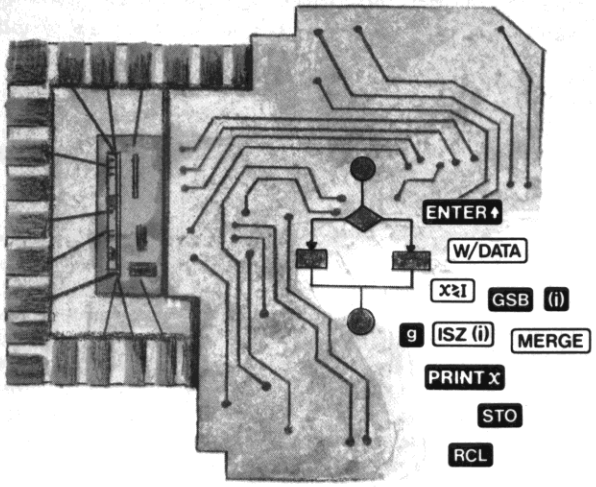


Featuring, this issue:

January-April 1981 Vol. 5 No. 1

HEWLETT PACKARD

HP Key Notes

Two HP-41's Shuttled Into Space

While the entire world's attention was riveted to tracking the progress of *Columbia* on its initial space shuttle flight and its spectacular, letter-perfect return to earth, did you know that two HP-41C Calculators were being used onboard *Columbia*? Well, we are proud to say they *were* used, and we thought that KEY NOTES readers would enjoy reading about this historic event.

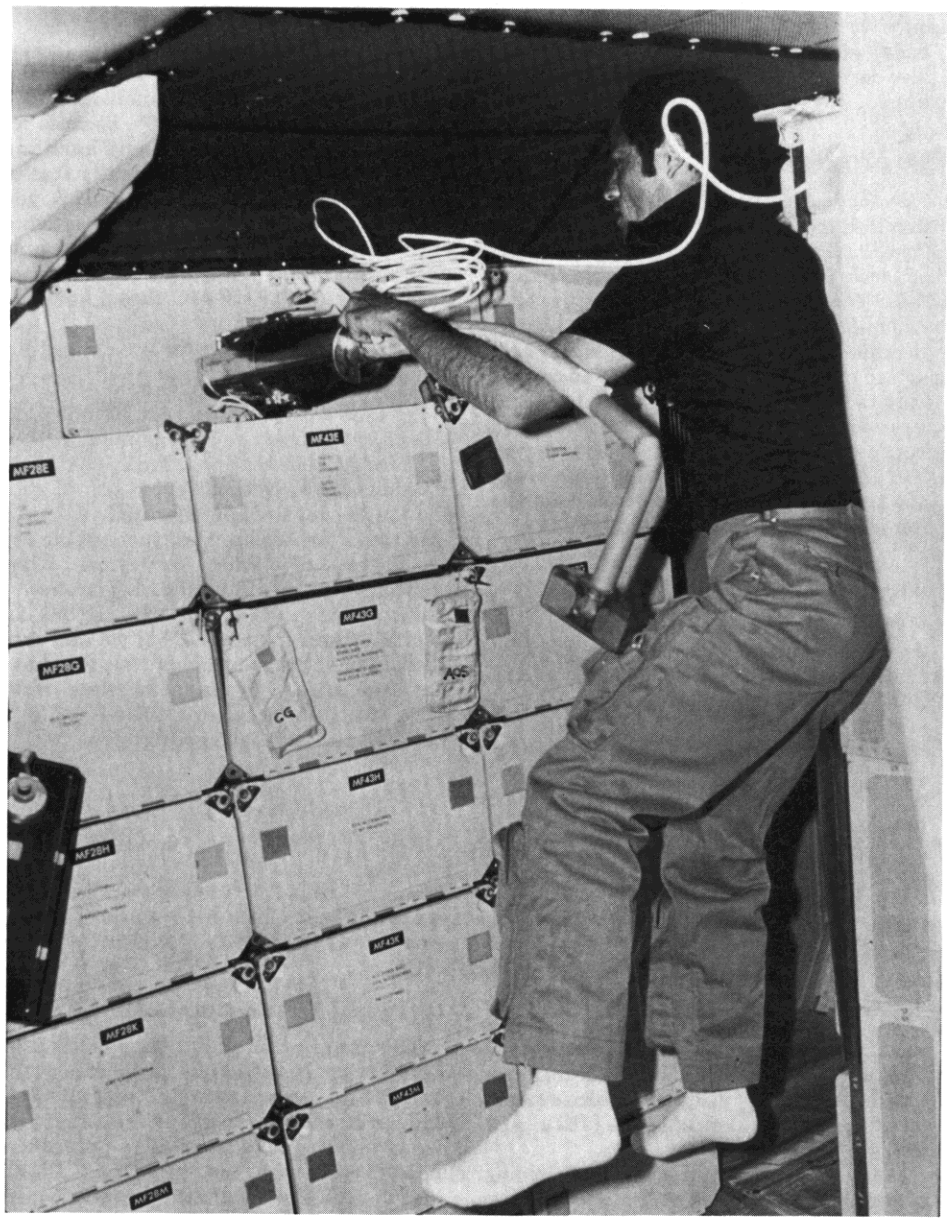
In 1980, when the National Aeronautics and Space Administration (NASA) of the U.S. began studying available calculators for use on space shuttle flights, they soon realized that the most important initial factor was large memory, an absolute necessity in order to accommodate the lengthy programs that had been proposed. The search for a calculator soon narrowed down to two machines, and a "fly-off" was held between the two. The HP-41C was chosen, NASA said, for a variety of reasons, chief of which was the HP-41C's alphanumeric LCD display.

Then the HP-41C was subjected to rigorous tests, as was all shuttle hardware, before being judged flightworthy. Some of those tests were conducted at NASA's station at the White Sands Missile Range in New Mexico, and included shock and vibration tests, and tests for outgassing. As a result of the tests, some minor changes in the HP-41C were made, and it was certified for flights in the space shuttles.

Two calculators were set up for the flight. Each HP-41C was outfitted with four Memory Modules, giving each the memory to handle more than 2000 program lines. The flight suit pouches for the calculators also held extra Memory Modules, extra batteries, and a card reader and magnetic cards containing the programs, just in case they had to be reloaded in flight.

For the first shuttle flight, one HP-41C was dedicated to the Center of Gravity program, and one to the Acquisition of Signal program. These programs were loaded into the calculators shortly before launch. The Center of Gravity program was

(Continued on page 16)



Astronaut Robert Crippen in "weightless suspension" in mid-deck area of *Columbia*, somewhere in orbit. The two HP-41C's, in protective cases marked "CG" and "AOS" (for the Center of Gravity and Aquisition of Signal programs) appear in the center of the photograph. (Photo by astronaut John Young, courtesy of NASA.)

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

Library Corner

All of the programs highlighted in KEY NOTES are available worldwide. However, *before you order any, be sure to read the paragraph below:* "**Ordering Programs.**"

GROWTH OF USERS' LIBRARY

In terms of raw numbers, the Corvallis Users' Library is rapidly expanding not only in programs but also in staff members. There are now over 4,500 HP-67/97 programs and approximately 800 HP-41 programs. Also, the Library has grown from three clerks one year ago to a current staff of ten, which includes three technical advisors. Therefore, submitted programs are now reviewed by the Library staff. Some new guidelines for submitting programs are now being formulated and will be published in the next KEY NOTES. The additional staff also will enable us to establish a 48-hour turnaround on orders, once we manage to reduce the present large backlog. The subscription coupon backlog also is fast approaching that turnaround time.

NEW CATALOG

By now, all Users' Library subscribers should have received the latest *Catalog of Contributed Programs.* This new *Catalog* replaces and supplements the November 1979 *Catalog* and the August 1980 *Addendum.* Any current Users' Library member who has not received his/her copy should notify the Library by phone or mail. Also, some catalogs with pages missing were accidentally sent to our subscribers from our mailing house. Pages 4-17 through 4-48 are missing in a few copies. So please contact us if your *Catalog* is not complete, and we will send a new one immediately.

ORDERING PROGRAMS

HP-67/97 and HP-41 programs mentioned 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 a prerecorded magnetic card (or cards). Also, for HP-41 programs, this 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 or if you are ordering from Europe, South America, or Asia, 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 Alaska and Hawaii (in Oregon call 758-1010).

Here's a helpful hint for customers outside the U.S.: We have found that your orders are handled in a more efficient and timely manner if you will send, **attached to your order**, an International Money Order, a Foreign Draft, or the equivalent. *Any of these must be in U.S. dollars, drawn on a U.S. bank,* otherwise they will be returned to you, which involves a long delay for you. Much time is wasted and orders are held up in trying to match orders and checks that are sent in separately, or written on checks for non-U.S. banks and in foreign currency. Another option for you is to use such major credit cards as American Express, VISA, or MasterCard.

Orders *not* delayed by the above problems can normally be shipped within 48 hours after they are received in Corvallis.

LIBRARY SUBSCRIPTIONS

In the United States and Canada, the fee for a one-year subscription to the Users' Library is $20.* If you live outside the U.S. or Canada, the fee is $30* because of considerably higher postage and handling charges. KEY NOTES is presently free in the U.S., but in areas outside the U.S. you must be a member of the Library in order to receive it. The only exception is the free one-year subscription presently offered to most purchasers of the HP-41C throughout the world.

SUBMITTING PROGRAMS

Up to now, every program submitted to the Library had to include a magnetic card (or cards). We have a good reason for this; without a card or cards, it would take far too long to review and check all the many program submittals. Also, there is always an increased chance for errors when someone keys in handwritten keystrokes.

