW        17 INT
06 RUNSW        18 .6
07♦LBL 00       19 *
08 RCL 00       20 +
09 ISG 12       21 .1210
10 GTO 00       22 -
11 STOPSW       23 END
12 RCLSW
```

Adjust the number in line 21 so that running this routine in your HP-41 gives a result close to zero (the same as we did above). Register 00 will store a constant parameter. Key-in SIN after the RCL 00 in line 08. XEQ "DEG." Store 30 in register 00. XEQ "PACK" [RTN] [R/S]. When it stops, run it again. The result will be the amount of time that your HP-41 takes to calculate the SIN of 30 degrees. Store 90 in register 00 and press [R/S]. The result this time will probably be smaller. The HP-41 uses a different path through the microcode to obtain the SIN of 90 degrees.

You can see that compiling a list of the *precise* execution times for all of the HP-41 functions would be an improbable task.

However, with your new Time Module, you now have the tools to determine relative execution times, and to approximate execution times for a given system configuration.

Methods for timing routines and functions such as the ones shown above are handy tools to help you shorten the execution times of lengthy or iterative programs. If, in a loop that iterates 100 times, you can replace the sequence: 2; *; with ST+ X, you trim about 6 seconds from the execution time. Now, that "doesn't sound like much, but it adds up."

**That's 22 degrees Celsius or 71.6 degrees Farenheit.*

Your Address . . . Revisited

In the last issue (V6N3p7c), we addressed the issue of keeping your address up to date. Unfortunately, there was some misunderstanding about where you should send address changes and corrections.

If you live in the United States or Canada, you should send your address changes and corrections to the Corvallis Users' Library. If you are a member of the Corvallis Users' Library and get KEY NOTES free as part of your membership, you must send address changes and corrections to the Corvallis Users' Library. If you live in any European countries and get KEY NOTES in an envelope that does NOT have a U.S. return address on it, DO NOT SEND YOUR ADDRESS CHANGES TO CORVALLIS. Your KEY NOTES came to you by way of the UPLE in Geneva, so send your changes and corrections to the Geneva UPLE.

For the rest of the world, if your KEY NOTES envelope does NOT have a U.S. return address on it, DO NOT SEND CHANGES AND CORRECTIONS TO CORVALLIS. Send them to the return address on the envelope in which you received your newsletter.

Back Issues . . . Revisited

ALL ORDERS FOR BACK ISSUES, IN PRINT, MUST BE SENT TO THE CORVALLIS USERS' LIBRARY, AND PAYMENT IN U.S. DOLLARS MUST ACCOMPANY YOUR ORDER. See page 19 for details and a list of available back issues still in print.

If you send your order for back issues to the Geneva UPLE you will cause a very long delay because it is time-consuming and expensive to transfer money and orders from Geneva to Corvallis. To keep your cost for back issues to an absolute minimum, they are stocked only in the Corvallis Library and mailed directly to you when you order.

For those who want a complete set of HP KEY NOTES back to V1N1 (Jan. 1977), or to replace a missing issue, The Corvallis Users' Library now offers a photocopy service at the prices listed below. An index will be furnished on request. **Remember that these are photocopies, not printed copies.** Also, the prices and any offers appearing in these or any other back issues are no longer valid.

V1N1 January 1977 (12 pages)
V1N2 June 1977 (12 pages)
V1N3 October 1977 (12 pages)
V2N1 February 1978 (12 pages)
V2N2 May 1978 (12 pages)
V2N3 August 1978 (12 pages)
V2N4 November 1978 (12 pages)
V3N1 February 1979 (12 pages)
V3N2 May 1979 (12 pages)

All prices include first-class or air mail. Payment must accompany your order and must be a check or money order in U.S. dollars, drawn on a U.S. bank. Or you may use your American Express, VISA, or MasterCard account, but be sure to include your account number and card expiration date. Make sure you mail your order and payment to the Corvallis Users' Library.

NO. OF ISSUES	U.S., MEXICO, CANADA	ALL OTHER COUNTRIES
1	$ 2.50	$ 3.00
2	4.50	5.50
3	7.00	8.00
4	9.00	10.50
5	11.00	13.00
6	13.00	15.50
7	15.50	18.00
8	17.50	20.00
9	19.50	22.00

These prices include photocopying, handling, envelope, and postage. Please allow a minimum of two weeks for delivery.

Local Label GTO's

In KEY NOTES, V5N2p6c, the "In the Key of HP" article discusses local-labels and the compiling (or "jump distance recording") of local-label GTO statements. **Paul Boltwood** of Stittsville, Ontario, Canada requested a more thorough discussion of compiled GTO statements. Thus, we have the following.

Upon the first execution of a local-label GTO statement, the HP-41 searches (line-by-line, starting at the GTO statement) for the local-label destination. Once found, the location of the local-label, relative to the calling GTO statement, is recorded with that GTO statement so that future executions will not require a line-by-line search. This process of recording local-label locations with their calling GTO's is known as *compiling*.

Whenever a program is PACKed to eliminate null bytes or expanded to allow for the addition of more lines, or whenever a program line is deleted, the local-label locations recorded with the GTO statements are "erased." This process is called *decompiling*.

When a program is packed, the calculator sets a bit (a "flag" called the *PACK-bit*) in the END statement of that program. Then, on future executions of GTO . . or PACK, the calculator checks the PACK-bit to determine if the program should be PACKed. If the PACK-bit is set, then that program will not be PACKed *or decompiled*. The PACK-bit and all compiled GTO's are maintained by *Continuous Memory*, whether the calculator is on or off. The PACK-bit is cleared whenever the program is altered.

Thus, a PACKed and compiled program will not be decompiled while in main memory unless it is altered.

When a compiled program is transfered to a mass storage medium, the compiled GTO's are recorded with the program. So, when the program is read back into the machine and executed, it will run just as quickly as it did before it was recorded; no compiling process is necessary. However, because the END statement, and thus the PACK-bit, is *not* recorded with the program, the first execution of PACK or GTO . . . will decompile the program.

Another Slice of PI

Okay, okay . . . now we all know that it is possible to access the PI function without touching the PI key. It all started in the last issue of KEY NOTES (V6N3p15a) when we published "PI A La Mode." There, we presented a somewhat challenging puzzle, and that puzzle read like this:

From a Master Clear state, and without using the 0 to 9 keys or the PI key, and with the fewest possible keystrokes, how do you get pi in the display?

The two solutions that we offered to this puzzle were [XEQ] RAD [COS-1] [STO] [+] [.] [X] and [XEQ] RAD [COS-1] [ENTER] [+]; or, 12 and 10 keystrokes, respectively. Now, the true "PI in the face" came when we began receiving letters that offered the five-keystroke solution: [XEQ] [ALPHA] PI [ALPHA]. As these letters claim, this solution does satisfy the given "rules." However, we do wish that those who sent these letters had included a napkin, with which we could wipe the meringue from our eyes.

Among the letters we received on this subject was this letter from **Herbert Gudehus,** of Hamburg, West Germany. Mr. Gudehus presents a logical argument for using his "recipe" for PI.

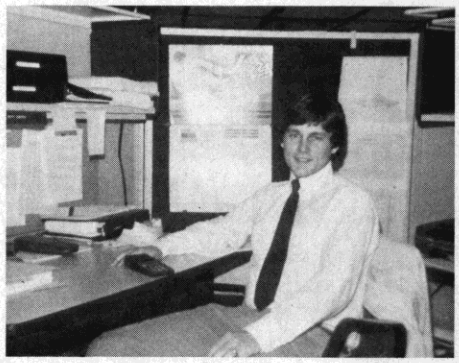
In V6N3p15 you published a nice puzzle, "PI A La Mode," and on page 16 of that issue you published two answers. Both of these answers must be supplemented by XEQ "DEG" (6 additional keystrokes) if DEGREE-mode has to be restored. Under this condition, a shorter solution is [COS-1] [XEQ] D-R [ENTER] [+] (11 keystrokes).

Another unique solution was sent to us by **Jean-Marc Delbos**, of Paris, France. Mr. Delbos' solution was this: [XEQ] RAD [Σ-] [R-P] [X<>Y] (11 keystrokes).

Thank you Mr. Gudehus and Mr. Delbos. And, thanks to everyone who sent in their solution. We only wish that we had the room to, at least, print all of your names.

Guest Article: About Program Files

Occasionally, one of your letters causes an antenna to go up to alert us that perhaps it could be the basis for an article that would be of benefit to all of our readers. *This* started as a letter, but we think it is now a very good *article*; we also think *you* will like it.



Our "guest writer" is **Jeffrey D. Smith,** a 24 years young (*his words!*) Systems Programmer from La Palma, California. He is currently employed by Southern California Edison (an electric utility) as a Systems Programmer specializing in APL (an interactive, high-level programming language) and Graphics (both interactive and batch processing). He was previously employed by I. P. Sharp Associates, Inc. (a Canadian-based APL time sharing vendor) as a Systems Programmer for APL development and maintenance. Both Mr. Smith and his spouse are native Southern Californians, and they have two daughters and a son.

In addition to APL, Mr. Smith also knows something about an HP-41. We also think he writes rather well. Here, then, is his "guest article."

While reading through several issues of KEY NOTES and other publications, it came to my attention that there are some HP-41 users who are not fully aware of the definition of "last program in memory," or of "program file." Plus, some HP-41 users do not fully understand the true behavior of certain program-loading functions, such as: RSUB, MRG, WNDLNK, etc. Yet, with the advent of the new HP-IL interface loop, precise and accurate knowledge of these terms and functions is *essential* for achieving a productive use of the HP-41 system.

First, HP-41 programs in main memory do not have names. Many users refer to the global line on line 01 as the program name, but this is not true. As the HP-41 owner's manual points out, programs in main memory are separated from each other by END statements. END statements serve to delimit "program files," and a program file is not required to have any global labels. Global labels serve only to locate program entry points and to enable the access of programs and subroutines from other programs in main memory. If a program file has two global labels, then what is its name? You may say that either name (as indicated by the labels) is valid. But, this is incorrect because the results obtained from the program usually depend on which global label is executed.

How, then, does a program file receive a name, if at all? A program file is given a name when it is stored in Extended Memory or on an HP-IL mass storage device (HP 82161A Digital Cassette Drive). The file name is entirely independent of any labels that may be inside the file, and it is used only for locating the file on the storage medium. Thus, many program files, with the same global labels, can coexist on the same storage medium. The labels only become effective when the program is loaded into main memory. In summary, global labels are used during execution, and program *names* are used for storage purposes. They are independent entities.

Now that we know how to identify and locate a program file, let's see what tools we have for loading and storing programs. There are three relatively inexpensive I/O devices: the Card Reader, the Optical Wand, and the Extended Functions/Memory Module.

CARD READER

This device allows the HP-41 to read and write unnamed program files via magnetic cards. Historically this was the first peripheral to provide mass storage. With the HP-41 positioned to a desired file in program mode, the file can be automatically recorded to cards. The Card Reader depends on the user to mark (or "name") and keep track of program files recorded on cards. The ability of the Card Reader to manipulate program files is limited, providing only two functions for loading programs — RSUB and MRG — and only one, non-programmable, function — WPRV — for writing a program file.

The WPRV (write private) function causes the current program file to be written to cards as a PRIVATE program. This prevents alteration and duplication of the program. Being non-programmable, the user must make a conscious decision to use WPRV, because its effect is irreversible.

The RSUB (read subroutine) function prompts the user for a card program that replaces the "last program file" in main memory. What is the "last program file" in main memory? It is the program file that is terminated by the permanent .END. statement. Note that after executing a GTO .., the "last program file" consists of a single statement — the permanent .END. — and every other program file is terminated by a normal END statement.

The program file that is executing RSUB (or to which the HP-41 is positioned when RSUB is executed manually) will not be replaced by the card program. If the "last program file" contains the executing RSUB, then the permanent .END. of this program will be converted to a normal END and the incoming program file will become the "last program file." Also, the conversion of the permanent .END. to a normal END will occur if RSUB is executed from the keyboard while the HP-41 is positioned to the "last program file." Therefore, the conversion of .END. to END will *always* occur if the HP-41 is positioned to the "last program file," regardless of whether RSUB is executed manually or from a running program. Usually, this is not a problem, but if you execute GTO .., and use RSUB to load a program, then a solitary END statement, that must be deleted manually, will be generated immediately preceding the new program.

Execution continues with the next statement after the RSUB function. In order to invoke the newly loaded program, the active program file must either call or branch to it via a global label. This presupposes that the newly loaded program is indeed the correct program; since card programs are not named by the calculator, there is no way for the Card Reader to validate the program. Indeed, even a successful call to a global label from the active program is no guarantee that the label is in the desired program (it could be that an older or even obsolete version of the intended program was mistakenly loaded). The problem of loading the incorrect program file is remedied by accurately marking each magnetic card. And, the solution to automatically invoking the new program lies in the MRG function.

The MRG (merge program) function causes a program file to be loaded into main memory immediately after the current instruction, replacing all program lines up to the permanent .END. statement. Notice that MRG works only for the "last program file." It cannot be used from Application Pac modules nor by private programs, because these types of program files cannot be altered, which is exactly what the MRG function is supposed to do. After the operation is complete, execution continues with the first line of the newly loaded program file. Hence, there's no need to know what global labels (if any) are needed. Because execution starts immediately after MRG (except when MRG is performed manually), the danger of loading the wrong program file becomes much more serious.

To load and automatically execute a card program, the following short routine could be used as the last program file:

```
01 LBL "RLNK" ; Read subroutine and link
02 MRG ;        Merge the card program
```

Each time another subroutine is needed, the main program can either call or branch to "RLNK" to read another program. Notice that the MRG function clears the pending return (RTN) stack. In order for the subroutine to return control to the main program, it must explicitly branch back to it via a global label. Execution will halt if RTN or .END. is encountered.

Although cards can be cumbersome, the RSUB and MRG functions provide a workable means of managing subroutines. However, both of these functions rely on the user to provide the correct program cards.

OPTICAL WAND

The Wand is an "input-only" device, as opposed to the Card Reader, which is an input and output device. The Wand provides two programmable functions — WNDSUB and WNDLNK — for loading bar-coded program files into main memory.

The WNDSUB (scan bar-coded subroutine) function operates identically to the Card Reader function RSUB, but uses bar code rather than magnetic tape as the storage medium.

The WNDLNK (scan bar-coded subroutine and link) function performs essentially the same task as the "RLNK" routine listed above. It causes the incoming program file to replace the last program file according to the same rules as WNDSUB and RSUB but, in addition, control is passed to the first line of the newly loaded program exactly as if it had been called via XEQ. There is no need to have a global label on line 01, because the HP-41 already knows where to begin execution. WNDLNK has two advantages over the "RLNK" routine: (1) the "RLNK" routine will be rendered useless if a private program file is read, whereas the WNDLNK function may be used to repeatedly load and execute PRIVATE, bar code program files; (2) the WNDLNK function preserves the pending RTN stack so that the subroutine will return control to the main program on the execution of a RTN or .END. statement.

The same problems that exist with unnamed card programs are still present with unnamed bar code programs. Both devices rely on the user to submit the correct file for loading, but otherwise they provide the same utility.

EXTENDED FUNCTIONS/ MEMORY MODULE

This new module provides many extensions to the HP-41, including Extended Memory. Extended Memory can be used by either the HP-41C or the HP-41CV, because it is slightly different than main memory.

Extended Memory is composed of *Continuous Memory* registers just like main memory, but you must use the functions that are provided by the module in order to gain access to that memory. The functions permit the creation, manipulation, and deletion of ASCII, data, and program files. All of these files must be given unique names. ASCII files are made up of characters, and

(Continued)

therefore are byte oriented. Data files are made up of complete registers (similar to data cards produced by the Card Reader, but more flexible). Program files are similar to card and bar code program files, but they must be referenced by name. The name is independent of any labels that may be present inside the program file.

There are two functions — GETP and GETSUB — provided for loading programs into main memory, and one function — SAVEP — for storing programs into Extended Memory. They are all programmable, and they represent a significant step ahead of their counterparts in the Card Reader and the Wand.

The SAVEP (save program) function is used to write a program file from main memory to Extended Memory. It requires a name for the target file in Extended Memory and some way of identifying the source program file in main memory. You can identify the source program file by specifying any global label that occurs within that program, or you can position the HP-41 to the program and then indicate that the current program file is to be saved. Whichever way you choose to indicate the source program file, you must specify a unique name for the target file. (If you specify the name of a program file that already exists in Extended Memory, it will be replaced by the source file.)

Incidentally, the SAVEP function will preserve the privacy status of the source program file in the target file. Thus, private program files may be transferred between main memory and Extended Memory. The HP-41 only prevents private program files from being transferred to an *external* storage medium.

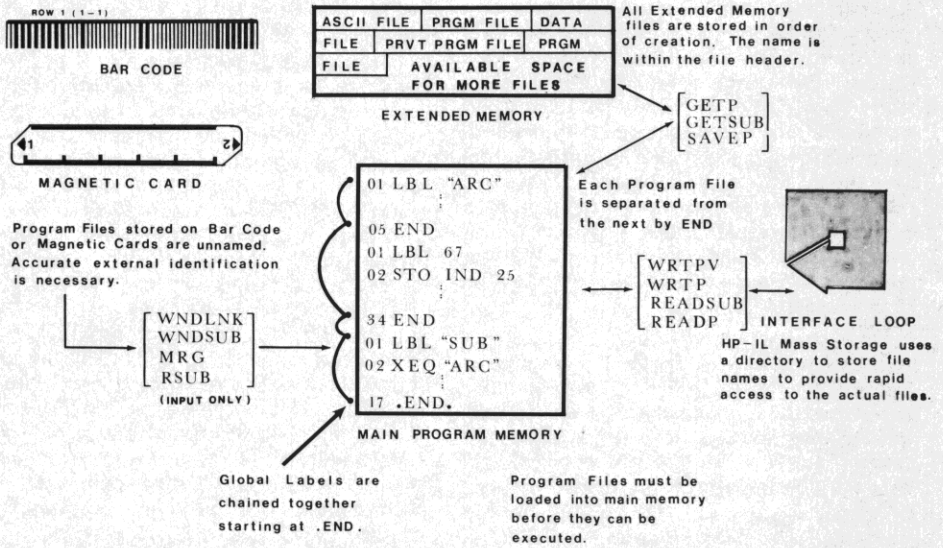
You may think that the GETSUB (get subroutine) function would be quite similar to RSUB and WNDSUB, but this is not so. The GETSUB function is actually more like the built-in HP-41 COPY function that is used to copy program files from Application Pacs into main memory. The GETSUB function always loads a program file into main memory *after* the last program file; it never overwrites the last program even if the last program contains only the permanent .END. statement. If the user isn't aware of this behavior, then repeated use of GETSUB could clutter main memory with many solitary END statements.

Why should GETSUB work differently from RSUB and WNDSUB? One reason might be that it was easier to design it that way. The GETSUB function might be "borrowing" some of the microcode used by the COPY function, thus saving the designers the trouble of reinventing the wheel. However, a closer look reveals that neither the Card Reader nor the Wand provide a means of appending programs; they only replace the last program file (with the minor exceptions mentioned earlier).

The GETP (get program) function is like WNDSUB or READSUB, and WNDLNK mixed together. The program that is loaded by GETP always replaces the last program file,

HEWLETT-PACKARD HP-41C/V

Program File Storage



even if it is the one executing GETP. If the active program is not the last program file, then it continues to run. This is the same as the RSUB and WNDSUB functions. However, if it is the last program file, then it is entirely replaced by the new program file and execution continues from the first line of the new program. In this respect, it is similar to the "RLNK" routine or the WNDLNK function and, like WNDLNK, it does not alter the pending RTN stack in any way.

Thus, Extended Memory provides an excellent means of dynamically storing and loading program files. In fact, a programmer can now design efficient overlay structures that can execute independently of user intervention (such as loading magnetic cards). A "supervisor" program can be developed that will load a program into main memory and then transfer control to the new program. When the new program terminates with RTN or .END. (or erases itself with PCLPS), control is passed back to the supervisor program, which then determines the next routine to be loaded into memory (replacing the previous program). Now, we have entered the era of "disposable" code, memory sharing, and time sharing. Attaching a mass storage device increases the possibilities.

If you are a little wary of using GETP for such a supervisor program, then you can use the GETSUB function during the initial staging of the supervisor to load an empty program file. This will ensure that a normal END exists somewhere between the supervisor and the last program file. From then on, the supervisor can use GETP without ever overlaying itself. (An empty program file can be created by keying GTO .., and then executing SAVEP, after placing a file name such as ",NULLPGM" in the ALPHA-register. The initial comma in the ALPHA-register indicates that the current file is to be saved in Extended Memory under the name "NULLPGM.")

Many advantages can be realized by dividing large, monolithic application programs into several small subprograms. For instance, the program region size can be greatly reduced, because only the portion of the application that is executing needs to be in main memory. Also, all of the advantages of "modular" programming become available. Each program file or "module" of an application package ideally performs a single, unambiguous function. If the application needs to be changed, then only the affected modules need to be edited and replaced. This eliminates having to load the entire application package into main memory (which may not be possible).

The Extended Functions/Memory Module represents a significant advance over the Card Reader and the Wand (although, not a replacement). The WNDSUB, or READSUB, and WNDLNK functions have been condensed into GETP, and a new function GETSUB has been added to permit appending program files in main memory. Also, the problem of incorrect user intervention has been eliminated.

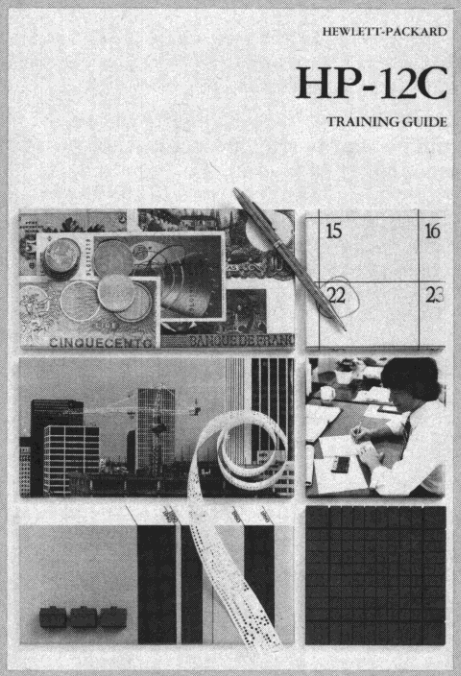
DID YOU KNOW?...

Did you know that *all* copies of KEY NOTES are printed in the U.S.A.? However, if you live in Europe or anywhere outside the U.S. and Canada, your copy actually comes from the place that shows as the return address on your envelope. So if you have problems or questions about your KEY NOTES, always direct your inquiry to whoever sent the copy to you. See page 5, lower corner, for more details.

New HP-12C Training Guide Released

Have you considered buying the new HP-12C Advanced Financial Programmable Calculator but were afraid you couldn't understand all those financial functions on the keyboard? Well, fear no more! The *HP-12C Training Guide* quickly puts the power of the HP-12C into the hands of the beginning calculator user. It very quickly dispels fears of financial calculations and programming, and it helps the novice to develop the skills that permit anyone to solve the toughest financial problems.

As a tutorial workbook, the *HP-12C Training Guide* emphasizes learning by doing. It is written in a friendly, self-paced format. Divided into 12 chapters, the training guide provides example solutions and follows each topic with review questions to ensure thorough understanding.



With over 150 pages, the *HP-12C Training Guide* takes the user from the features and basic operation of the HP-12C to financial concepts and then on to elementary programming. By making it possible to understand how to use and apply the HP-12C, the training guide not only makes the powerful financial features of the HP-12C readily available to beginners, it also serves as a refresher course for experts.

So if you are involved in financial and business calculations, yet afraid or intimidated by complex computations and equipment, then the *HP-12C Training Guide* is for you. Plus, it also makes an excellent, affordable Christmas present, and it's available now at your local authorized HP Dealer. The order number is 00012-90022. We'll also bet that it is tax-deductible, too; check with your income tax rules or your tax consultant.

Care and Feeding of Cassettes

The HP 82161A Digital Cassette Drive is a rugged, but sophisticated electronic device. As such, it is very capable and durable, but a few common-sense rules should be followed to prolong the life of the drive and the tapes it uses. The main area of concern is tape wear, but there are also some general-care procedures that will extend the life of your equipment. For example, avoid unnecessary abuse of your equipment; it is an investment that will continue to serve you for many years if common sense is applied during its use.

WEAR

Tape wear is the greatest cause of malfunction on the cassette drive. Excessive directory accesses are the main cause of early tape failure. The tape will tolerate a minimum of 500 accesses, but 1500 accesses for any particular tape is more typical. Following are a few procedures that will help extend the life of a cassette.

1. Minimize directory accesses. The directory is the most-used section of tape. Such functions as WRTP, WRTA, and DIR use the directory every time the function is executed. Continued rewriting of program files during editing is usually unnecessary and should be avoided. It is best to rewrite at the completion of a session, if possible.
2. Avoid partial data file writes. Register-by-register writes should be avoided. It is better to use WRTRX where $X > 32$. This will assure that you are storing at least one record at a time, thus minimizing the wear on any particular segment of tape.

GENERAL CARE

The tips in this section will aid in understanding the critical areas of your tapes and drive. Again, use common sense when handling tapes or using the drive.

1. Always rewind the tape. When you have completed a session, it is recommended to rewind the tape, even if it is left in the drive. This will protect the storage areas of tape from accidental mechanical damage.
2. Avoid touching the tape. The tape recording surface is very sensitive to any contact by foreign materials. Fingers or cleaning material will damage the tape. Rewinding will help avoid accidental contact with the tape.
3. Remove slack before using a tape. Tapes kept in the storage compartment will sometimes partially unwind. Tensioning the tape before inserting will avoid harsh movements by the drive. Insert an instrument, such as a pencil or pen, into the hub of the cassette, and rotate the hub until the tape appears taught inside.

CAUTION DO NOT OVER-TENSION THE TAPE.

4. Avoid temperature extremes. The inside of an automobile on a hot summer day can exceed the recommended storage temperatures. Also, sudden temperature changes can cause condensation to occur, and expansion or shrinkage of the tape.
5. Clean the head frequently. Use isopropyl alcohol, in accordance with manual instructions, to clean the head at least once a month under normal operating conditions. Be careful not to touch the tape viewing window with the cotton swab, as it will cause permanent stains.
6. Keep the drive clean. Dirt, dust, and liquids will damage your drive if they reach the mechanism inside the tape access door. Do not allow foreign matter to enter this area!
7. Avoid magnetic fields. The drive is protected, as far as possible, from radio and magnetic interference; however, placing the drive in close proximity to a strong magnetic field will cause disruption of the drive. For example, keep the drive as far as possible from a T.V. or video monitor.
8. Avoid operation with the BAT light on. The low-battery indicator is telling you to stop and recharge the battery. Follow instructions in the user's manual for the drive. Failure to do so could cause loss of data on the tape.
9. The tape drive is not a toy! Do not use the cassette door as release for nervous energy. Repeated unnecessary opening and closing of the cassette door will cause premature failure of the head cable.
10. General care of exterior. The exterior of your cassette drive is made of high-quality material that should last a lifetime if it is properly maintained. Cleaning with soap and water and a damp cloth is acceptable, as is the use of a mild plastic cleaner/polish such as "Snap." Do not use harsh household cleaners or abrasives. Cleaners containing fluorinated, or chlorinated hydrocarbons also will cause damage to the surfaces.

Again, the primary "rule of thumb" is to always use common sense when handling tapes or using the drive.

"Pioneering" With HP-41 Peripherals

The HP-41 is a real pioneer in the hand-held computer market! As such, it has its own set of nuances that often need some explanation to enable both the experienced computer buff and the neophyte to feel comfortable with its operation. The implementation of the printer functions in the HP 82160A HP-IL Interface Module fall into this category. During the design phase, it was decided that the HP-IL version should be "backward-compatible" with the dedicated HP-41 printer as much as possible. This meant that the character sets had to match to allow previously written programs to function without modification. Another very important objective was to have an ASCII-compatible printer that would allow future mainframe computers, such as the HP-75, to function properly without any modification or interpolation. As you might imagine, these two objectives are mutually exclusive, so a compromise had to be struck.

The solution entailed creation of an alternate character set (a modified HP-41 set) that would allow backward-compatibility. As stated, it is a modified set, as it was also necessary to maintain

(Continued)

the ASCII compatibility in the area of providing a carriage return and line-feed command at their respective positions of decimal 10 and 13. This cost two character locations in the set, and the chosen path to make it as transparent to the user as possible was to use character 0 to replace character 10 (it was a duplicate anyway), and put character 13 at location 124, causing, in-turn, the loss of character 124 (a|, or Sheffer, sign). Not a big loss, considering it can be easily created using either the BLDSPEC or ACCOL commands. Now, whenever a character 10 or 13 is sent, the HP-41 (actually the Interface Module) goes to a look-up table that will in-turn send either a 0 or 124, respectively, to the 82162A printer to enable the corresponding character. Additionally, if an HP 82163A Video Monitor is used, character 126 (the Σ sign) is not translated, as it does not exist; and the ~ at that location in the video interface (and the ASCII set) was deemed not an acceptable substitute, therefore a blank was substituted.

Summary: For the HP 82163A Video Interface, and the HP 82162A Thermal Printer, the following responses to HP-41 commands will occur. (SF 17 on the HP-41 prior to attempting verification of this table.)

COMMAND / CHAR CODE	DEVICE	ACCHR	ACA	OUTA	PRA
10	PR	◆	NO RESP	NO RESP	NO RESP
	VIDEO	NO RESP	LF	LF	EXT LF
13	PR	⌿	⌿	CR	⌿
	VIDEO	↑	↑	CR	EXT CR
124	PR	⌿	⌿	⌿	⌿
	VIDEO	↑	↑	↑	↑
126	PR	Σ	Σ	Σ	Σ
	VIDEO	NO RESP	NO RESP	~	NO RESP

82162A Thermal Printer Applications: Since it is likely that some of you would want to print character 124, the | sign, here are two routines that will accomplish the task.

1. Using BLDSPEC

CLX	Clear the "X" register.
ENTER	Clear the "Y" register.
BLDSPEC	
0	
BLDSPEC	
0	
BLDSPEC	Build the character.
127	
BLDSPEC	
STO 00	Store it in reg. 00 for later use.
ACSPEC	Accumulate it into print buffer.
PRBUF	PRINT IT!!!

2. Using ACCOL

3	
SKPCOL	Move column pointer to 4th column.
127	
ACCOL	Create a full column of dots at 4th dot column.
PRBUF	PRINT IT!!!

Feedback

This column contains reader feedback about articles or routines that appeared in previous issues. Though much of the information presented here is useful on its own, you will find that it is a good idea to have your library of KEY NOTES handy while reading this column.

As the production deadline for each issue of KEY NOTES approaches, those of us on the KEY NOTES staff begin pushing things aside in favor of getting KEY NOTES to the printer. This letter was pushed aside during the production of the last issue, and it shouldn't have been. The letter and routine are from **Mike Edwards,** of Urbana, Illinois.

(V6N3p13a) I regret that I sent you an inaccurate routine (FLEFT) a few days ago. It doesn't allow for the overhead bytes contained in ASCII files. This revised routine has been tested, and works under the following assumptions (see page 24 of the *Extended Functions/Memory Module Owner's Manual*): (1) there is one unusable byte in every ASCII Extended Memory file, plus (2) one unusable byte for each record within the file. I suppose (1) is an end-of-file pointer and (2) is a record length indicator. This routine calculates the number of available bytes within an ASCII Extended Memory file.