Since the advent of the HP 82153A Digital Wand for the HP-41, we can now accept HP-41 program submittals that have bar code instead of magnetic cards. However, **the bar code you submit with a program must be reproducible.**

NEW PROGRAMS

Here are some recent submittals to the Corvallis Users' Library. All of the programs in this issue are available worldwide, *but before you order, be sure to read (above):* **"Ordering Programs."**

(41) Wand Scatter (#00734C)

This program is an adaption of #00219C Scatter (from the *Games* solutions book), in which the player has to find up to 9 atoms hidden in a box by probing with rays and watching the reflections. The program is now played with the HP 82153A Digital Wand and the bar code layout for the Search and Destroy game in the *Wand Owner's Manual,* making the game both simpler and more enjoyable. *Required accessories: One Memory Module, HP 82153A Digital Wand, and Wand Owner's Manual.* (259 lines, 8 pages)

Author: **Neil M. Hunter-Blair**
Thailand

(41) Model Airplane Design—Radio Control Competition Pattern (#00707C)

Thirty-five design parameters are calculated from empirically derived data coefficients and analytical relationships. The program is organized so that data coefficients for different types of model airplanes (for example, pylon racers, sport trainers, etc.) may be maintained on separate data cards. *Required accessories: Two Memory Modules, Card Reader, Printer.* (394 lines, 12 pages)

Author: **Karl L. Remmler**
Palmdale, California

(41) Triads, Chords of the Seventh, and Chords of the Ninth (#00708C)

Given the key (tonality), chord (in arabic numeral), and mode (major or minor), the program outputs the triad, chord of the seventh, or chord of the ninth. Given a triad, it provides a complete list of the appropriate keys, chords, and modes. *Required accessories: Two Memory Modules.* (419 lines, 11 pages)

Author: **Han Y. Rhyu**
Seal Beach, California

(41) Shovelton & Karup-King Interpolation Formulae (#00631C)

This program interpolates by either Shovelton's osculatory six-point formula or Karup-King's osculatory four-point formula. The values are printed and are stored for computation, printing, and review of the first through fourth differences. *Required accessories: Three Memory Modules and Printer.* (572 lines, 16 pages)

Author: **Walter W. Steffen**
Indianapolis, Indiana

(97) Polynomial Curve Fit Coefficients (Equally Spaced Points) (#04470D)

This program computes the coefficients of an Nth degree polynomial passing through N + 1 equally spaced points. Up to 21 points (N = 20) can be handled. Input, review and correction, and series and single point evaluation routines are included. Coefficients of a 9th degree polynomial are determined in 4.5 minutes. Direct evaluation (using coefficients) is much faster than indirect evaluation (Lagrange interpolation). Coefficients for unequally spaced data (up to 10 points) may be calculated without restoring data, after conversion to

**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.*

equally spaced data by program #04471D. (147 lines, 7 pages)
Author: **Christopher R. Stevens**
Phoenix, Arizona

(97) Lagrange Interpolation With Equal Spacing Conversion (#04471D)

This program performs Lagrange Interpolation between up to 10 arbitrarily spaced points (N = 9). Run time is proportional to N. Input, review and correction, series and single point interpolations, and equal spacing conversion routines are included. Direct evaluation (using coefficients) of the polynomial is much faster than Lagrange Interpolation (indirect evaluation). The equal spacing routine converts arbitrarily spaced data to data equally spaced and stored such that the coefficients of the polynomial can be calculated using Polynomial Curve Fit Coefficients (equally spaced points) program #04470D. (224 lines, 7 pages)
Author: **Christopher R. Stevens**
Phoenix, Arizona

(67/97) Electric Transmission Line (#04496D)

This three-card program calculates the complex line currents, bus voltages, and losses on either a radial or looped electric transmission line of up to nine line sections and loads. The source or end-of-line voltage, impedance of each line section, and the load taken off of the system at each bus are the required inputs. Intended for electric utility engineers, this is number 1 of a utility series. (530 lines, 21 pages)
Author: **Daniel H. Mulkey**
Salinas, California

(67/97) Conductor Sag and Tension (#04497D)

Designed for utility distribution cables involving both even and uneven terrain, this program includes: calculation of horizontal tension and sag at any point along the catenary curve; cable length; and resulting changes due to a change in temperature. Inputs are initial temperature, cable weight per unit length, difference in support elevation, span length, and either the cable sag or horizontal tension. Written for the HP-67/97 for maximum exposure, this is number 8 of a utility series. (196 lines, 18 pages)
Author: **G. Robert Harvey**
Weimar, California

(67/97) Coefficient of Evaporative Heat Exchange (#04498D)

In order to determine the quantity of water evaporated over a period of time, the surface heat exchange coefficient for evaporation must be known. This program calculates this parameter as a function of windspeed, air dry-bulb temperature, and air dew-point temperature. From these, the evaporation rate in gallons/day or inches/day are calculated. (224 lines, 8 pages)
Author: **Michael Krabach**
Framingham, Massachusetts

(67/97) Chess—The Eight Queens Problem (#04505D)

It is possible to place eight queens on an 8×8 chessboard in such a way that no queen attacks another. This program will find (and list on the HP-97) all twelve distinct solutions. The solutions are stored in the data registers, so that HP-67 owners also can use the program. Execution time is about eight hours. (224 lines, 6 pages)
Author: **Kiyoshi Akima**
Boulder, Colorado

(67/97) Countersink Design (#04518D)

Without special equipment it is difficult to make direct measurement of a countersink diameter. It is possible, however, to indirectly measure the countersink diameter by measuring the height of a precision ball placed in the countersink. This, also, indirectly measures and assures that the corresponding countersink angle is within the specified requirement. This program was developed to provide the engineer with a method to determine the correct (indirect) specification of a countersink for an engineering drawing. (190 lines, 13 pages)
Author: **Michael U. March**
Lomita, California

SOME SPECIAL PROGRAMS

Occasionally, programs submitted to the Library are put in a category of "Special Program," by virtue of length, value, etc. The programs that follow are such "special" programs. *These programs are available only from the Corvallis Library and carry the 10 percent postage and handling charge to overseas locations.* See "Ordering Programs" before you order.

(41) Coordinate Geometry System CGS1 #67000-99964 is a prodigious piece of work—practically a book! There are pages and pages of description, and it is replete with a logic diagram. For the price of $12* you get 68 pages, 25 magnetic cards, and 1445 lines of programming. It requires two Memory Modules (or HP-41CV), the card reader, and the printer. The author is **John T. Potts, Jr.,** a consulting engineer from Rifle, Colorado. Here's the abstract:

CGS1—a complete system for coordinate geometry calculations. The program stores unlimited-size files of N/E coordinates on cards. It also computes traverses, inverses, intersections, areas, curve data, and field angle traverse. Useful for subdivision design, highway alignment, or wherever lines and curves are used to obtain coordinates.

(41) Advanced Star Trek #67000-99962 just has to be, for all games fans, the bargain of the year. For the price of $12* you receive a total of 56 *typed* pages, 20 magnetic cards, and 1085 lines of programming. It requires three Memory Modules if you use a printer and four if you don't. Or you can use an HP-41CV. It is actually the HP-41 version of the HP-67/97 program #00369D. And to help load the lengthy programs, WALL cards are provided for those who have a card reader. Here's the abstract:

There are two formats: one with printer and one without printer. The non-print version has a practice firing range. The ALPHA displays for status are more extensive in the non-print version. But both versions are essentially identical. Fully automated functions include: course control; advanced sensor systems; adjustable shields; phasers; photon torpedoes; transporter/tractor beam (for Nubian freighter); three enemy ships: Klingon, Romulan (with cloak), and Vallician; and corbomite maneuver (with self-destruct). Also, the computer with the non-print version gives the course to middle of "mission sector," plots the course to any coordinates, and gives weapons firing angles. And with the printer version, target practice simulates firing on an enemy vessel. The game is played in a three-dimensional cube.

Last but not least, the author of this remarkable opus is **James A. Patterson,** who is presently somewhere up around the Arctic Circle in the Northwest Territories of Canada.

(By the way, don't forget that many programs you purchase may be deductible on your income tax. In fact, even your calculator and accessories might qualify for such deductions. It is worth your time to check on this!—Ed.)

KEY NOTES Going to Subscription

No matter where you live in the world, you know that skyrocketing inflation has caused severe increases in the costs of labor, materials, postage, and freight. And since KEY NOTES has more than doubled its mailing list in just the last year, we have to make a decision to either discontinue the newsletter or charge a small subscription price for it.

We are very much aware that you like and enjoy your KEY NOTES, and that it favorably carries the HP message far and wide. So we have no intention of discontinuing the newsletter. Economics, however, force us to start charging a subscription price starting in January 1982.

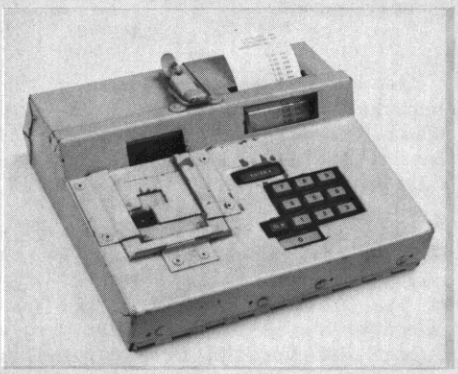
We realize this might evoke many questions, but we promise that you will be kept well-informed of our plans as they become final. In the next issue (Vol. 5 No. 2) we will give you more details, so please do not write for information before that time. Your KEY NOTES is now an enormous project, crossing the borders of nearly every country in the world, and with a readership somewhere over 250,000 people, it takes a bit of time to iron out all the wrinkles in any changes we make.

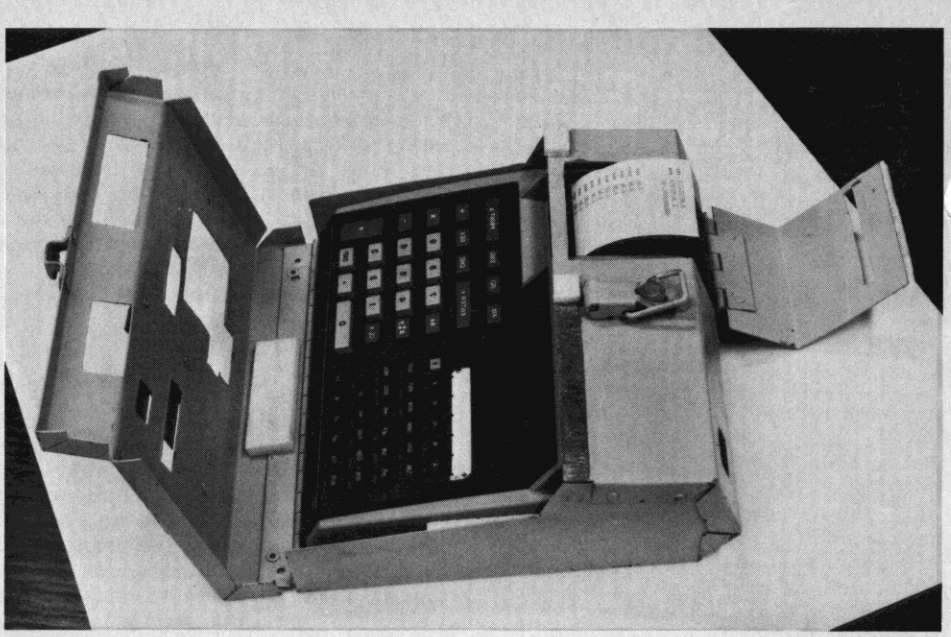
So please be patient for a while longer. We promise that you will not regret the wait.