```
01♦LBL "FLEFT"      21 FC?C 25
02 FIX 0            22 GTO 01
03 CF 29            23 ALENG
04 "FILE NAME?"     24 -
05 AON              25 FS? 17
06 STOP             26 GTO 00
07 AOFF             27 1
08 0                28 -
09 SEEKPTA          29 GTO 00
10 FLSIZE           30♦LBL 01
11 ASTO Y           31 CLA
12 ASHF             32 ARCL X
13 ASTO Z           33 "├ BYTES LEFT: "
14 7                34 ARCL Y
15 *                35 ARCL Z
16 1                36 AVIEW
17 -                37 FIX 4
18♦LBL 00           38 SF 29
19 SF 25            39 END
20 GETREC
```

Here's a "Feedback" letter that brings back many memories to long-time KEY NOTES readers. Because of that and because we do not want to again start a flood of factorial routines, we are publishing it with the warning that there are many, many ways to compute factorials of numbers greater than 69 . . . and we've probably seen all of them . . .. This routine was contributed by two people, **Kent Krumvieda** and **Mike Daniels,** of Boulder Colorado.

(V2N3p9c) Recently, we were going through our old KEY NOTES and we noticed the HP-97 routine, written by **James Grandstaff,** that computes factorials of numbers larger than 69. We decided to rewrite it using the capabilities of the HP-41. The following is a listing of this revision.

```
01♦LBL "FAC"        19 LOG
02 STO 01           20 ST+ 00
03 70               21 DSE 01
04 X<=Y?            22 GTO 00
05 GTO 01           23♦LBL 02
06 X<>Y             24 RCL 00
07 FACT             25 INT
08 RTN              26 LASTX
09♦LBL 01           27 FRC
10 .069             28 10↑X
11 ST+ 01           29 CLA
12 69               30 ARCL X
13 FACT             31 "├ E"
14 LOG              32 FIX 0
15 STO 00           33 ARCL Y
16♦LBL 00           34 AVIEW
17 RCL 01           35 FIX 5
18 INT              36 END
```

Kongsberg is a town of about 20,000 people, just south of Drammen in Norway. It is the home of **John Erik Setsaas,** and this is his tip.

(V5N3p14b) Sometimes you need a routine that changes a number into a corresponding character. For example: **Fred Scheifele's** routine requires that 26 characters are stored, one in each register. This occupies 182 bytes of program memory. If you have a printer available, there is a much shorter way to do this.

```
01♦LBL "D-C"
02 0
03 X<>Y
04 BLDSPEC
05 END
```

Just put the number of the character (according to the list on page 37 in the printer handbook) into the X-register, and execute "D-C." The character will then be in the X-register.

Next, here's a nice letter from **Axel Harvey,** of La Macaza, Canada.

(V6N2p14b) In the last issue your editor suggests assigning the functions of the top two rows on the HP-41 to themselves: X<>Y to the X<>Y key, RDN to the RDN key, and so on. It's well worth repeating. This is the best way to avoid waiting forever for a RDN when your program pointer is in the middle of an undocumented 850-byte masterpiece (you wish to inspect the stack without losing your place), and it's 2 A.M. (At 2 A.M. the 4-second wait is forever!)

Of course, a simpler way is to get out of USER mode. The trick here is to remember to get back into it after you have seen the stack, so that your next keystroke won't be a disaster. This is not always possible at 2 A.M.

In the last issue of KEY NOTES (V6N3p13a), we printed what we claimed to be the shortest sequence of program lines to round-up in absolute value any number. This sequence was — INT; LAST X; FRC; X=0?; SIGN; +. Since then, we have received several inputs claiming to be shorter versions of this sequence, but these only made us realize that we hadn't done a very good job of explaining what we meant by "rounding-up a number." What we meant is this: if an integer (whole number) is input, the sequence will return that integer; otherwise, if the input is not an integer, it will be rounded to the next (greater in absolute value) integer. Thus, an input of 0 will return 0, an input of 3 will return 3, and an input of −3 will return −3. And, an input of .0004 will return 1, an input of −.002 will return −1, an input of 3.67 will return 4, and an input of −4.033 will return −5. Though we thank everyone who sent in a "shorter version," we have yet to receive a shorter version — than ours — that fulfills the above description of "round-up."

Now, we bring you some adaptably useful feedback from **Lawrence Vassallo,** who lives in Rochester, Michigan.

(V6N3p12c) In response to **Dr. Keith Bernstein's** submission to "Routines, Techniques, Tips, Etc . . .," I have enclosed my own HP-41 subroutine "PRC" for producing column-formatted printer output without use of the HP 82180A Extended Functions/Memory Module.