An "Armored" HP-97!

When you have millions of customers buying your products, you occasionally get some of them back for repair. Generally, however, those products look just like they did when they left the factory. So imagine our surprise when we saw the "armored" HP-97 shown here. It had arrived in Service just the way you see it here, with a heavy steel box surrounding it and a hasp on top so it could be locked and chained in a particular location.

As you can see, it also has a built-in metal slide so that only the A and B user-definable keys can be pressed. And, although you cannot see it in this print, the armored box is painted a bright yellow.





Very clever, right? Well, *we* thought so. And being curious about its background, we called the owner, which happened to be Davidson Rubber Division (a subsidiary of Ex-Cell-O Corporation) in Farmington, New Hampshire, and talked to **Dick Leland**, who had sent in the HP-97 for repair.

The HP-97 was set up like this to make it tamper-proof and to protect it in its location on a production line where crashpads for automobile dashboards were being manufactured. The armored box prevented access to keys that weren't needed, and it also kept plastic material out of the calculator. It was programmed to accept such data as the mold number, part number, a record of the weight (in grams) of the part, and so on. The tapes could then be used to determine the productivity from the molds and to act as records for future budgeting and planning.

Very ingenious, wouldn't you say? Here's an application idea you might want to use in your business.

And, by the way, do any of you use our calculators in an underwater environment? We'd love to see *that*!

HP-97 Used in Steel Structure Design

If you are interested in steel structure design, you might want to investigate a two-volume booklet we recently saw. It is entitled: *HP-97 Programs for Design of Steel Structures*. The first volume contains ten programs and volume 2 contains three programs. And although they were written specifically for the HP-97, they can be used successfully in the HP-67 and the HP-41.

These programs were developed by the Canadian Institute of Steel Construction (CISC) in-association with the Canadian Steel Construction Council (CSCC). And since this subject is not everyone's cup of tea, we suggest you contact the source, at the address below, before you order the two-volume set (also available as separate volumes). Magnetic cards for the programs are available, and if you send sufficient blank cards, the service will cost you very little. However, there is a charge for the booklets and postage and handling. For more information, write to:

Mr. Michael I. Gilmore
Manager of Engineering
Canadian Institute of Steel Construction
201 Consumers Road, Suite 300
Willowdale, Ontario
Canada M2J 4G8

Quad RAM Questions

The HP 82170A Quad Memory Module (Quad RAM) for the HP-41C was introduced in January 1981 at the same time the HP-41CV was introduced, and there have been many questions written to "—Ed." ever since. So, rather than answer them separately, here's the entire rundown on these two new products.

HP-41CV: This calculator *looks* exactly the same as an HP-41C. The real difference is internal. The HP-41CV, as purchased, contains 319 *built-in* storage registers and *all* of the memory is *Continuous*. You cannot add either single-density, original, HP 82106A Memory Modules or the new Quad RAM, because the HP-41CV already has *maximum memory*. You *can* use the HP-41CV's ports for peripherals and modules (ROM's) from the Application Pacs.

Quad RAM: The HP 82170A Quad Memory Module *replaces* four standard HP 82106A Memory Modules. In other words, it *adds* 256 registers to the HP-41C, and these 256 additional registers are *Continuous Memory* as long as the module is plugged into a port. When you have the Quad RAM in a port, you *cannot add* more HP 82106A Memory Modules. You *can* add peripherals or modules from Application Pacs. And, yes, the Quad RAM will work in any port. And, no, the original Memory Modules are *not* obsolete, but you cannot use one in the HP-41CV or in an HP-41C that contains a Quad RAM.

Software Changes

After the introduction of the *HP-41 Real Estate Pac* (00041-15016), Hewlett-Packard discovered several software errors in some of the programs. Because of this, it is possible to obtain results that appear correct but which, in fact, may be incorrect. These types of results may occur under the following conditions.

1. Using "$" or "AMORT" in BEGIN mode.
2. Using the reassigned keys (n, i, PV, FV, ...) after using "IPA."
3. Using "IPA" when inputting a third mortgage.
4. Using the editing capabilities of the "MIRR" program.

While the Pac is still very usable, it does not meet Hewlett-Packard's high quality standards. Because of this, we have temporarily removed the Pac from distribution, and we are now correcting the Pac to make it consistent with HP quality. The revised Pac will be available in June of 1981.

Also in June, a corrected version of the module may be obtained by sending the old one to your nearest HP Service Center, along with a request for a ROM update. Hewlett-Packard will continue to provide replacement Real Estate modules for up to

one year from the original date of purchase. In the U.S., send your module to:

Hewlett-Packard Company
Service Department
1000 N.E. Circle Boulevard
P. O. Box 999
Corvallis, Oregon 97330 U.S.A.

If, by any chance, some of these Pacs have gotten to other countries, contact your nearest HP Service Center **after June** for further instructions.

A detailed description of all the errors, including the ones mentioned above, plus ways in which to "get around them," may be obtained by writing to:

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

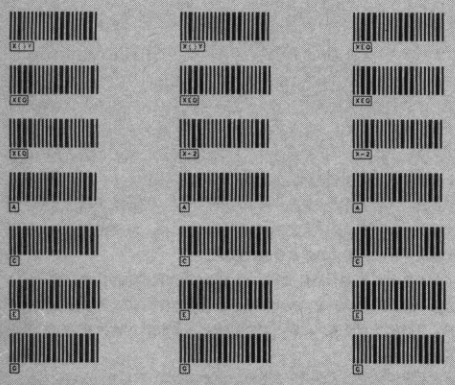
Generating Your Own Bar Code Programs

Since the introduction of the HP 82153A Digital Wand, many of our readers have written to ask us, "What is the best way to generate bar code?" or "How is bar code generated?" You can, of course, tediously make it by hand or use labels. However, the best thing is to have it made for you.

So we asked our bar code supplier to contribute an article for KEY NOTES. Besides doing a good job of generating bar code for us (and you), we think they also did a good job of presenting an article for our readers. Here it is.

For users of the HP-41 calculating system, there are several ways to generate custom programs in bar code. But in terms of time and expense needed to produce high-quality, reproducible bar code, one method is clearly the easiest to use.

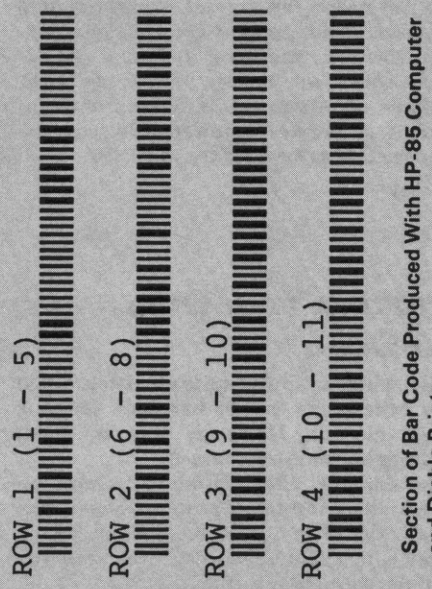
The simplest way to assemble a program in bar code is to use the pressure-sensitive bar code labels provided by Hewlett-Packard. The labels are available for every function built



into the HP-41 and its peripherals. For short programs, this method works very well. With a long program, however, the process can be a tedious one. Because each label is read individually by the HP 82153A Digital Wand, a series of pressure-sensitive bar code labels cannot be scanned as rapidly as a row of standard program bar code. And since duplicating bar code labels on an office copier can produce shadows along the edges of the raised labels, the Wand may detect these shadows, and produce an error.

The HP-41 printer can be used to generate patterns that resemble bar codes; but the patterns are not recognized by the Wand.

HP also supplies programs in BASIC that will generate bar code in the HP-41 format. These programs are available for HP 9845A and HP-85 Computers, and require access to a plotter, daisy-wheel printer, or dot matrix printer. (The plotter must be capable of producing solid, dense lines at least 0.015 inch (0.381 mm) in width for a narrow bar and at least 0.030 inch (0.762 mm) in width for a wide bar.) Anyone with a different hardware configuration must adapt the language used, as well as rewrite the sections that print the bar code. Also, the bar code program must be manually keyed into the computer. Because the printers use a series of round dots to generate straight lines, the results, at best, are average in quality. They also work very slowly—up to 10 minutes per page of bar code.

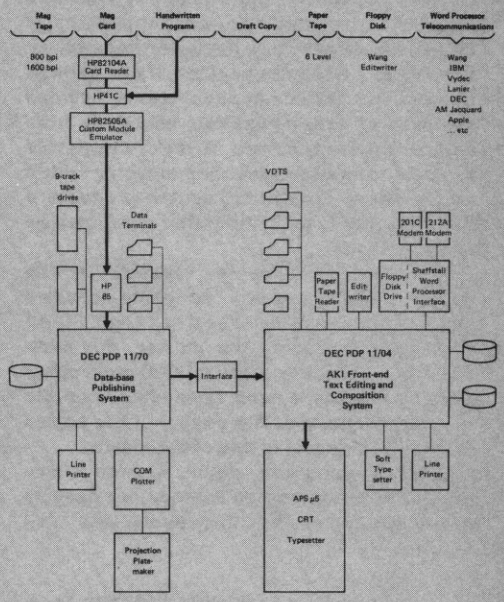


Section of Bar Code Produced With HP-85 Computer and Diablo Printer.

If the possibility of producing your own programs in bar code looks pretty bleak at this point, take heart. There is a simple, inexpensive solution to the problem. Working with Hewlett-Packard, we (George Lithograph, a San Francisco graphics firm) have developed a process that makes high-quality bar codes available at a reasonable price. Our production service will convert your programs stored on HP-41 magnetic cards to sheets of bar code. The cost for a single master of a one-card program is only $7.* Depending upon the quantity, high-quality copies from the master are 15 to 30 cents each. For a higher fee, we will also convert handwritten programs. Data/direct execution and customized programming are also available. Standard orders are processed and mailed within seven days. (Allow *at least* two weeks for overseas orders.)

This low-cost production service is the result of a unique combination of hardware and software, some of which is proprietary. The computer/typesetting system at George Lithograph and the path of the conversion process is illustrated in the accompanying diagram.

GEORGE LITHOGRAPH'S TYPESETTING SYSTEM



An order for bar code received on magentic cards is first entered into the memory of the HP-41, using a standard HP 82104A Card Reader. The contents of the HP-41's memory are then transferred to an HP-85 Computer via an SDS box (HP 82505A Custom Module Emulator). Through the use of this interface, keyboard entry is eliminated (Handwritten programs must be keyed into the HP-41 and then proofread, thus the slightly higher cost of conversion.) The HP-85 expresses the program in binary code, representing the thin and thick bars.

The program is then processed through a DEC PDP-11/70 computer, which produces commands that will cause an APS-MICRO 5 typesetter to produce the desired bar codes. These commands are sent to a DEC PDP-11/04 computer, which acts as a control center for various composition devices. Once in the composition system, the bar code program is cued for typesetting, along with other composition work in progress. This marriage of computer and typesetting hardware makes it possible to produce high-quality graphic output from computer input.

The bar code program is then shuttled to the APS-MIRCO 5 digital CRT typesetter. With this kind of typesetting device, characters are stored as digitized information, instantly accessible in any size. Instead of projecting characters through a rotating drum and lens (the method used in mechanical phototypesetting), entire lines of type are beamed onto photosensitive paper from a cathode-ray tube within the typesetter. This eliminates most moving parts, making the speed of this typesetting process as much as ten times faster than that of non-digitized typesetting equipment. At 1,250 lines per minute, the APS-MICRO 5 can generate a page of bar code in about 40 seconds.

(Continued)

DATA: 'GEORGE LITHO'

Quality isn't sacrificed for speed. When converting letter-forms or bar symbols into digitized information, each character is broken down into overlapping vertical strokes. With up to 3,615 lines per inch, the resolution is excellent. For comparison, we've included samples of first-generation bar code from several different sources. To make it easier for the user to scan the bar code and stay within its boundary, George Lithograph produces a bar code that is taller than that generated by other methods.

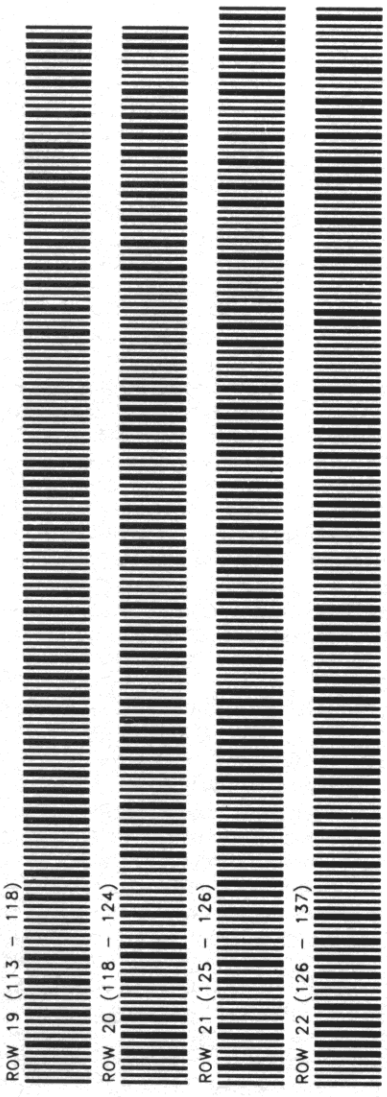
After the bar code is "typeset" by the digitized CRT typesetter on photo-sensitive paper, the paper is developed in a Log E PC-13 processor, producing the master. For additional copies, either a Xerox 9500 or an offset printing process is used, depending upon the quantity requested. The quality of the copies is virtually identical to that of the master.

Without going into detail, George Lithograph has experimented with various aspects of the printing process (exposures, inks, etc.) in order to determine the best ways to produce high-quality bar code in a number of applications, such as pressure-sensitive labels.

The production service offered by George Lithograph now makes it as easy to generate your own programs in bar code as it is to use the medium. Their prices are reasonable, especially when compared with the time and/or equipment needed to do it yourself. 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, California 94107
(415) 397-2400

* *U.S. dollars. In the U.S., $7 includes postage. For orders outside the U.S., add $3 per order for special handling and postage. Also, for orders anywhere outside the U.S., you must include a negotiable check (or money order), in U.S. dollars, drawn on a U.S. bank. For large orders or special requirements, contact George Lithograph before you order.*



Section of Bar Code, Printed by George Lithograph, from the *Real Estate* Solutions Book.

Creating Your Own Bar Code

Elsewhere in this issue is an article about how others can create bar code for you; don't miss it. However, this is about *creating* bar code *for yourself*.

On April 1, 1981, Hewlett-Packard released the manual, *Creating Your Own HP-41 Bar Code*. But before you think, "That's just what I need ...", be sure you read the balance of this article.

We have produced this manual to provide a technical base for generating HP-41 bar code. In other words, the manual supplies the information necessary for you to develop your own bar code printing capability, tailored to your specific computer printer or plotter system. The minimum system needed would be as simple as a minicomputer with a plotter or a printer. A sample system, then, would be an HP-85 Computer using an HP 7225A Plotter or an HP 2631G Matrix Printer.

Any minicomputer (or larger) with a BASIC compiler that has 16k bytes of user memory will be able to compile and run, with modifications for specific BASIC implementations, the sample software listed in the manual. The input needed to generate the desired bar code for an HP-41 program listing and/or HP-41 functions can be entered through a terminal. As an alternative input method the generation program may be altered to accept punched cards or paper tape.

If you use a plotter it must be able to create solid, dense lines at least 0.015-inch (0.381 mm) in width for a narrow bar and at least 0.030-inch (0.762 mm) in width for a wide bar. Alphanumeric capability is also desirable but not necessary.

The software in the main body of the manual was written in HP 9845A BASIC and was used to print bar code on a Diablo 1650 Daisy Wheel Printer with a "Titan 10" 96-character wheel. Flowcharts for the programs appear in appendix A, and appendix B contains the listings of the same programs adapted into HP-85 BASIC.

It is probable that you will have a hardware configuration different than that used to produce the two programs in the manual. In that case, you will have to not only adapt the language used but also rewrite the sections that actually print the bar code.

As you can tell, before you can *create* your own bar code, you need a good knowledge of programming and an intimate knowledge of the HP-41 memory architecture.

This new book should be available from your local HP Dealer by the time you read this article. The order number is 82153-90019 and the list price is $12.50* in the U.S. (probably slightly higher in overseas areas because of added shipping, taxes, and so forth). You may order it from the Corvallis Users' Library, but you will have to include a postage and handling charge of $3.50,* which covers all areas worldwide. Payment from outside the U.S. must be by International Money Order or a Foreign Draft and must be in U.S. dollars, drawn on a U.S. bank.

* U.S. dollars. See note at bottom edge of cover.

On Un-marking Cards (!)

Over the past 6+ years we have written many words about how to mark (write on) our magnetic cards. But what if you used india ink, or so-called "permanent" ink, and you now want to unmark some cards or change the markings? Well, just read this letter from **Edward R. Gabel** of Minneapolis, Minnesota, and you'll have an answer to the question.

"Regarding the marking of magnetic cards with a capillary pen and permanent ink for film, there *is* a way to erase that permanent ink.

"KOH-I-NOOR® Rapidograph® makes a Rapidraw Eraser Kit that will remove all of the permanent ink if the instructions in the kit are followed. I have erased the ink from the same card several times and with excellent results.

"The kit number is 290S and is available at any art supply store. Maybe this would be of interest to other KEY NOTES readers."

®Registered trademarks of Kohinoor Rapidograph, Inc.

HP-41 Flags—Part 2

Part 1 of this article appeared in the last issue of KEY NOTES (September-December 1980, Vol. 4 No. 3). General flag concepts and flags 00-29 (except flags 12, 13, and 21) were covered. This part of the two-part article covers all remaining flags.

Part 1 contained a table (table 1) that listed all HP-41 flags, their name, function, and status at turn-on. **William C. Tempelmeyer** of Evanston, Illinois, wrote to point out that the fix 4 mode is the default mode after Master Clear. This means that the "Status at Turn-on" column should be M,3 for flags 37 and 40 instead of M,1 as shown.

Now, let's continue our flag discussion with the printer-associated flags.

Flag 12. Flag 12, if set, instructs the HP 82143A Printer to print double wide. This is useful for special characters, emphasis, titles, etc. This flag is cleared when the HP-41 is turned off. For this reason, it is a good practice to set flag 12 whenever double wide printing is desired rather than only once at the beginning of the program. If a program is stopped and the HP-41 is turned off, or the machine "timed out" and turned off, the output may not be as expected when the HP-41 is turned on and program execution is resumed.

Flag 13. Set flag 13 to print in lower case. Flag 13, like flag 12, is cleared when the HP-41 is turned on.

Flag 21. This flag gives the user control of the printer and its automatic response to VIEW and AVIEW instructions. Flag 21 is automatically set when flag 55 is set. But, before flag 21 is described in detail, let's examine flag 55.

Flag 55. This flag is set whenever the printer is plugged in. If the printer is not plugged in when the HP-41 is turned on, flag 55 and flag 21 are cleared. When the HP-41 "discovers" that the printer is connected (either on or off), flag 55 is set. Flag 21 is set when flag 55 is set. You do *not* have any control over flag 55, but you may set and clear flag 21 as desired.

The *HP 82143A Printer Owner's Handbook* shows the set and clear conditions of flags 21 and 55. Perhaps the most confusing situation arises when AVIEW is used in a program and the printer is connected but not turned on. When this occurs, program execution stops at the AVIEW instruction. The unwary programmer may search for nonexistent bugs if he or she is not aware of this "problem." The solution is to turn on the printer and press R/S. You can verify that program execution has stopped because the PRGM annunciator is off. Sooner or later every HP-41 user will experience this situation.

The use of flags 21 and 55 must be carefully planned and tested if the desired combinations of display, printed outputs, or both are to be obtained. A simple example illustrates this. Suppose you wish to write a short program that computes the value of Y, given the input value of X, with the two being related by the equation $Y = 2X + 3$. A program to do this is shown in figure 1.

```
01 LBL "Y DEMO"
02 "INPUT X?"      } Display prompt, stop
03 AVIEW           } for input.
04 STOP
05 ENTER           }
06 +               } Compute output from
07 3               } value of input.
08 +               }
09 "Y= "           } Display answer
10 ARCL X          } "label" with
11 AVIEW           } computed value.
12 RTN
```

Figure 1. Simple program illustrating use of AVIEW. Printer will print ALPHA register if connected and flag 21 is set.