```
01♦LBL "PRC"    16 SKPCHR
02 RND          17 RDN
03 CF 00        18 CLA
04 X<0?         19 ARCL X
05 SF 00        20 FS?C 00
06 ABS          21 GTO 00
07 ENTER↑       22 "├ "
08 X≠0?         23 ACA
09 LOG          24 RTN
10 INT          25♦LBL 00
11 X>0?         26 CHS
12 ST- Z        27 "├-"
13 X<> Z        28 ACA
14 1            29 END
15 -
```

To call "PRC," the main program must first set the proper number of decimal places (FIX #) and clear flag 29. Then, the main program must place the number of digits to the left of the decimal point in the Y-register, the number to be output in the X-register, and XEQ "PRC." The value in the X-register will be accumulated into the print buffer with a trailing sign. PRC may be called multiple times before printing the formatted line with PRBUF or ADV. The method used for formatting is not iterative and requires only the stack registers, the ALPHA-register, and temporary use of flag 00. The X-register contents are preserved.

Here are some comments about the last issue of HP KEY NOTES from **Jeffrey Smith,** of La Palma, California.

(V6N3) Richard Partridge's "HP-35" display simulator (V6N3p11c) fails for values that overflow or underflow into scientific or engineering notation (or if the calculator is already set to SCI or ENG). It's limitations should be clearly understood by anyone who wishes to use it to obtain "suppressed zero" formatting.

Ed Keefe's method of establishing FIX/ENG notation (V6N3p12a) has two flaws: (1) The flag range of 41.043 should have only two significant figures in the fractional portion (that is: 41.43), and (2) since only flag 41 needs to be set, a value of just 41 is sufficient for a range specification.

(The modified sequence would be: ENG n; RCLFLAG; FIX n; 41; STOFLAG — Ed.)

Routines, Techniques, Tips, Etc . . .

The routines and techniques furnished in this column are contributed by people from all walks of life and with various levels of mathematical and programming skills. While the routines might not be the ultimate in programming, they do present new ideas and solutions that others have found for their applications. *You might have to modify them to fit your personal application.*

The Extended Functions/Memory Module has quickly become a strikingly popular addition to the HP-41 system. In Houston, Texas, **Bill Rudersdorf** is using his Extended-Functions/Memory Module to make life just a little bit easier for all of us.

(41) Here's a quick and useful routine for finding the number of bytes in a program. It illustrates a few uses of the X-Functions Module. I've been using an HP-41 for almost three years now.

```
01♦LBL "BYTES"  16 FC?C 25
02 FIX 0        17 GTO 00
03 SF 21        18♦LBL 02
04 CF 29        19 SF 25
05♦LBL 00       20 RCLPTA
06 CF 00        21 FC?C 25
07 "NAME?"      22 GTO 01
08 AON          23 FS?C 00
09 PROMPT       24 PURFL
10 AOFF         25 "├:"
11 GTO 02       26 ARCL X
12♦LBL 01       27 "├ BYTES"
13 SF 00        28 AVIEW
14 SF 25        29 GTO 00
15 SAVEP        30 END
```

Near the border of South Australia and New South Wales, on the banks of the Murray River, lies the small town of Renmark. And, in this small town, we know of at least one person, **Chris Tolley,** who owns an HP-41, a printer, and an Extended Functions/Memory Module. Here's Mr. Tolley's contribution.

(41) Here is a routine to print the contents of an ASCII file matched, record to register, to the contents of a data file. The output will appear on one line, assuming the total number of characters in the ASCII record and the data register does not exceed 21 (10 if flag 12 is set). Execution will halt when the END of either file is reached.

```
01♦LBL "MATCH"  20 CLA
02 0            21 ARCL 01
03 STO 00       22 RCL 00
04 STO 01       23 SEEKPTA
05 STO 02       24 GETREC
06 "FILE MATCH" 25 ACA
07 PRA          26 2
08 ADV          27 SKPCHR
09 "ASCII FILE?" 28 CLA
10 AON          29 ARCL 02
11 PROMPT       30 RCL 00
12 ASTO 01      31 SEEKPTA
13 AOFF         32 GETX
14 "DATA FILE?" 33 ACX
15 AON          34 ADV
16 PROMPT       35 ISG 00
17 ASTO 02      36 X<> X
18 AOFF         37 GTO 01
19♦LBL 01       38 END
```

Here's another impressive routine that requires the Extended Functions Module. It was contributed by **Harold Schumann,** of Munster, Germany.

(41) The new Extended Function/Memory Module is really an incredibly powerful means for advanced programming. Recently, I have written a short routine that replaces single characters in the ALPHA-register by other single characters. It can easily be seen that the application of Extended Functions facilitates this task.

```
01♦LBL "REPA"   11 XTOA
02♦LBL 01       12 X<> Z
03 AVIEW        13 1
04 ENTER↑       14 +
05 POSA         15 CHS
06 X<0?         16 AROT
07 RTN          17 RDN
08 AROT         18 GTO 01
09 ATOX         19 END
10 R↑
```

How to use REPA:

1. Place the equivalent code of the character to be replaced into the X-register and the code of the replacing character into the Y-register.

(Continued)

2. Place any string into the ALPHA-register.

3. XEQ "REPA"

The "old" character will be replaced by the "new" one at each occurrence in the string.

(If, instead of the numerical character codes, you would rather ASTO the respective ALPHA characters in the X- and Y-registers, this routine will still work. You'll notice, Mr. Schumann, that I replaced the global GTO "REPA" that you had at the end of this routine with a local GTO 01 and LBL 01 at line 02. This conserves bytes and search time — Ed.)

Not far from Munster, Germany, on the north coast of Germany, is a town called Cuxhaven. Cuxhaven is the home of **Gunter Merten,** another owner of an Extended Functions/Memory Module.

(41) Following the operation GETKEY, I needed the number 0, 1, 2,..., 8, 9 in the X-register instead of the key code for the keys 0 to 9. I found two short, quick routines to implement this. The first (labeled KEYNUM) returns a −1 to the X-register if any key besides keys 0 through 9 is pressed. The second (labeled KEYNM) is shorter but it doesn't account for the user pressing a "non-number" key.

```
01♦LBL "KEYNUM"
02 "RHIJ)?"
03 64
04 XTOA
05 "├456"
06 GETKEY
07 POSA
08 END
```

```
01♦LBL "KEYNM"
02 GETKEY
03 45
04 MOD
05 13
06 MOD
07 11
08 MOD
09 END
```

(The "goose" will pause in the display during the execution of these routines. It is just waiting for you to press a key — Ed.)

This next routine is being printed with the permission of **Richard Nelson** of PPC*. The routine was published on the last page of the *PPC Calculator Journal* V9N4, and it struck us as being pretty clever.

(41) Recall sigma is a routine to recall the SIGMA-X and SIGMA-Y values from the statistical registers when the location of these registers is unknown. Lines 02 through 05 can be omitted if n is still in the X-register from a previous SIGMA+ or SIGMA−.

```
01♦LBL "RΣ"
02 CLST
03 Σ+
04 CLST
05 Σ-
06 MEAN
07 LASTX
08 ST* Z
09 *
10 END
```

**Founded in June 1974 by Richard J. Nelson, PPC is the world's first and largest organization dedicated to Personal Programmable Calculators. The Club is a volunteer, non-profit, loosely organized, independent, worldwide group of Hewlett-Packard personal programmable calculator users. PPC is not sponsored, nor in any way officially sanctioned, by Hewlett-Packard.*

For more information about PPC and a sample issue of the Club's newsletter, send a self-addressed, large (folded) envelope (9 X 12 inches; 23.8 X 30.5 cm) with first-class postage for 2 ounces (56.7 grams) to: PPC Calculator Journal; 2545 W. Camden Place; Santa Ana, California 92704 U.S.A. If you live outside the U.S., make sure you include a legible address label and international postal coupons for 56.7 grams (2 ounces). A letter is not necessary and will only slow the response.

What is the biggest number you can think of? What is the biggest number your HP-41 can think of? Is it around 10^{100}? If "yes" is your answer, **Thomas Excelsior Valere,** of Paris, France, may beg to differ.

(41) Sometimes you need to calculate values for large exponents like 41^{67}, but $41^{67} > 10^{100}$, so the HP-41 cannot compute it with the Y^X function. So, try this lifesaving routine based on the formula $Y^X = 10^{Y \text{ LOG } X}$. It works just like the Y^X function, except that x is not stored in the LASTX-register. The resulting mantissa will be in the X-register, and the respective exponent in the Y-register.

```
01♦LBL "Y**X"
02 X<>Y
03 LOG
04 *
05 INT
06 LASTX
07 FRC
08 10↑X
09 END
```

(This also works for negative exponents. For example, 86^{-85}. But the true question is: excepting the human imagination, is there anything in the universe that demands a number greater than 10^{100} or less than 10^{-100}? And, if there is, perhaps we should switch to an HP-75C, which flaunts a numerical range from greater than 10^{-500} to less than 10^{500} — Ed.)

Peter Rushworth, who lives in Lakewood, Colorado, contributed this routine to KEY NOTES. It is a "sort of humorous" application of the Time Module.

(41) Here is an HP-41 routine for use with the Time Module. This program will display the increasing labor costs of a business meeting, given the number of people and their labor rate in dollars per hour. The routine requires a minimum SIZE of 003.

```
01♦LBL "MT"
02 0
03 STO 00
04 STO 01
05 SF 27
06 CF 21
07 FIX 0
08♦LBL 00
09 CF 22
10 "NO. PEOPLE?"
11 PROMPT
12 FC?C 22
13 GTO 01
14 "RATE?"
15 PROMPT
16 *
17 ST+ 01
18 GTO 00
19♦LBL 01
20 FIX 2
21 "    READY"
22 AVIEW
23♦LBL 02
24 STOP
25 GTO 02
26♦LBL A
27 CF 27
28 TIME
29 STO 02
30♦LBL 10
31 TIME
32 RCL 02
33 HMS-
34 HR
35 RCL 01
36 *
37 "   $"
38 ARCL X
39 AVIEW
40 GTO 10
41 END
```

Example Problem:

Number of People	Labor Rate ($/hr)
3	8.50
5	12.75
1	17.50

Solution:

Input	Display
XEQ "MT"	NO. PEOPLE?
3 [R/S]	RATE?
8.50 [R/S]	NO. PEOPLE?
5 [R/S]	RATE?
12.75 [R/S]	NO. PEOPLE?
1 [R/S]	RATE?
17.50 [R/S]	NO. PEOPLE?
[R/S]	READY

At the start of the meeting, press [A] and the cost of the meeting will accumulate in the display.

C. Lamar Williams lives in San Jose, California. Lately, Mr. Williams has been thinking of ways that we can save money.

(41) Budget minded grocery-shoppers must repeatedly choose the least costly (lowest unit cost) of equivalent items. The classic decision is exemplified by: "13 ounces at $1.63 for brand A, or 17 ounces at $2.15 for brand B — which is cheaper?" And, it is easily solved by the routine "SHOP." Just execute "SHOP," at the prompt: A=?, ENTER the cost (1.63 [ENTER]), key-in the quantity (13), and press R/S. Do the same at the prompt: B=?. The appropriate answer (A IS CHEAPER, B IS CHEAPER, OR A = B) will be displayed. Try it! Note that A IS CHEAPER for the above example.

```
01♦LBL "SHOP"
02 "A"
03 XEQ 00
04 XEQ 00
05 X>Y?
06 "A"
07 "├ IS CHEAPER"
08 X=Y?
09 "A = B"
10 AVIEW
11 RTN
12♦LBL 00
13 "├=? $↑QUAN"
14 PROMPT
15 /
16 "B"
17 END
```

Many HP-41 owners have an interest in music. This routine was submitted to KEY NOTES by one of these music-loving readers. **Paulo de Salles Mourao** lives in Belo Horizonte, Brazil and this is his routine.

(41) In my spare time (and when not programming), I like to play the violin (for internal consumption, for that matter). Today I was considering adjusting my tempo, when I realized that my daughter had borrowed

my metronome some days ago. But, I am an HP-41 owner, and this type of person has many resources. Why not write a metronome program?