If the program in figure 1 is executed without a printer, it will display "INPUT X?" and stop. If a value is keyed in and R/S is pressed, the HP-41 will stop with the computed value, preceded by Y =. This is as expected. Try X = 3 as an input and see 9 as an output (Y = 9.0000). Now, plug in a printer. Do not turn on the printer. Execute the program. (Assign "Y DEMO" to a key). Key in 3 as before and press R/S. Nothing happened! Switch to PRGM and see ENTER, line 05. Backstep once to see the instruction that caused the stop. It is STOP! Repeat. When the display shows "INPUT X?" switch to PRGM and press BST. See that AVIEW, line 03 caused the program to stop before it was supposed to. AVIEW WILL CAUSE PROGRAM EXECUTION TO STOP IF THE PRINTER IS CONNECTED AND TURNED OFF! Now, repeat this procedure with the printer connected, turned on, and in MAN mode. Key R/S with 3 as an input. See the printout shown below in figure 2(A).

```
INPUT X?          XEQ "Y DEMO"
Y = 9.0000        INPUT X?
                        3.0000    RUN
                  Y = 9.0000
   (A)                  (B)
```

Figure 2. Printed outputs obtained by running the program in figure 1. (A) is MAN mode. (B) is NORM mode.

Now, what if you want to print the input prompt, input value, and the Y output value? One method is to add a VIEW X after line 04, following the stop. Another method is to set the printer to NORM mode. See (B) in figure 2. If the VIEW X instruction is added and the printer is not connected, the display will briefly show the input value before stopping with the computed output displayed. If the printer is connected and turned off, two stops will be encountered, one for each VIEW or AVIEW.

How can this simple program be made to work properly if the printer is used? Well, most of you learn to turn off your printer when it is not in use, so you can save battery energy. But this practice may cause unnecessary stops when AVIEW or VIEW is used. And AVIEW is a useful instruction that allows programs to run with and without a printer. So try replacing the first AVIEW in figure 1 with PRA. If the printer is off, a display of PRINTER OFF results. Now, turn on the printer, press R/S, and the program will print as desired. This technique has corrected one "problem"—stopping at AVIEW or VIEW if the printer is off—but has created another problem. If the printer is not connected, the program will stop, showing NONEXISTENT at the PRA instruction.

Whenever you want a specific set of features, you usually write your own program or routine to do things the way you want them done. The following routine was written to replace AVIEW and has the features of:

a. No Printer—simply AVIEW.
b. Printer is off, and the HP-41 displays PRINTER OFF. You merely turn on the HP-41 and press R/S to print and display the ALPHA register.
c. Printer is on, and it prints and displays the ALPHA register.
d. Flag 21—Flag 21 does not control the printer and retains its set or clear status.

```
01 LBL "AV"
02 FS? 21      Uses flag 14 to store status
               of flag 21.
03 SF 14
04 FC? 55      Test for printer being
               connected. Go to label 14 if
               not.
05 GTO 14
06 SF 21       Insure printer prints.
07 PRA         "Tests" for printer off.
08 LBL 14
09 CF 21       Does not print.
10 AVIEW       Displays ALPHA without
               printing.
11 FS?C 14     Restores flag 21 to original
               status.
12 SF 21
13 RTN
```

Figure 3. AVIEW routine using "flag logic" to avoid ambiguous program stops when using AVIEW and an "OFF" printer.

The routine shown in figure 3 was written to aid users who normally operate their HP-41 system with the philosophy that if their printer is connected it should print, and they should be reminded to turn it on if it is off. To use this routine with the routine in figure 1, simply change lines 03 and 11 to XEQ AV.

The routine in figure 3 also illustrates another use of flags. Lines 02 and 03 use flag 14 (the flag that allows you to record on a clipped corner card) to store the status of flag 21. Lines 11 and 12 restore both flags to their original status. There is little danger in using flag 14 in this way because it is very unlikely that you will stop the routine to record on a clipped-corner card. This technique is useful in the situation that flag 21 is left in an unknown state by a ROM routine. If flag 21 is left cleared by a ROM

(Continued)

routine, there is also another flag sequence that may be useful in your programs. Use the sequence shown in figure 4 at the beginning of any RAM program that has called a ROM routine.

```
CF 21
FS? 55
SF 21
```

Figure 4. Flag sequence that insures flag 21 is tracking flag 55 when ROM routines leave flag 21 clear.

The routine in figure 3 is useful for many situations and for system operating philosophy, but there is still a situation where the user is inconvenienced. Suppose the printer cable is accidently pulled out, leaving flag 55 set and no printer. Routine execution causes a STOP. The routine in figure 5 will *never* cause a program stop. If all is well with the printer, it prints and displays, otherwise it only displays using AVIEW.

```
01 LBL "AVN"
02 FS? 21      Set flag 14 if flag 21 is set.
03 SF 14
04 SF 21
05 SF 25       Set error flag in case no
               printer.
06 PRA         Print if printer can.
07 CF 25       Clear flag if everything is
               "OK."
08 CF 21       Insure that AVIEW
               doesn't print.
09 AVIEW       Display ALPHA register.
10 FS?C 14     Restore flag 21 to original
               status.
11 SF 21
12 RTN         (30 Bytes)
```

Figure 5. Non-stopping AVIEW routine that prints if the printer is able. Flag 21 status is preserved.

There are many ways to use flags 21 and 55. The examples shown above illustrate a few of these. Also, when programs are designed to print differently than they display, flag 21 is heavily used.

Flags 30-35. These flags provide very little helpful information to the user in normal programming applications.

Flags 36-39. These four flags may be tested to determine the display setting. They also may be used as "control logic" flags, with byte savings obtained by having one instruction control several flags. The first example illustrates the use of flags 36-39. The routine shown in figure 6 determines the display setting. The operation of the routine may be illustrated with the help of figure 7.

```
01 LBL "DSP"
02 0   }
03 +   }  Clear LAST x.
04 1   }
05 FS? 39 }
06 ST+ L  }  Add 1 to LAST x if flag 39
          }  is set.
07 2   }
08 FS? 38 }
09 ST+ L  }  Add 2 to LAST x if flag 38
          }  is set.
10 4   }
11 FS? 37 }
12 ST+ L  }  Add 4 to LAST x if flag 37
          }  is set.
13 8   }
14 FS? 36 }
15 ST+ L  }  Add 8 to LAST x if flag 36
          }  is set.
16 LAST X }
17 FIX 0  }  Recall accumulated
          }  binary weights and
          }  display the display
          }  setting.
18 RTN
```

Figure 6. This routine tests flags 36-39. If set, the binary weight is summed in the LAST *x* register and displayed.

Display Digits	Flag 36	37	38	39	Notes
0	0	0	0	0	0 is clear.
1	0	0	0	1	1 is set.
2	0	0	1	0	Observe that the
3	0	0	1	1	display digit setting
4	0	1	0	0	forms a 4-bit binary
5	0	1	0	1	counter when the flags
6	0	1	1	0	are arranged as
7	0	1	1	1	shown.
8	1	0	0	0	
9	1	0	0	1	
Binary Weight	8	4	2	1	

Figure 7. Flags 36-39 define the number of display decimal digits with a binary relationship as shown.

The DSP routine in figure 6 illustrates how the test-only flags may be used to obtain information for program use. The number of digits set could be used to "store" the display setting. Test by FIX 4, XEQ DSP and see 4 in the display. Execute the routine again and see 0 because of the FIX 0 at the end of the routine.

Flags 40 and 41. These flags control the display mode. They may be tested in a manner similar to that shown in figure 7 to accumulate their values using the binary weight method. These concepts are carried through to make a program that stores and recalls the display mode. (See figure 8.) Register 01 is used to store the display mode. In this program, flags 40 and 41 are stored to the right of the decimal point with simple data packing. This program is not optimized, but provides a practical example of using the test-only flags. The STO and RCL routines allow the user to store an unknown display mode, run a routine that sets the display for its purpose, and then return to the original display mode.

Flags 42 and 43. These are the angular mode flags. They may be tested and stored in a manner similar to that used for flags 36-39. It is most useful to "set" the correct angular mode when using the TRIG functions. As an exercise, you might want to write a pair of routines that store and recall the angular mode. Figure 9 will help.

```
01♦LBL "STO"
02 0
03 +
04 1
05 FS? 39
06 ST+ L
07 2
08 FS? 38
09 ST+ L
10 4
11 FS? 37
12 ST+ L
13 8
14 FS? 36
15 ST+ L
16 .1
17 FS? 41
18 ST+ L
19 .2
20 FS? 40
21 ST+ L
22 LASTX
23 STO 01
24 CLX
25 RTN
26♦LBL "RCL"
27 RCL 01
28 RCL 01
29 FRC
30 10
31 *
32 XEQ IND X
33 X<> Z
34 RTN
35♦LBL 00
36 SCI IND Y
37 RTN
38♦LBL 01
39 ENG IND Y
40 RTN
41♦LBL 02
42 FIX IND Y
43 RTN
```

Figure 8. Store display mode and recall display mode routines that utilize test-only flags 36-41.

Display Mode	Flag 40	41	Angular Mode	Flag 42	43
SCI	0	0	DEG	0	0
ENG	0	1	RAD	0	1
FIX	1	0	GRAD	1	0
INVALID	1	1	INVALID	1	1
Binary Weight	2	1	Binary Weight	2	1

Figure 9. Extension of figure 7 that includes flags 40-43.

Flag 44. This continuous-on flag should be set if the HP-41 is expected to be "standing by" for periods greater than about 10 minutes. This flag is set by keying XEQ ON. Because this instruction is not programmable, flag 44 should be tested, if clear, and display "XEQ ON" as a reminder to the operator.

Flags 45-47. These flags have little application, because they always test clear.

Flag 48. This flag could be tested to determine if an alpha input is used or a numeric input is used. A program that converts to/from hexadecimal could test flag 48 to determine if the input is to be taken from the ALPHA or X-register.

Flag 49. This flag is set and the BAT annunciator turned on in the display when the battery supply voltage gets low. When flag 49 is set, the card reader motor will not turn on. The sequence FS? 49, OFF is useful to protect program and data memory if long-running programs cause the batteries to be discharged. Long-running programs should contain the flag 49 test, off sequence as good programming practice.

(Continued)

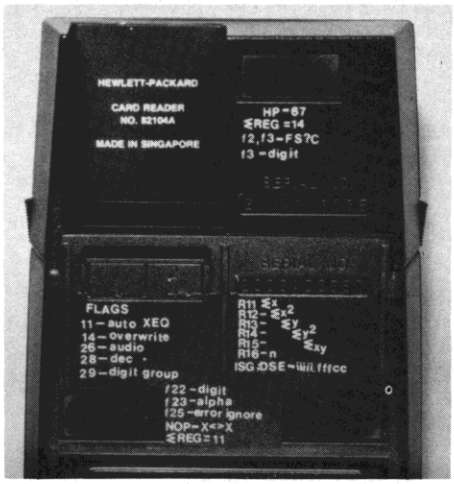
Flags 50-54. These flags always test clear and have little use in programs.