```
01♦LBL "METR"   10♦LBL 02
02 "INDEX"      11 DSE 01
03 PROMPT       12 GTO 02
04 X=0?         13 TONE 9
05 GTO 03       14 GTO 01
06 STO 00       15♦LBL 03
07♦LBL 01       16 TONE 9
08 RCL 00       17 GTO 03
09 STO 01       18 END
```

On XEQ "METR", you have access to the "INDEX" entry. Zero is *vivace,* 1 is *vivace non troppo,* 4 is *allegro,* 7 is *adante,* and 10 is *moderato* (incidentally, the goose flies backwards), but the nuances I leave to the *maestros.* The routine is stopped manually ([R/S]). It is another unusual application of the HP-41, *n'est-ce pas*?

Along these same lines, we have this routine that was contributed by **Tom Nguyen,** of Issaquah, Washington. This routine can be used on any HP-41.

(41) This routine is used when transposing music from one key to another according to the interval of half-steps specified by the user. The interval is specified only once for successive notes. The notation used by the program is best explained by referring to the piano keyboard. The white keys are represented by capital letters (C, D, E, F, G, A, B,) and the black keys are represented by the small letter that corresponds to the flat (not sharp) note of that key. Thus, D-flat, E-flat, G-flat, A-flat, and B-flat are represented by d, e, g, a, and b respectively. (The ALPHA character "9" is used for a small g.)

```
01♦LBL "TR"     23 ASTO 11
02 "A"          24 "a"
03 ASTO 01      25 ASTO 12
04 "b"          26♦LBL 05
05 ASTO 02      27 12
06 "B"          28 "INTERVAL"
07 ASTO 03      29 PROMPT
08 "C"          30 X<=Y?
09 ASTO 04      31 GTO 04
10 "d"          32 GTO 05
11 ASTO 05      33♦LBL 04
12 "D"          34 STO 00
13 ASTO 06      35♦LBL 03
14 "e"          36 "NOTE:"
15 ASTO 07      37 AON
16 "E"          38 STOP
17 ASTO 08      39 ASTO 14
18 "F"          40 AOFF
19 ASTO 09      41 1.012
20 "9"          42 STO 13
21 ASTO 10      43♦LBL 01
22 "G"          44 RCL 14
45 RCL IND 13   54 RCL 00
46 X=Y?         55 +
47 GTO 02       56 12
48 ISG 13       57 X<Y?
49 GTO 01       58 ST- Y
50 GTO 03       59 RCL IND Y
51♦LBL 02       60 PSE
52 RCL 13       61 GTO 03
53 INT          62 END
```

The specified interval always must be positive. For example, to transpose from the key of G to the key of C, specify an interval of 5 (the number of half-steps between the roots of the two scales). At the prompt "NOTE:," key in the letter that corresponds to the note you wish to shift. Shifting a note down an interval X is the same as shifting it up an interval 12 - X.

Jim Boardman, of Tucson, Arizona, sent us this next routine and with his letter he included the following P.S.: "I am an avid reader of HP KEY NOTES and I feel as though I do a lot of *taking* of information from its pages. Here, hopefully, I have done a small amount of *giving.*" Yes you have, Mr. Boardman, and here is your contribution.

(41) Occasionally, I find a need for saving the current date and time. The Time Module is very handy for providing this data. Normally, two registers are required for saving the information. But, by making use of two other Time Module functions, the data can be packed into one register and later re-expanded into two registers.

```
01♦LBL "PAKDTM" 13 RTN
02♦LBL 00       14♦LBL "RESDTM"
03 23.5958      15 FRC
04 TIME         16 1.01198
05 X>Y?         17 LASTX
06 GTO 00       18 INT
07 1 E2         19 DATE+
08 /            20 X<>Y
09 1.01198      21 1 E2
10 DATE         22 *
11 DDAYS        23 END
12 +
```

"PAKDTM" will pack the current date and time into the X-register. A routine calling "RESDTM" with such a packed number in X will receive back the original date in Y (in the form MM.DDYYYY or DD.MMYYYY) and the original time in X (in the form HH.MMSS). If you are not concerned about the date changing at midnight while "PAKDTM" is executing, lines 02, 04, and 05 of that routine may be deleted.

(You'll note, Mr. Boardman, that we replaced that 8-byte GTO "PAKDTM" with a GTO 00 (2 bytes) and we added a LBL 00 (1 byte) after LBL "PAKDTM", thus, conserving 5 bytes. For more information on this, see "Add a Local Label," KEY NOTES V6N1p7b — Ed.)

Aston, Pennsylvania, is the home of **Vasant Patel,** who sent us this next contribution. The routine will be of interest to those who are developing programs that solve systems of linear equations.

(41) Here is a short, efficient subroutine for solving n simultaneous equations using the Gauss-Jordan method. Prior to accessing this routine and starting at register 21, the n coefficients are to be stored row-by-row, with each row followed by its corresponding constant. You can change the starting register by changing line 04. The value of n must be contained in register 00. At the end of execution, the constant vector is replaced by the solution. Note that in addition to the matrix elements, registers 01 through 08 are altered.

```
01♦LBL "LIN"    31 RCL 02
02 RCL 00       32 +
03 STO 07       33 STO 03
04 21           34 RCL 01
05 STO 08       35 X=Y?
06 +            36 GTO 15
07 STO 04       37♦LBL 14
08♦LBL 11       38 RCL IND 01
09 RCL 04       39 RCL IND 02
10 RCL 07       40 *
11 -            41 ST- IND 03
12 RCL 04       42 1
13 3            43 ST- 01
14 10↑X         44 ST- 03
15 /            45 DSE 05
16 +            46 GTO 14
17 RCL IND X    47♦LBL 15
18♦LBL 12       48 RCL 00
19 ST/ IND Y    49 1
20 ISG Y        50 +
21 GTO 12       51 ST+ 02
22 RCL 08       52 DSE 06
23 STO 02       53 GTO 13
24 RCL 00       54 ST+ 04
25 STO 06       55 1
26♦LBL 13       56 ST+ 08
27 RCL 04       57 DSE 07
28 STO 01       58 GTO 11
29 RCL 07       59 END
30 STO 05
```

Now, here's a tip that was sent to us by **Paul Murteira,** of Lisboa, the capital of Portugal. This is a work-conserving suggestion.

(41 and Card Reader) I have a suggestion that might help users to find how many tracks will be needed to record a program onto cards. (1) Assign PACK and WPRV to HP-41 keys. (2) Key-in the program and leave the HP-41 in program mode. (3) Press [PACK] and then [WPRV] in user mode, and the display will show "RDY kk of nn." (4) If you don't want to record in private, simply press the backarrow key, then insert the cards to record your program.

(Continued)

If you want to reduce the number of tracks required, then shorten a long ALPHA message or shorten any long ALPHA labels, and repeat from step (3). This procedure will conserve your batteries because you won't have to insert a card to find how many tracks are needed.

For example, solve the two equations:

2x + 6y = 27

5x + 9y = 13

Store, in registers 21 through 26, the numbers 2, 6, 27, 5, 9, and 13 in that order. Also, store 2 in register 00. After executing the routine, the solution to x will be found in register 23 and the solution to y will be found in register 26.

(This routine is, just as Mr. Patel says, short and efficient. It does not rearrange the rows of the matrix to obtain the largest absolute pivot values and, thus, it does not minimize rounding error. It does not test for singularities or estimate the numerical condition of the matrix. So, any program that calls this as a subroutine should take these things into account, beforehand — Ed.)

Now, we have a fine contribution from **John Chaffer**, of Seattle, Washington. Mr. Chaffer presents a great idea that many of you will be able to build on.

(41) Here is a short routine I wrote to print a single byte of "WNDSCN"-type barcode on the HP 82143A (non-HP-IL) printer. Decimal integers from zero to 255 (with no checksum) can be printed in barcode. Black printer paper is needed, and the darker settings seem to increase the reading reliability. The barcode is printed so that the stop-bits come before the start-bits, but the Wand doesn't care. To view the results, use the routine "SCN."

```
01♦LBL "BC1"     21 2
02♦LBL 00        22 *
03 1.008         23 X≠0?
04 STO 00        24 SF 00
05 ADV           25 XEQ 01
06 SF 12         26 ISG 00
07 CF 00         27 GTO 02
08 "NUMBER?"     28 XEQ 01
09 PROMPT        29 XEQ 01
10 ENTER↑        30 GTO 00
11 XEQ 01        31♦LBL 01
12 SF 00         32 127
13 ENTER↑        33 ACCOL
14 XEQ 01        34 FS?C 00
15♦LBL 02        35 ACCOL
16 2             36 1
17 /             37 SKPCOL
18 INT           38 RCL T
19 LASTX         39 RTN
20 FRC           40 END
```

```
01♦LBL "SCN"     05 VIEW 01
02♦LBL 00        06 PSE
03 CLD           07 GTO 00
04 WNDSCN        08 END
```

Surely, the esthetics of the next routine will brighten your day. It makes good use of the "input during pause" feature of the HP-41. This routine was contributed by **Jean-Luc Marechal**, of Liege, Belgium.

(41) This idea could help some KEY NOTES readers. It is a short routine that can be used in place of PROMPT or STOP. This routine tests to see if the input is to be an ALPHA or a numerical input (checks ALPHA-mode flag 48). Then, after pausing for the input, it returns to the main program.

```
01♦LBL "IN"      08♦LBL 00
02 23            09 PSE
03 FC? 48        10 FC? IND L
04 22            11 GTO 00
05 STO L         12 AOFF
06 CF IND L      13 END
07 AVIEW
```

New Orleans, Louisiana, is a city known for its high-energy festivities. And no doubt, part of the celebrant atmosphere found in this city is attributable to the elated HP-41 owners who live there. **Edward Scheinuk** is an HP-41 owner who lives in New Orleans, and this is his contribution.

(41) Having gone through the experience of having someone erase a brilliantly conceived program — "I just pressed a few keys, honest" — that was not yet recorded on cards, I am fully aware of the need for "guard" routines as mentioned in V6N3p11. Thus, I would like to contribute my own version. It requires an HP 82182A Time Module.

```
01♦LBL "Z"       01♦LBL "ZZ"
02 SF 11         02 DATE
03 "↑↑ZZ"        03 DOW
04 CLST          04 X≠Y?
05 .0004         05 GTO "Z"
06 OFF           06 END
07 TIME
08 +
09 XYZALM
10 END
```

After "Z" has been executed to turn-off the HP-41, the user must key-in the current DOW (day of week) within a short time after pressing the [ON] key or the HP-41 will not remain ON. The time interval can be adjusted by changing line 05 (I suggest starting with a 4-second time interval). There are only three methods (that I know of) of circumventing the workings of the routine and all three would be obvious only to someone who is experienced in the use of the HP-41.

Fredrikstad, in southern Norway, is the origin of this routine by **Hans Aspenberg**. The subject addressed by this routine is "sort-of" popular.

(41) Here is a cute sorting routine for the HP-41. Key-in the block of data-registers to be sorted using the format bbb.eee, where bbb is the begin-address and eee is the end-address of the block. Then XEQ "SORT." Note that bbb > = 3 because the routine uses registers 00, 01, and 02 for pointers. SORT is short, using only 49 bytes, and it is surprisingly fast!