The HP-41 provides a wide variety of general-purpose and dedicated flags. Unfortunately, this brief discussion of flags could not cover all possible applications. For example, some clever programmers use flags as decision trees to provide multiple paths through a program. Three flags, for instance, could control eight different situations! Also, flags are widely misused and are not always well understood. However, if properly used, flags are powerful and memory-efficient instructions. You might not be able to master them just by reading this two-part article, but we're sure that a *second* reading will help you and, when you've become a flag *aficionado*, you can write some really clever programs.

HP-41 Homemade Overlays

When you work in an engineering group that is concerned with "human factors," you tend to have ideas about how *you* would like to see things made more convenient to use or operate. Well, such a person is **Art Leyenberger**, who works for Bell Laboratories in Whippany, New Jersey. In the accompanying photos you can see what he's done to make the HP-41 easier and quicker to use. Here's his letter.

I am enclosing two 35 mm slides that show the front and back of my HP-41C. The overlay was first covered with aluminum-colored spray paint. The black labels are the normal mode functions for the keys; the blue labels are the reassigned functions used in USER mode; and the red labels are for program names that have been assigned to a particular key. The high contrast of the colors really facilitates locating functions.



The back of the calculator has yellow rub-on letters showing some information that I most often forget, including some information relating to the HP-67 from the standpoint of program conversion.

(Thanks for the photos, Mr. Leyenberger. Too bad we couldn't print them in color, but I think our readers will be able to "see" the idea. Mr. Leyenberger later sent another sample overlay with black, blue, green, white, and red labeling. It can be described with only one word: superb. The rub-on lettering is covered with a clear plastic spray to protect it, making a very professional-looking job. Try it; you'll find the results both rewarding and satisfying. —Ed.)

Roof-Coating by HP-41C!

We recently received a letter from **Glenn Lindsay**, director of construction for the L.E. Lindsay Construction Company of Granite City, Illinois.

"Enclosed is a recent mailing from the Midlands Protective Coatings Company to existing or potential distributors of their roofing products. It outlines an intense application of the HP-41. (Yes, even the construction industry can put the HP-41 through its paces.) Thought you might appreciate the news.

"Whether in hardcore engineering or extraneous construction applications, the HP-41 has repeatedly proven itself to be a personally valued asset—not to mention an excellent tool."

The mailing then goes on to describe how, using the "Rapid Roof System Energy Evaluation" program, you can calculate the amounts of solar radiation absorbed and reflected by a surface during a day, and the amount of solar radiation that gets transferred, in the form of heat, through a roof or wall section into a building during a day. Midlands Protective Coatings Company uses the HP-41 and an evaluation form (filled-in by the client) to show prospective customers—in minutes—how much they can expect to save in air conditioning costs by applying the Rapid Roof System in lieu of conventional roofing. They can also show how much can be saved in heating costs in winter months by the use of extra insulation. The Evaluation Program is copyrighted by Conklin Company, Incorporated.

Midlands also offers for sale to their distributors the fully programmed HP-41, which shows you that this calculator has a nearly endless variety of applications.

Changes to "Taxes" Book

Since the introduction of the HP-41 Solutions Book, *1980 Taxes*, certain errors have come to our attention. They are:

1. Page 50: In the example problem "SX" should read "SY" and SIZE should be set to 049.
2. Page 59: The amount on line 16 should be $0.00, not $123.00 and, resultingly, the amounts on lines 19 and 20 should be $1500.00 and $12,250.00 respectively.
3. Page 60: The amounts on lines 3 and 6 should be $0.00, not $123.00.
4. Page 61: Program lines 27 x < 0? and 28 CLx should be deleted and the keystrokes RCL22, −, X < 0?, CLx should be inserted between the current lines 85 RCL 03 and 86 2.
5. Page 65: The keystrokes x = 0? and GTO 01 should be inserted between the current steps 15 RCL 21 and 16 +, and LBL 01 should be inserted between steps 32 CLx and 33 "ADJ DEDS".
6. Page 12 of bar code: The bar code for program SX is incomplete. For a complete bar code listing or further information, write or phone:

Users' Library
Hewlett-Packard Co.
1000 N.E. Circle Blvd.
Corvallis, OR 97330
Tel: (503) 757-2000

Custom Services Sells Solutions

Does your organization or company have customers or employees who perform time-consuming repetitious calculations? Have you ever dreamed of having a personal calculator that will give you answers to long tedious problems at the touch of a button or two? Well, *now* you can simplify your calculations and get quick answers ... when and where *you* need them!

The new Hewlett-Packard Custom Services Program produces firmware (custom ROM's and custom magnetic cards) from customer-supplied software. The firmware tailors HP-41 Calculators and the HP-67/97

(Continued)

series (mag cards only) to *your* applications and dedicates the calculator to *your* special needs. Custom bar code, printed by George Lithograph (see article in this issue), also can be used to customize the HP-41.

With customized calculators, mistakes are minimized, speed is increased, and productivity is improved.

Custom keyboard overlays provide the final professional touch that personalizes the calculator to *your* application or organization.

Custom Services is not for everyone, because custom firmware comes in minimum quantities of 250. But if *your* organization or firm requires portability and on-the-spot solutions to problems, then maybe Custom Services can help you.

If you would like more information about this exciting new program, call us toll-free at (800) 547-3400 excluding Alaska and Hawaii. (In Oregon call 758-1010.) Or you can write to:

HP-41C Custom Services
Hewlett-Packard Company
1000 N.E. Circle Blvd.
Corvallis, OR 97330 U.S.A.

Editorial

If you don't read anything else in this issue, make sure you read the entire article entitled: "KEY NOTES Going to Subscription." It appears on page 3. More about that subject will appear in the next issue.

Bigger and Better ...

Although the October 1977 KEY NOTES (V1N3) was a 16-page issue, four of those pages comprised an order blank and lists of Library Solutions Books. So, in truth, this issue is another landmark for KEY NOTES: the first 16-page "all KEY NOTES" issue. Hope you like it.

That Misleading Pause

On page 7 of the last issue, the article entitled: "The Pause That's Misleading," gave me more pain than a swift kick to the knee-cap. For reasons unknown to man or machine, our automatic, computer-driven typesetting equipment garbled a few lines, added a word or two, and effectively ruined a perfectly good article. So, to make a long story short, the next-to-last paragraph on page 147 of the HP-41C *Owner's Handbook and Programming Guide* should read: *Pressing any other keys during a pause, that is, any keys not associated with data entry, causes the pause to terminate and program execution halts. The pressed function is executed.* If your handbook isn't printed exactly that way, be sure to change it to the correct statement. And you can bet your last dollar that—this time—I'll check that statement *before* it goes to press.

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, 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.

Some Good News

You may find it hard to believe, but this entire KEY NOTES operation, since its inception, has been a one-person project. At times—especially of late—I've wondered if an issue would *ever* get out the door. So we are adding an assistant very shortly, and that will make things operate more smoothly in the future. Not to mention getting the newsletter out on time! I'll bet *that* won't make any of you unhappy. So please be patient a wee bit longer, as we are doing everything possible to make your KEY NOTES bigger, better, and more punctual. For example, the last issue was being *mailed* throughout Europe *before* it even got to the East Coast in the U.S.

Routines, Techniques, Tips, Et Cetera . . .

If you are looking for the column, "25 Words" (More or Less!), you have found it. Okay, why did we change the name of this ever-popular column? Well, many people misunderstood the former title, and so they struggled like crazy to keep their contributions at or under 25 words. Or they went way over 25 words and then requested an editing job. So we decided to eliminate the restrictions imposed by the title. We'd rather have more words and more "understanding" than a problem. Ergo: a new title!

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 always 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 first contribution is from **David G. Motto**, who lives with an HP-41 in Jackson, Michigan, and here is what he does with it.

(41) How about this little (!) routine to clear out a range of registers? Key in the starting register, press ENTER, key in the number of registers you want to clear, and XEQ "CL". This uses a programming hint from **Ernesto A. Malaga**, to whom I am indebted. The routine also clears the stack and resets ΣREG to R_{01} if the number of registers cleared is greater than six. LBL 01 is used only for small clearing operations (less than six registers). The routine uses 65 bytes, and it *is* big, but it is a general-purpose routine and quite fast.

```
01♦LBL "CL"      24 ΣREG 01
02 ENTER↑        25 CLST
03 ENTER↑        26 RTN
04 6             27♦LBL 01
05 X>Y?          28 X<>Y
06 GTO 01        29 R↑
07 MOD           30 +
08 ST+ Z         31 LASTX
09 CLX           32 1
10 LASTX         33 -
11 -             34 1 E3
12 LASTX         35 /
13 /             36 +
14 INT           37 1
15 6             38 -
16♦LBL 00        39 0
17 ΣREG IND Z    40♦LBL 02
18 CLΣ           41 STO IND Y
19 ST+ Z         42 DSE Y
20 DSE Y         43 GTO 02
21 GTO 00        44 CLST
22 ΣREG IND T    45 END
23 CLΣ
```

There are several ways to calculate the sum of all digits of a number, but this routine seemed clever enough to bring it to your attention. It is the creation of **Ralf Pfeifer** of Köln, Germany.

(41) Problem: Calculate the sum of all digits of any number. If the sum is greater than 9, calculate the sum of *this* number, and so on. Example: 14307; the sum of the digits is 1+4+3+0+7=15; and 15 is greater than or equal to 10, so calculate the sum of the sum: 1+5=6. A quicker solution is offered by the following routine. If only integer values are keyed in, lines 02 through 08 can be left out.

```
01♦LBL A         09 9
02 ENTER↑        10 MOD
03 LOG           11 X≠0?
04 INT           12 STOP
05 9             13 CLX
06 -             14 9
07 10↑X          15 RTN
08 /
```

In Vol. 4 No. 2 (page 11, column 1), **Nai Chi Lee**, formerly of Singapore and presently in New York, gave us a few tips on finding and saving bytes. Here's another tip from him on how to *count* bytes.

(41) For those of us who do not own an HP-41 printer, the following procedure can be used to determine the exact number of bytes in a program.

1. Execute packing (GTO..).
2. Switch to PRGM to see the number of program registers remaining.
3. Compute: X = original SIZE + registers remaining − 1.
4. Execute SIZE X.
5. Switch to PRGM and repeatedly key in a single-byte operation (such as SIN, LN, etc.) until the TRY AGAIN message is displayed. Note the last line number (n) before clearing all the lines.
6. The number of bytes (B) in the program is: B = total bytes available − 7X − n.

For example, a basic HP-41C has 445 bytes. If X = 35 and n = 9, then B = 445 − 245 − 9 = 191 bytes. *(Note that this method is not valid if there are any Catalog 2 or Catalog 3 key assignments!—Ed.)*

Not what you could call a "well-known" place is Brondby Strand, Denmark, but it *is* the home of **Jon M. Jonsen**, who contributed the following idea.

(67/97) When I first read about the HP-41C (truly irresistible!) and noticed that the status of some flags is displayed, I recalled something I discovered years ago on my HP-25, a trick that can be used to check flag status manually on the HP-67; namely: When a test proves false in a running program, the program counter (PC) advances an extra step, thereby deleting execution of the step immediately following the test. It happens that, if you press the test function in run mode, the PC will do that extra increment if the test proves false. So, for example, to test flag 0:

- Switch to W/PRGM.
- Notice current PC setting (nnn).
- Switch to RUN.
- Press hF? 0.
- Switch to W/PRGM.
- PC = nnn, means F0 set.
- PC = nnn + 1, means F0 clear.

Notice that F2 and F3 will also be cleared when tested manually.

Here is a suggestion about an article that appeared in Vol. 4 No. 2. It is from **Richard H. Hall** of Washington, D.C., whose HP-41 contributes to running the U.S. Government, through the Office of Personnel Management.

(41) I think I can suggest a slight improvement to your "clear assignments" procedure (V4N2P7).

1. "Master Clear" (same as yours).
2. Assign any function to any key, say [×] to itself (essentially the same as yours).
3. Add to step 2:
   a. Clear this assignment. (At this point, no keys are assigned and PRKEYS will verify this, but (for reasons known only to the microprocessor), one key assignment register is still "occupied" and unavailable to you.)
4. (*Without* PACKing) XEQ WSTS.
5. Feed track 1 of the card (same as yours). You *will* get RDY 02 of 02, even though there are no assignments, because of that still-"occupied" register.
6. Feed track 1 again (same as yours). The status card so produced can be used exactly like yours (read track 2 only and back-arrow key away the TRK 01 prompt) but no key needs to be cleared. However, the "occupied" register will remain unavailable until you either turn the machine OFF then ON, or PACK (with either PACK or GTO...) *or* the HP-41 itself starts PACKing because it has run out of room. If (like me) the average users need the clear assignment card primarily for those occasions when they run out of space while keying in a program, they need do nothing other than feed in this card. Thereafter the machine will automatically PACK (and free up that register) as soon as it runs out of room again.

(About that "occupied" register, Mr. Hall; you are overlooking something. When you assign a function even to itself—same key, that is—you still are using a register. You do not *clear that register unless and until you PACK. The HP-41, you see, is a lot smarter than we think it is. It* always *knows what is going on, even when* we *don't!—Ed.)*

Now we bring you a clever routine from **Steve Hageman**, and we would *guess* that he is an engineer. We *do know* that he lives in Vallejo, California.

(41) Below is a short routine that might be useful to anyone (for example, engineers) who must select between linear or log plots or printouts.

The sample program asks for a start frequency ("FSTART") and the frequency increment ("FINCR"). If the FINCR is positive, a linear sweep is indicated and FINCR is used as an additive increment (that is, FINCR is added to FSTART each time LBL00 is run). If FINCR is entered as a negative number, a log sweep is indicated and the absolute value of FINCR indicates the number of points per decade to be input (that is, $10^{1/FINCR}$ is used as a multiplicative increment).

Gone are the long hours (and keystrokes) required to calculate the number of points per decade when the old method is used (that is, HP-41 Circuit Analysis Pac, "GNAP"). The sample shown in the Circuit Analysis Pac book shows the basic algorithm used. Of course, this algorithm must be modified for use in other programs, but the idea is the same.

```
01♦LBL "SAMPLE"   14 10↑X
02 "FSTART?"      15 STO 01
03 PROMPT         16♦LBL 00
04 STO 00         17 "F="
05 "FINCR?"       18 ARCL 00
06 PROMPT         19 PROMPT
07 CF 00          20 RCL 01
08 X<0?           21 FS? 00
09 SF 00          22 ST* 00
10 ABS            23 FC? 00
11 FS? 00         24 ST+ 00
12 1/X            25 GTO 00
13 FS? 00         26 END
```

Over in England (or is it Great Britain?) there is an HP-67 living in East Essex. Its owner, **P. A. Maillard**, has it toggling rather nicely. To wit:

(67/97) This subroutine gives the option of entering a function either as a set of data points or as an equation. The toggle key "fA" sets the calculator in "data mode" or in "function mode." When the "function mode" is selected, the program will stop at the right line for keying in the function's equation. This equation can be checked by calling "fE." The main program places the xi value in the X-register; GSB1 will call the corresponding yi. In "data mode" the display will flash xi until yi is entered, then the program will continue. In the "function mode," xi will pause for a progress check. If the progress check is not wanted, F?1 and GTOe should be placed immediately after LBL1.

```
Main program
CF3
.
.
GSB1        call yi
.
.
RTN
LBLa        f/data toggle
CF1
CLX         "0" → data
RTN
LBLa
SF1
EEX         "1" → function
GTO2
LBL1        yi SUB
F?3         RTN after
RTN         entry
PAUSE
F?1         JUMP if
GTOe        function mode
GTO1
LBL2        W/PRGM
R/S         STOP
LBLe        function SUB
RTN
```

(I think those values should be written x_i and y_i, but I printed the letter as written—Ed.)

Now, here's a routine that relates back to the last issue. It is from **Walter M. Miller, Jr.** of Daytona Beach, Florida.

Dr. Schmitz' routine for the interconversion of decimal and almanac formats (Vol. 4, No. 3, p. 11) is short enough, but is not a perfect analog of the built-in monadic functors HR and HMS, inasmuch as the latter do not affect the contents of the Y, Z, and T registers, and they preserve LAST X. The following routine, which I have been using for some time, interconverts navigator's format (degrees and decimal minutes, DDD.MMM) and decimal format (DDD.DDDD) without pushing the stack; it also preserves LAST X. The only disadvantage is the use of one storage register.

(Continued)

```
01♦LBL "DD"   11 FRC
02 FIX 4      12 ST* 00
03 SF 00      13 CLX
04♦LBL "DM"   14 LASTX
05 STO 00     15 INT
06 CLX        16 ST+ 00
07 .6         17 X<> 00
08 FS? 00     18 FC?C 00
09 1/X        19 FIX 3
10 X<> 00     20 RTN
```

Example: The altitude of the center of the sun at today's LAN for this locality is 43°33'4. To express as a decimal: XEQ "DD." See 43.5567. (LAST X: See 43.3340.) Registers Y, Z, and T remain undisturbed as in a monadic functor. The information-cost of the routine is 40 bytes plus a storage register. The cost of Dr. Schmitz' routine is 36 bytes, not 29 as stated.

PS: The Schmitz routine loses not only T but Z as well! (I also suspect his routine as published by you contains a typographical error.)

(I could find but one error, Mr. Miller. When Dr. Schmitz stated his routine used only 29 bytes, he had an asterisk at that point. Then, where he stated that several lines could *be omitted, that was supposed to be a footnote. I believe he was correct, and I could not find any other errors—Ed.)*

Let's now travel to Merry Olde England to see what **John van Rossum** of Birmingham is doing with his HP-41C.

(41) Whilst reading through "25 Words," I was interested to see a routine for clearing data registers. But I found it rather difficult to use and noticed that it did not use the calculator's alphanumeric capabilities. In view of this, I have written my own version.

The difference is that my routine clears all the registers between, and including, two limits. Register 00 is cleared individually and should not be used as one of the limits.

```
01♦LBL "MOP2"   13 ST+ 00
02 0            14♦LBL 99
03 STO 00       15 0
04 "START?"     16 STO IND 00
05 PROMPT       17 ISG 00
06 ST+ 00       18 GTO 99
07 "END?"       19 0
08 PROMPT       20 STO 00
09 1 E3         21 TONE 9
10 /            22 "READY"
11 1 E-5        23 PROMPT
12 +            24 RTN
```

Another input from England tells us not how to *do* something but how to *undo* something. It is from **J. Hartland** who lives in Cornwall.

(41) With reference to your remarks in KEY NOTES V4N2P7 about the danger of reading incomplete sets of HP-41 WALL cards, thus causing MEMORY LOST, I decided that used sets of WALL cards are too dangerous to have around.

Use of the following routine, DEWALL, recorded with flag 11 set for automatic execution, helps quickly and safely to overwrite WALL cards with the minimum amount of harmless DATA.

```
01♦LBL "DEWALL"   07 "DEWALL CARD"
02 0              08 AVIEW
03 STO 00         09 WDTAX
04♦LBL 01         10 GTO 01
05 FS? 00         11 .END.
06 SF 14
```

Instructions for use are:

1. If the card is "clipped," set flag 10.
2. Key GTO.. (HP-41 packs).
3. Insert DEWALL card (HP-41 prompts "DEWALL CARD").
4. Insert first WALL card (HP-41 overwrites and prompts for another WALL card).
5. Insert any other WALL cards.
6. Key XEQ ALPHA CLP ALPHA ALPHA ALPHA (HP-41 clears DEWALL program and packs).

Note that reading the DEWALL program card only lifts the stack and clears register 00. Accidental reading of the overwritten DATA cards only clears the contents of register 00.

Moving on to Fulda, West Germany, here is some feedback about a routine on page 11 of the last issue (V4N3). It was contributed, then, by **Karl-Ludwig Butte**, and so is this.

(67/97) Thank you very much for printing my contribution to the "25 Words" column, that ON X GOTO ... instruction, in the last issue. You are right that there will be an endless-loop if X ≤ I. To eliminate this and to shorten the routine and execution time, here is a corrected ON X GOTO-routine.

001 LBL A
002 STO I
003 GTO (i) or GSB (i)
004 RTN

If necessary, there could be an address calculation algorithm inserted between LBL A and STO I. An address-test is also possible to branch to an error-handling routine for false or inexecutable jumps.

Now, here is a clever little routine to relieve the monotony of seeing the "flying goose" during program execution. We have received many routines lately that do pretty much the same thing, but we picked the one submitted by **Patrick Shiblí** of Berneck, Switzerland. There are many, many variations ...