```
01♦LBL "SORT"    13 GTO 02
02 STO 00        14 RCL 01
03♦LBL 00        15 STO 02
04 RCL 00        16♦LBL 02
05 STO 02        17 ISG 01
06 1.001         18 GTO 01
07 +             19 RCL IND 00
08 STO 01        20 X<> IND 02
09♦LBL 01        21 STO IND 00
10 RCL IND 02    22 ISG 00
11 RCL IND 01    23 GTO 00
12 X<=Y?         24 END
```

(The bubble sort — the method used by this routine — is the traditionally chosen method used on computers for sorting data. An alternative method that is rarely used in practice, utilizes, for each successive element, a binary-search to locate the position of that element in the previously ordered elements. This binary-search method greatly reduces the number of comparisons involved. However, on most computers the additional matrix-index management required to insert an unordered element into its proper position in the previously ordered elements is extensive. It is for this reason that the binary-search method has been neglected. But now the REGMOVE function in the Extended Functions/Memory Module makes it easy to insert a value anywhere in a block of registers. Thus, at the risk of dropping a hint, perhaps the REGMOVE function makes the binary-search method a viable method for sorting data on the HP-41 — Ed.)

Language Name Delayed . . .

Many of you are on the edge of your seat just waiting to see what name we choose for the HP-41's language and to see who won the contest (V6N3p16a), but we have some bad news for you. Because some of the most popular names we chose were already being used by other companies, and because we want to make sure that there is no infringement on trade marks or trade names, we are delaying the announcement of a winner until the next issue of KEY NOTES, which will be mailed the third week of February 1983. So hang onto your seat until then!

NEW HP-75C PAGES

Turn to page 16 for the first edition of HP-75 KEY NOTES. There are four extra pages just for this latest star in the HP sky.

Book Reviews

Books are reviewed or announced in KEY NOTES only as a service to our readers. A review here does **not** represent an endorsement by Hewlett-Packard. If you are unsure about the contents or usefulness of a book, we suggest you first check with a local bookstore; if that fails, write to the publisher, not to KEY NOTES. Availability problems also should be addressed to the publisher, not to KEY NOTES.

CALCULATE BASIC STATISTICS, by **Mark Finkelstein** and **George McCarty,** is a new 352-page softbound book published in October. ISBN number is 0-936356-01-04 and the size is 5.4 by 8.5 inches (13.7 by 21.6 cm).

This book represents a friendly new calculator method for learning statistics, even if you are a novice with the subject or studying the subject in school. It features interesting, realistic examples that you follow step-by-step — instead of a lot of algebra. There are straight-to-the-point explanations for such topics as "confidence intervals" and "contingency tables" that are so confusing in ordinary textbooks. Also, any calculator or "home" or "personal" computer will do fine to make the calculations.

It's all here, including Mean and Standard Deviation, Sample Means, Inference and Hypothesis Testing, Chi-Square Distribution — the whole gamut of statistics. Plus, there are two bonuses at the end: a new type of table simplifies frequently used tests, and a BASIC program is given for multiple linear regression.

The authors are both faculty members at the University of California, Irvine. And you will remember that another of George McCarty's books, *Calculator Calculus,* appeared before in KEY NOTES.

This new book should soon be in book stores and is available now from the publisher, below. You also should check with your authorized HP Dealer; many of them are now stocking books reviewed in KEY NOTES — or they will order them for you. The price is $14.95 plus any state or local taxes. Add $2.50 per book for air mail in the U.S.; $3.50 for air mail to Canada and Mexico; and $6.00 for air mail to all other countries. The publisher is:

EduCALC Mail Store
27953 Cabot Road
Laguna Niguel, CA 92677 U.S.A.

HP-41 SYNTHETIC PROGRAMMING MADE EASY, by **Keith Jarett,** is another new book just off the press in October. It is spiral-bound, 190 pages, and 6.75 by 8.5 inches (17.2 by 21.6 cm) — roughly the same size as the HP-41 owner's handbook.

This is the *second* "synthetic programming" book to appear in KEY NOTES, the first being *Synthetic Programming on the HP-41,* by **Dr. William C. Wickes** (see V4N3p8 and V6N1p7c).*

For the uninitiated, here is the definition of synthetic programming — or SP — from Jarett's new book: "Synthetic programming encompasses the creation and use of synthetic instructions. Synthetic instructions cannot be keyed into program memory by normal means, but must be created by splicing normal instructions. Synthetic programming will work on any HP-41, does not require any modification of the calculator, and will not harm it. Thousands of HP-41 owners have used synthetic functions to obtain extended key assignment capability, 21 additional display characters, over 100 additional TONEs, extra 'scratchpad' registers, improved control of lowercase printer characters, and much more."

Because a great many of you are interested in SP, and because the Corvallis Library now accepts *limited* SP in programs (see V6N1p7c), it is very likely that some of you will be interested in this new book. It provides an introduction to SP, using recently developed techniques that make SP even easier than before. It is full of examples and step-by-step instructions. Also included are the latest "byte-grabbing" and "byte-loading" techniques for simple, fast creation of synthetic codes. Plus, you will find applications and utility programs for the Extended Functions and Time Modules. Also included in the book is a separate, handy, three-color plastic Quick-Reference Card for Synthetic Programming, a $3 value. This card contains a byte equivalence table that is the "Rosetta Stone" of SP. It also shows all 56 flag functions and incorporates a quick-reference summary of how to construct each type of synthetic instruction.

Keith Jarett is an inveterate "key-puncher," a steadfast member of the PPC (see footnote on page 12), and is, in his words, "addicted to HP calculators since I bought an HP-45 in 1973 while an undergraduate in Electrical Engineering." He graduated from Culver Military Academy in 1972 and received his B.S. degree in Electrical Engineering from Cornell in 1975, his M.S. degree in Electrical Engineering from Stanford in 1975, and his Ph.D. in EE from Stanford in 1979. He's currently a Systems Engineer at Hughes Aircraft. This book was started in January of this year, and here is how you can obtain a copy; write to:

SYNTHETIX
1540 Matthews Ave.
Manhattan Beach
CA 90266 U.S.A.

or you can check at your local bookstore or authorized HP Dealer.

Prices, according to actual mailing costs for various places throughout the world, are:

$16.95 to U.S., shipped fourth class.
$18.45 to U.S., priority mail or UPS.
$18.95 to Canada, Mexico, or Central America, air mail.
$20.45 to Europe or South America, air mail.
$21.95 elsewhere, air mail.

Be sure to add state or local taxes (California residents add $1.10 tax), and checks or money orders must be payable through a U.S. bank.

**Hewlett-Packard does not "support" synthetic programming. The HP-41 series hand-held Computers are only guaranteed to operate in accordance with the instructions and information provided in the* HP-41C/CV Owner's Handbook and Programming Guide. *Do not refer questions about synthetic programming to the factory. References for SP can be found in V6N1p8b.*

Editorial

Although this column has been missing for a while, your "Ed." is still alive and kicking. Some recent letters have asked why the editorial staff is not listed on the back cover, so we changed that to satisfy those who would like to see our names.

Because we have to make a "trade name search" to make sure that the name we pick as the winner of the "Name That Language" contest is not an infringement on the registered name used by someone else, we will not announce the winner until the February issue. So, if you sent an entry from a faraway place and worried about it getting in the contest, stop worrying; I included all of them. But, please, NO MORE! The contest is officially closed.

If you use your calculator or computer or any software or peripherals to increase your productivity in your business or at your job, you really should look into the probability that you can deduct those expenses from your income tax. This can save you a lot of money ... and then you can go see your local authorized HP Dealer and buy yourself a nice Christmas present.

A lot of you faithful readers will soon be receiving KEY NOTES subscription renewal notices from the friendly computer, and I sincerely hope you decide to renew. We are going to get more information into KEY NOTES next year, and a lot of it will deal with all the peripherals and the Interface Loop. At only $5 a year for the U.S. and Canada, you have to admit that KEY NOTES is a real bargain. For example, with 64 pages a year, that works out to 7.8125 cents a page. And although I've never accurately counted all of them, there are about 1400 words per page, which works out to somewhere over 180 words for each U.S. penny you spend on KEY NOTES. So make your tired old editor happy by sending in your renewal on time; then he won't have to get the computer on your trail. By the way, your expiration date is at top right on your label.

Letters to the editor should be sent to:

Henry Horn, Editor
HP KEY NOTES
Hewlett-Packard Co.
1000 N.E. Circle Blvd.
Corvallis, Oregon 97330 U.S.A.

We cannot guarantee a reply to every letter, but we do guarantee that every letter will be read by the editor or technical editor, and that as many as possible will be answered in KEY NOTES or in a personal response. Be sure to put your address *on the letter*; sometimes the envelope gets lost.

This is the ninth year that it is my pleasure to thank all of you for your support of KEY NOTES, for your steadfast participation, and for all your nice comments about this newsletter. So, on behalf of my assistant, **Ted Wadman**, and all the staff in the Users' Library, plus all of Hewlett-Packard, I wish every one of you, all over the world, a happy, safe, and joyous holiday season; a prosperous New Year; and many happy hours of productive programming ... and that you get a subscription to KEY NOTES for Christmas.

HP-75C Computer KEY NOTES

You will notice that there are 20 pages in this issue of KEY NOTES. That's because there are 4 extra pages devoted to the new HP-75C Portable Computing System. We are NOT going to usurp pages from the original KEY NOTES. You will find 16 pages, as in the past, devoted to the HP-67/97, and to the HP-41 system.

Because there are not many HP-75C's in the hands of users as yet, we have not had much feedback from them — except to tell us that they love the HP-75C. So we have written a couple of articles about the HP-75C to give you a head-start on using it, and this will also show our other KEY NOTES readers something about the world of computers and programming in BASIC. If you have some comments or ideas, we would like to hear from you.

Testing the HP-75C

Testing is an important part in the development of a new product.

Furthermore, thorough testing benefits both the customer and Hewlett-Packard, because the customer can purchase a new machine knowing that it was built to last, and HP saves the time and money that would have been spent repairing returned units.

The HP-75C had to go through a rigorous series of tests before it was released to production.These tests included the standard tests for HP calculators: line voltage variations, storage temperature range, operating temperature range, magnetic susceptibility, humidity, ESD (electrostatic discharge), vibration and shock, conducted and radiated RFI (radio frequency interference), 1-meter drop, condensation, supersoak (storage for 24 hours in a high-temperature, high-humidity environment), altitude, and a drop test of the final packaged product.

In addition, the HP-75C also was put through initial performance checks and 48 hours of preconditioning (temperature cycling) before the standard tests, and it also endured 6 weeks of temperature and power cycling with card reader checks after the tests. HP-75C's also were tested with X-ray levels hundreds of times more powerful than those in typical airport security systems to verify that users could take their portable computing machines with them without causing harm to the system.

Along the assembly line, each HP-75C receives a diagnostic test, a benchdrop test, and a card reader test. Following final assembly, 20% of the units go through temperature cycling, and a different 20% go through a shake test.

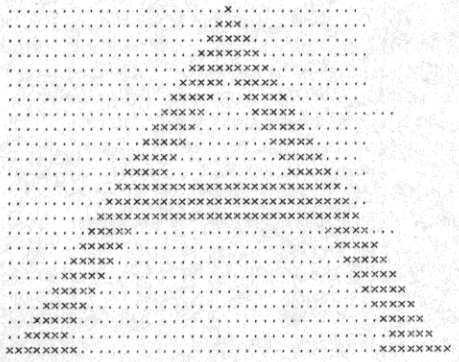
So, as you can see, the HP-75C has already proven itself before it even gets out of the factory, resulting in a better machine for you, and fewer repair bills for Hewlett-Packard.

Graphics with the HP-75C and HP 82905B Printer

The HP-75C can be used to do dot graphics on the HP 82905B (option 248) HP-IL Printer using simple PRINT statements. The PRINT statements control the dot patterns printed, and they also can be used to prevent the vertical spacing between lines of characters.

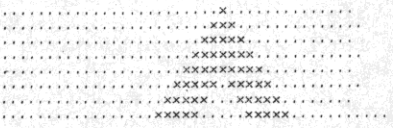
For example, let's look at creating an alphabet of letters whose size is three times larger than the standard printed characters. First, draw on graph paper what you will want the characters to look like. One pass of the printhead will print 8 dots, so the height in dots for each character will be 24 dots.

Here's an example for the letter 'A'.



Now, let's look at just the first row of 8 dots. Wherever a dot is presented, we want a dot to be printed. Starting at the left, we will look at each column of dots as if it were a binary representation for a number, with the least significant bit on the bottom, and the most significant bit on the top.A dot means that that bit is set, and a space means that bit is zero.

Starting at the left in our example, for the first column we read (from the top down) 00000000 binary, or 0 decimal. The next column is also equal to 0, and so on until we get to column 18. In column 18, we read 00000001B, or 1 decimal. Column 19 equals 00000011B, or 3 decimal, and so on.



After we have calculated the decimal values for all the columns in the first row, we will calculate the values for the columns in the second and third rows.

We are now ready to write a program to print our letter 'A' on the printer. For our example, we will assume that the printer is the only HP-IL device connected to the HP-75. The first thing to do is to assign the printer as the PRINTER IS device, and set the print width to infinite:

```
10 Assignio ':PR'
20 Printer is ':PR'
30 Pwidth inf
```

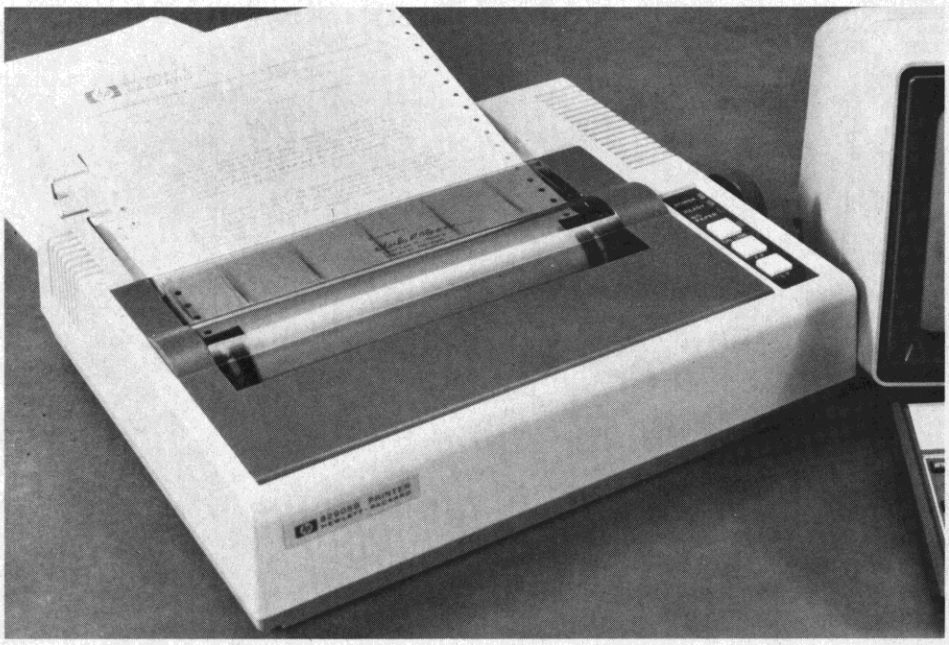
We now change the vertical spacing on the printer from 6 to 9 lines per inch, so we won't get a white space between lines of dots:

```
40 Print Chr$(27)&'&l9D';
```

(Notice that it is an 'l' and NOT A '1' in front of the 'D'.)

Since our letter has 49 dots across, we now tell the printer to expect 49 dots of dot graphics information:

```
50 Print CHR$(27)&'*b49G';
```



Now we send the decimal values of the dot columns for the first row:

```
 60 For I=1 to 49
 70 Read A @ Print Chr$(A);
 80 Next I
 90 Data 0,0,0,0,0,0,0,0,0,0
100 Data 0,0,0,0,0,0,0,1,3,7
110 Data 15,31,62,124,248,124,62,31,15,7
120 Data 3,1,0,0,0,0,0,0,0,0
130 Data 0,0,0,0,0,0,0,0,0,0
```

We now send a carriage return, and tell the printer to expect 49 more bytes of dot graphics information:

```
140 Print @ Print Chr$(27)&'*b49G';
```

The byte values for the second row are now sent, and the process is repeated for the last row of dots. We now finish our printing by printing a carriage return, setting the vertical spacing back to 6 lines per inch, and the width to 80 characters per line.

```
•
•
•
500 Print @ Print Chr$(27)&'&16D'
510 Pwidth 80
```

Some other uses for dot graphics include logos, signatures, personalization of memos and letters, and illustrating with graphs, maps, or charts.

The HP-75C Card Reader

The "hand-pulled" card reader is one of the most innovative features of the HP-75C. It is a small, light-weight, low-power design with no moving parts. Both the analog and digital circuits are contained on the same CMOS chip, including over 20 operational amplifiers. The card density is double that of the HP-67/97 cards.

All HP-75C cards have some prerecorded information on them, even the blank cards. If this information is erased, the card will be useless, so users must be careful about getting the cards near strong magnetic fields.

The information on the cards consists of a data track and timing track for each direction of recording. The data track contains four records:

(1) A card header block that holds information about the origin of the card (HP-Corvallis), and the maximum possible size of the card. This allows for possible future extensions in the card-handling capabilities of the HP-75C.

(2) A write-protect field that will prevent protected card tracks from being overwritten.

(3) A data header that identifies which file is stored on the card.

(4) A data field that holds up to 650 bytes of data from that file.

The timing track tells the hardware how fast the card is moving, enabling the hardware and software to accurately read the data from the card. The hardware can compensate for some variations in speed this way, allowing different users to pull cards at different rates, or even for a given pull to vary in speed.

Redefining Keys on the HP-75C

Another powerful feature of the HP-75C is the ability to redefine almost all the keys on the keyboard, and even some that cannot be accessed from the keyboard. The key redefinitions are kept in a text file, and can be listed or edited at any time. Key redefinitions can be added to the file by EDITing the file KEYS (notice that there are no quotes around the name for the keys file), or by executing a DEF KEY statement.

Editing keys, such as the left and right arrows or the insert/replace key, can be included in key redefinitions. For example, if you wanted to define [CTL][C] to print out: Copy card to '' and then backspace the cursor to the last quote and turn on the insert cursor, you could look up the decimal codes for [CTL][C], [←], and [I/R] in the reference tables in the HP-75C manual, and write the DEF KEY like this:

```
Def Key Chr$(3), "Copy card to'' "'"&Chr$
    (134)&Chr$(136);
```

Alternately, you could type in those keys directly by pressing [SHIFT] [I/R] (the 'literalize' sequence) before pressing the [I/R] and [←] keys. The literalized [I/R] key will show up in the display as a ≃, and the literalized [←] key will show up as a Γ.

```
Def key '←', "Copy card to 'Γ≃'";
```

The semicolon on the end of the DEF KEY means that the key will simply print the characters assigned to it and wait for the user to type. If there is no semicolon, the key would print out the string and do an immediate carriage return.

If you want to disable a particular key on the keyboard, you can assign a null string to it, followed by a semicolon.

The key definitions will stay in effect until a new definition is assigned or the KEYS file is purged. To get back the original definition for the key, press [SHIFT] [I/R] before pressing [CTL][C]. To permanently restore the original definition to the key, do a DEF KEY and assign the key back to itself. Remember to press [CTL][C] to literalize the key if you want the original definition.

Here's an idea for you to try. When the HP-75C times out to sleep after 5 minutes of inactivity, it does it by pretending key #160 [SHIFT][ATTN] was pressed. Try redefining that key to clear the TV screen before going to sleep, to run a program, or to display the time and stay awake.

Notes on HP-75C User Functions

User functions are normally used to compute a numeric or string value, based on parameters passed to the function. However, the multiple local parameters of the HP-75C's user functions allow another use that has some interesting possibilities. Functions may be used as a parameterized subroutine that does not destroy the parameters passed to it, and that returns a value simply because it has to.

For example, suppose we want a subroutine that prints on the display device all the integers between two numbers. For a normal GOSUB subroutine, this would require that the starting and ending values be assigned to two global variables, and that a GOSUB be done (i.e., X=5 @ Y=8 @ GOSUB 100). This requires that two variables must be tied-up by this call. Also, if the variables are changed by the subroutine, this change is reflected throughout the program. Using a user function for this subroutine (i.e. X=FNC(5,8)), means the values can be passed as parameters, and any changes in the subroutine are local and do not affect the rest of the program. However, the function must return a value, so in this case, we could return the count of numbers printed.

This technique can lead to more readable code, since functions can be written as self-contained modules with better defined inputs and outputs.

Multi-line string functions are limited to a returned string length of 32 characters. This can be gotten around in two ways: (1) single line definitions are not subject to this restriction, the value they return can be any length; (2) if you

(Continued)

cannot express your function as a single line, then the value can be returned in a global variable.

Global variables are normal BASIC variables, they can be accessed from anywhere in the program, and any change is reflected throughout the program.

Local variables are the variables declared in the parameter list of the function definition. They are only accessible to the statements located between the DEF FN and the END DEF, and any changes to them are only effective within this range; the value or variable passed-in is not affected. It is important to note that the parameters are only accessible to statements PHYSICALLY located in the function definition, NOT logically. This means that, if a program uses variable X and a function with parameter X, the program will access the variable outside the function, and the parameter inside the function. For example:

```
 10 X=5
 20 DEF FNA(X)
 30 GOSUB 100
 40 FNA=X
 50 X=10
 60 END DEF
 70 DISP FNA(6)
 80 STOP
100 X=X+2
110 RETURN
```

This program will display '6', and the final value of X will be 7. This is because the value of the parameter X was not incremented in the subroutine (the variable X was, so it now equals 7); parameter X still equals 6 (passed in) when it is assigned to the function value, the parameter X is then set to 10, but this has no effect on the variable X (which still equals 7).

How EXACT Time Is Calculated

When the first EXACT is done on the HP-75C after a system reset, the time of that EXACT is noted, the error and adjust accumulators are cleared, and the EXACT flag is set. All changes to the clock (by SET or ADJST) after this are broken into two parts: the time zone adjustments, and the errors. Time zones are to the nearest half hour, the error is what is left after the time zones are removed.

For example: +25 minutes is +1 time zone and −5 minutes error (5 minutes too fast); −25 minutes would be −1 time zone, and +5 minutes error (5 minutes slow). These adjustments and errors are accumulated until the next EXACT. The time of the new EXACT is adjusted to remove all the accumulated time-zone adjustments (this time is saved for the next EXACT sequence). The previous EXACT time is subtracted from this time to obtain the accurate elapsed time between the EXACTs. This timebase is used with the accumulated errors to determine the rate of clock adjustment needed to cancel the error. The sign of the accumulated errors determines whether the clock is slow or fast (+ means slow, − means fast). Clock adjustments are done in 0.25-second jumps, so an error of 1 second slow per day will result in four 0.25-second increments of the clock per day, or 1 every 8 hours. This is done automatically, regardless of what else the HP-75C is doing at the time.

Accessing Text Files From Programs

Text files can be accessed both from the keyboard or from a program by the use of the ASSIGN # command. This is handy for using or updating distribution lists, form letters, or address lists.

Under user or program control, you can create new text files, write lines of text to a text file, sort or rearrange lines in a text file, delete lines in a text file, update lines in a text file, and even search for a specific word in the text file. Text files created with an ASSIGN # command are treated the same as any other text file in the HP-75C. You can list them, edit them, rename them, merge them, and even transform them into BASIC. Of course, the lines of text would need to look like BASIC statements for the file to transform without errors.

The text file may be set up as a simple collection of lines of text, or it may be set up as a series of text records with a different item on each line. The first way is the easiest way to store letters, memos, lists, and most things that involve sentences or phrases. These text files are usually accessed serially, meaning that you access the first line first, then move to the next line, and continue until you reach the end of the file.

Setting up a text file as a series of records is useful if you want to store the same several pieces of information for different items. Using an address list as an example, you may want to store last names, first names, titles, employers, addresses, and telephone numbers for several people. This file of address records might be set up as follows:

Line Number	Information
1	Last name
2	First name
3	Title
4	Employer
5	Address line 1 - information
6	Address line 2 - for person
7	Address line 3 - #1
8	Address line 4
9	Work phone
10	Home phone
11	Last name
12	First name
13	Title
14	Employer
15	Address line 1 - information
16	Address line 2 - for person
17	Address line 3 - #2
18	Address line 4
19	Work phone
20	Home phone
21	Last name
•	
•	
•	

The information may be put in the file originally either by doing an ASSIGN # TO 'FILE', TEXT and then PRINT#ing the lines of text to the file, or by simply EDITting the text file and inserting the lines that way. It is not necessary to create the file ahead of time if you use ASSIGN#, since the ASSIGN# will create the file for you if it doesn't already exist. One word of caution: only strings may be PRINT#ed to text files, and only one string is allowed per line.

The information may now be accessed by using PRINT#s, READ#s, and RESTORE#s. For example, let's say you want to see if you have an entry for John Jones in your address file. Assuming that the file 'ADDRFILE' is set up in records

as shown earlier, you could search for John Jones like this:

```
10 Assign #1 to 'Addrfile'
20 On error goto 120
30 Input 'Last name?';a$
40 For I=1 to 9999 step 10
50 Read #1, I;b$
60 If uprc$(a$)=uprc$(b$) then goto 80
70 Next I
80 For J=1 to 10
90 Read #1,j;c$ @ Disp c$
100 Next J
110 Wait 1 @ goto 30
120 Disp 'Sorry, '&a$&' not found.'
130 Goto 30
```

To print out the address file, you could either write a program to print out the names and addresses in whatever format you wanted, or you could simply PLIST the address file. All of the line numbers in a text file are stripped off when the file is PLISTed.

HP-75C I/O (Input/Output)

The HP-75C is the first HP computer that has, built into it, the connectors and accompanying hardware for the Hewlett-Packard Interface Loop (HP-IL). So, the HP-75C is capable of input and output communication (I/O) with HP-IL devices without the purchase of an additional module or interface card. The implications of this step are many. The major implication is that understanding HP-75C I/O is a requirement for making full use of the machine. It is on this premise that this article is based.

After realizing that HP-IL I/O capabilities are built into the HP-75C, there are two more discoveries to be made. First, I/O may be turned on and off, and second, devices on the HP-IL loop must be assigned device codes with which they can be addressed by the HP-75C. These two features do allow more flexibility in directing information flow.

The standard HP-75C contains 13 functions for controlling HP-IL. These functions are listed below where we have adopted this notation: dc = device-code (any two-letter code that you choose); fn = file name; fs = file specifier, in the form "fn:dc" where fn is the name of the file to be found on the device specified by dc.

ASSIGN IO: allows you to assign names (two letter device-codes) to devices on the HP-IL loop.

DISPLAY IS ":dc": declares a display device by its previously assigned device-code.

PRINTER IS ":dc": declares the printer device by its device-code.

CLEAR LOOP: resets devices on the loop.

OFF IO: turns off loop communications.

RESTORE IO: restores loop communications.

INITIALIZE ":dc": initializes the mass storage medium.

CAT ":dc": displays the directory of files on the mass storage medium.

COPY "fn" TO "fs": copies a file from memory to mass storage.

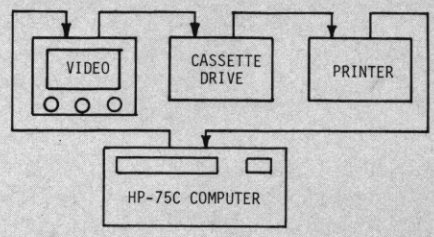
COPY "fs" TO "fn": copies a file from mass storage to memory.

RENAME "fs" TO "fs": renames a mass storage file.

PURGE "fs": purges the specified mass storage file.

PACK ":dc": packs a mass storage media.

Before we can execute any of these commands, it is necessary to ASSIGN IO. An example loop is diagrammed below, and the following command sequence would be used to assign "pr" as the device-code for the printer, "cd" as the device-code for the cassette drive, and "tv" as the device code for the video.



After plugging both cables into your HP-75C, the smaller of the two free ends indicates the direction of data flow and device location #1. After assigning I/O, it is necessary to tell the HP-75C which device on the loop is the display and which device is the printer. In the example, we would type-in: PRINTER IS ':PR' [RTN] DISPLAY IS ':TV' [RTN].

The terms 'tv' and 'cd' are two-letter codes that you make-up. Once a device in the loop is assigned a code, then all references to that device are made by using a colon, followed by the code within either single or double quotation marks.

Once a device is specified as a display, it will receive all information that appears in the HP-75C display. Similarly, a device specified as a printer will receive all information bound for a printer. An example would be to specify a video display device, such as the HP 82163A Video Interface, as the DISPLAY IS device. Now, the display will be duplicated on the screen. The edit keys will even move the cursor on the screen. Ad-

KEY IN	SEE DISPLAYED	ACTION TAKEN
assignio	3 DEVICE(s) ON LOOP DEVICE #1 = ' '	
tv	DEVICE #1 = 'tv'	Assigns device 1 as the video.
	DEVICE #2 = ' '	
cd	DEVICE #2 = 'cd'	Assigns device 2 as the mass storage.
	DEVICE #3 = ' '	
pr	DEVICE #3 = 'pr'	Assigns device 3 as the printer.
printer is':pr'	printer is':pr'	defines the selected printer.
display is':tv'	display is':tv'	defines the selected display.

Back Issue and Subscription Information

BACK ISSUES

Back issues of KEY NOTES are available back to V3N3, which introduced the HP-41. **All back issues are available ONLY from Corvallis, Oregon.** An index of these will be furnished on request. Available issues are:

V3N3 August 1979 (12 pages)
V3N4 November 1979 (12 pages)
V4N1 March 1980 (12 pages)
V4N2 Jun-Aug 1980 (12 pages)
V4N3 Sep-Dec 1980 (12 pages)
V5N1 Jan-Apr 1981 (16 pages)
V5N2 May-Aug 1981 (16 pages)
V5N3 Sep-Dec 1981 (16 pages)
V6N1 Jan-Feb 1982 (16 pages)
V6N2 Mar-May 1982 (16 pages)
V6N3 Jun-Aug 1982 (16 pages)

Prices for KEY NOTES back issues are as follows. All prices include first-class or air mail. Payment must accompany your order and must be a check or money order in U.S. dollars drawn on a U.S. bank. Or you may use your American Express, VISA, or MasterCard account; be sure to include your account number and card expiration data. Your order will be promptly mailed in an envelope.

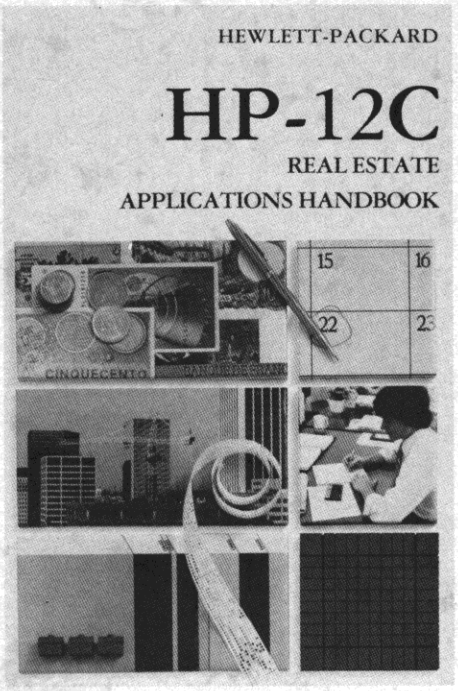
NO. OF ISSUES	U.S., MEXICO CANADA	ALL OTHER COUNTRIES
1	$ 2.00	$ 3.50
2	$ 3.50	$ 5.00
3	$ 5.00	$ 6.50
4	$ 6.00	$ 8.00
5	$ 7.00	$ 9.00
6	$ 8.00	$10.00
7	$ 9.00	$11.00
8	$10.00	$12.00
9	$11.00	$13.00
10	$12.00	$14.00
11	$13.00	$15.00

SUBSCRIPTIONS

KEY NOTES is published quarterly in February, May, August, and November. A one-year subscription in the U.S. and Canada is $5* a year. It is free (worldwide) if you are a member of the Corvallis Users' Library ($20* U.S. and Canada; $35* elsewhere). Send your payment and complete name and address to the Corvallis address on back cover.

To get a subscription to KEY NOTES in Europe, contact the UPLE (Geneva address on back cover). To get KEY NOTES elsewhere, contact your nearest HP Sales Office or send your name, address, and calculator serial number to the Corvallis Users' Library, and we will forward your request to your nearest HP Sales Office.

**U.S. dollars. Payment must accompany order.*



HP Targets the Real Estate Industry

Continuing its commitment to provide portable solutions for the real estate industry, Hewlett-Packard recently announced the new *HP-12C Real Estate Applications Handbook*. This handbook is the first industry-specific software written for the HP-12C Advanced Financial Programmable Calculator.

Written by two industry practitioners to exacting HP specifications, the *HP-12C Real Estate Applications Handbook* provides the solutions needed by not only real estate professionals but also home buyers. Brokers, appraisers, investors, analysts, and others who daily have to make wise real estate decisions will benefit from the wide range of real estate solutions.

Over 100 pages of easy-to-read and easy-to-use pre-written programs, keystrokes procedures, and detailed explanations provide quick and accurate solutions for real estate decision-making. The range of solutions available in this handbook is nothing short of amazing. From alternative financing to appraising, and from complex investing to cash-flow analyzing, the HP-12C and this handbook make it easy for the adept professional or the first-time calculator user to become an expert at solving complex real estate problems, especially those that are prevalent in today's economic situation.

But this new handbook only *complements* the powerful built-in features of the remarkable HP-12C. However, with *both* at your command, you will have a truly awesome set of tools to conquer any and all real estate industry challenges.

The handbook number is 00012-90015, and it is available now at your local authorized HP Dealer. It is not only a very useful and special Christmas present for an HP-12C owner, it is undoubtedly a tax-deductible asset for everyone in the real estate industry . . . maybe even for prospective home buyers.

ditionally, if we were to specify a printer as the DISPLAY IS device, it will mimic the display.

There are two methods you can use to avoid the manual assignment of devices on the loop. The first method assumes that the loop configuration is known. The second method can be used with various loop configurations.

The first method is to take advantage of the HP-75C key-definition feature. Simply assign, to a key, the command sequence required to ASSIGN IO. If we were to define the [*] key to accomplish the above task, we would use the command: def key "*", "assignio':tv,:cd,:pr'@ printeris':pr'@ displayis':tv'" [RTN]. Then, we could make the loop assignments at the touch of the [*] key.

The second method requires the User's Library *I/O Utilities Solutions Book* and the magnetic cards that go along with it. Plus, it requires a short BASIC routine. The following routine is an example of how a simple loop could be automatically assigned without regard to the order or number of devices used. It handles the HP 82161A Digital Cassette Drive, the HP 82162A Thermal Printer, the HP 82163A/B Video Interface, and the HP 82905B Option 248 Printer.

We gave the name "AUTOIO" to the file containing this program and assigned it to our [CTL]/[ATTN] key. We used the command: defkey chr$(192), "run 'autoio'" [RTN] to define this key. Now, whenever we set up an HP/IL loop consisting of any of the above peripherals (one of each), in any order, it's a simple press of the [CTL]/[ATTN] key and we're in business.

```
10 ASSIGN IO ':zz'              !wake up the loop
20 SENDIO '','aau,aad1',''      !auto address the loop
30 Y$=''                        !null the Y$ string
40 FOR I=1 TO 30                !set up to address each device
50 IF I=1 THEN Z$='' ELSE Z$=','    !place a comma between each device code
60 B$=ENTIO$('','tad'&STR$(I)&',sai)!ask for accessory I.D.
70 IF B$='' THEN GOTO 140           !is there a device here
80 IF NUM(B$)=48 THEN X$=':tv'   !is this a display device?
90 IF NUM(B$)=16 THEN X$=':cd'   !is this a mass memory?
100 IF NUM(B$)=32 THEN X$=':p1'  !is this a strip printer?
110 IF NUM(B$)=33 THEN X$=':p2'  !is this an 80 column printer?
120 Y$=Y$&Z$&X$                  !build the device code string in order
130 NEXT I                       !get the next device
140 ASSIGN IO Y$                 !assign the loop according to Y$
150 A=POS(Y$,':p1') @ B=POS(Y$,'p2') @ C=POS(Y$,':tv') !find the printers
                                 and video positions in the loop
160 IF A>0 THEN PRINTER IS ':p1' @ PWIDTH 24 !make stip pr system printer
170 IF B>0 THEN PRINTER IS ':p2' @ PWIDTH 80 !make 80 col pr system pr
180 IF C>0 THEN DISPLAY IS ':tv'             !make video system display
190 !if both printers are present then the 80 col printer is system printr
```

HP KEY NOTES
Editor
Henry C. Horn
Technical Editor
Ted Wadman
Published quarterly (see page 19) for owners of Hewlett-Packard handheld computers. Reader comments or contributions are welcomed. Please send them to one of the following addresses.

Hewlett-Packard Company
Users' Library
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 U.S.A.

Hewlett-Packard SA
Users' Program Library Europe
7, Rue du Bois-du-Lan
P.O. Box, CH-1217 Meyrin 2
Geneva-Switzerland

Hewlett-Packard Company
Corvallis Division
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 U.S.A.

Address Correction Requested

FIRST-CLASS MAIL
U.S. POSTAGE
PAID
PERMIT NO. 814
PORTLAND, OR

FIRST CLASS

Printed in U.S.A.