(41) While working on my HP-41C, I discovered a trick to simulate the "eagle cursor." My demonstration routine first sets error ignore flag 25. After loading the AVIEW register, it causes a built-in error in line 05! During execution, the contents of the display now scroll to the right every time the HP-41C passes a label. This movement works comparable to the "eagle" that flies by in programs without (A)VIEW functions.

I use this routine to simulate movement in calculator games or to make long-lasting programs seem a bit shorter. I hope you like this routine and find a place for it in your excellent newsletter.

```
01♦LBL "↑"    06♦LBL 00
02 SF 25      07 BEEP
03 "WORKING"  08 GTO 00
04 AVIEW      09 RTN
05 GTO "?"
```

(We did *like it, and we found an interesting feature you didn't mention. During operation, press R/S to stop the "show" in the display. Then press R/S again. The "eagle"—or as we call it, the goose—is back again, only "honking" this time! Also, line 03 can be many, many things, limited only by your imagination. —Ed.)*

For a change of pace (*I couldn't resist the pun!*), here is a routine especially helpful for those who own an HP-41 and seem to "override" the highway speed limit, and who don't have a stopwatch to use to check their speedometer—or a radar detector in case they forget! It is the contribution of **Thomas M. Pace** of Pensacola, Florida.

(41) Try this program the next time you go on a driving trip and have your HP-41 along. XEQ "A" at a mile post and then XEQ "B" at the next one.

To find what the constant in line 17 would be, I let "A" run for exactly one minute, then pressed R/S. I then recalled register 00 and divided it into 60. I was surprised to see it come out to an exact 1.5.

```
01♦LBL "TIMER"      13 RCL 00
02♦LBL A            14 10000
03 "DRIVE SAFELY"   15 /
04 AVIEW            16 HR
05 0                17 1.5
06 STO 00           18 *
07♦LBL 00           19 1/X
08 PSE              20 "AV="
09 1                21 ARCL X
10 ST+ 00           22 "├ MPH"
11 GTO 00           23 AVIEW
12♦LBL B            24 RTN
```

To make the routine continuous, insert a LBL 01 after line 04 and a GTO 01 just before the END statement. Use "A" to start and "B" at all mile posts.

There are several telephone timer programs in existence, but most of them are quite long and complex so that they account for the various time and rate problems. This routine was sent to "25 Words" last September by **Peter Linlor** of Mountain View, California. While it isn't as sophisticated as others, we thought you'd like it.

(41) Here is a simple routine for dynamically displaying long-distance telephone time and charges. At the prompt, store the first-minute rate in register 01 and the additional-minute rate in 02. The timer starts with R/S, updating the display every second. Accuracy is 0.2 percent until the tens digit of the "charges" appears and slows the formatting. Also, a connected printer will decalibrate the timer. Afterwards, elapsed time is in register 00 and charges (tax not included) are in 01, or you can check the ALPHA register.

Now, when you "reach out and touch someone," you'll know how long you touched!

```
01♦LBL "LDPT"      20 X≠0?
02 CLRG            21 GTO 01
03 CF 05           22 RCL 02
04 FIX 2           23 ST+ 01
05 .01             24 ISG 04
06 STO 03          25 GTO 01
07 .009            26 SF 05
08 STO 04          27♦LBL 01
09 "LOAD AND RUN"  28 CLA
10 PROMPT          29 "├ "
11♦LBL 00          30 FC? 05
12 +               31 "├0"
13 +               32 ARCL 00
14 +               33 "├ $"
15 RCL 00          34 ARCL 01
16 RCL 03          35 AVIEW
17 HMS+            36 GTO 00
18 STO 00          37 RTN
19 FRC
```

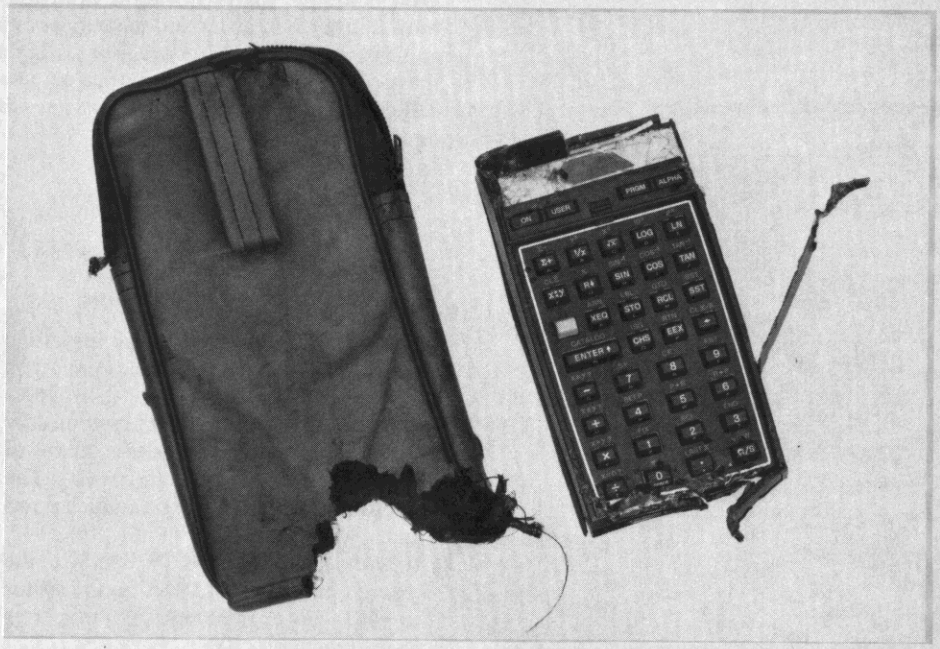
machine was completely unharmed after such 'a condition that can traumatize the calculator,' though I did have to remove and reinsert the battery pack to restart the display."

Yours faithfully,

Michael Prior (Riyadh, Saudi Arabia)

(When the HP-41 occasionally loses memory because of various "traumas," we Americans call that a "crash." Seems to me you did it the "hard" way, Mr. Prior, but we're happy to hear your HP-41 recovered all right. Thanks for the letter—Ed.)

Now, here's what happens to your HP-41 when you leave it where "man's best (?) friend" can reach it. For some reason, dogs just *love* to chew on calculators, and HP's are no exception. This one showed up in our Service Department, so we photographed it to show you what can happen to your pride and joy when Fido "bones-up" on his/her math.



We Get Letters . . .

With probably over a quarter of a million KEY NOTES readers sprinkled all over the world, you can imagine the variety of letters we receive. Some, we enjoy sharing with you. Like these:

"I recently acquired an HP-34C. I was quite pleased with the calculator, but even more so after a recent experience. Last week, due to an unfortunate chain of events while on my way to school, I ran over my backpack, which had my calculator in it. I found the case of the calculator to have tire marks on it, but the calculator itself was in perfect working condition. The body of the calculator was slightly bent, but it survived [the incident] better than I imagined. I thank you for a product which is as durable as it is useful."

Yours truly,

Karen Lund (Provo, Utah)

"Yesterday I found out that an HP-41C (with card reader and three Memory Modules) can fall from a motor car at 40 kilometers per hour and survive. I had left it on the roof of the car while loading the trunk. After driving about 200 meters and hearing a strange sound, I looked in the rear vision mirror to see my calculator sliding and somersaulting down the road. Gingerly I unzipped the soft pouch, expecting to find only dust and crunched calculator parts. To my great relief, the

"Dear Mr. Horn:

The following may be of interest to other readers of KEY NOTES.

"When the Wand is plugged into the HP-41 and the calculator is off, depressing the scan switch on the Wand will turn on the HP-41. Using this feature, one could turn on, load program, run the program, and turn off the HP-41C without touching the keyboard."

Regards,

Edward R. Gabel (Minneapolis, Minnesota)

(This doesn't appear in the book, Mr. Gabel, but you are correct in what you state. However, there is *one small flaw: if the photodetector built into the tip of the Wand does not "see" a reflection, it will* not *turn on the HP-41. So all you have to do is point the Wand at the paper, and it will do exactly what you wrote—Ed.)*

Although it created quite a few "waves" in the scientific community when Hewlett-Packard announced the HP-35, you could hardly call the event "earth-shaking." And, speaking of earth-shaking, when's the last (first?) time you used *your* calculator to measure an earthquake? (With Mt. St. Helens erupting only 130 miles north of Corvallis, maybe *we* should use this system?) Anyway, here's an interesting letter from someone with great imagination along these lines. (The letter was published in the European KEY NOTES about a year ago.)

Gentlemen:

I have succeeded in demonstrating one more use for my HP calculator, in an advanced "Appropriate Technology" application: as a recording seismograph.

Following the Livermore, California, earthquake of 24 January 1980, I suspended the AC adapter/recharger by its cord from the ceiling of my office. Just below it, I placed a sheet of paper, marked off in concentric circles of increasing magnitude. With a rubber band (recycled!), a soft pencil was attached as a pointer, then the contraption was lowered until it just touched the paper.

By this means, I successfully recorded the aftershocks that occurred that night during my absence. Of course, my intensity scale of quake magnitude had to be entirely relative; and the device will record only the S-waves. But I have demonstrated to my chagrined cohorts, who use the "other" brand of calculator, the superiority and stunning versatility of my HP—to my undisguised glee.

Sincerely,

Ed Chatfield, Sacramento, California

(Nice going, Mr. Chatfield! I couldn't find any "fault" with that *application. Or . . . could I?—Ed.)*

*(**Note:** The HP-41 subroutines on page 14 appeared a year ago in the European KEY NOTES, but are being published again for the rest of the world.—Ed.)*

HP-41 Subroutines

Most of the time, we print in KEY NOTES the ways that *you* develop, on your own, to solve little programming problems. This time, however, we'll show you some subroutines that are being used at the factory to develop future software. If you learn and apply these subroutines, you will find more consistency between user programs and HP programs, plus you will find that programs will work properly with or without the printer.

Input/Output Subroutines

"IN" is used to prompt for, store, format, and print input values. "IN" consists of the following steps:

```
01♦LBL "IN"
02 CF 22
03 1
04 ST+ 00
05 RCL IND 00
06 "⊢="
07 ASTO Y
08 "⊢?"
09 CF 21
10 AVIEW
11 SF 21
12 CLA
13 ARCL Y
14 STOP
15 STO IND 00
16 FS? 22
17 FC? 55
18 RTN
19 ARCL X
20 PRA
21 RTN
```

"IN" uses register 00 as a pointer to decide where a value should be stored. Thus, before calling IN, you must be sure that register 00 contains the correct pointer. If you want an input stored in register 20, register 00 must contain 19 before IN is called. IN will automatically increment register 00 so that the next call to IN will cause a value to be stored in register 21.

IN prompts the user for data input. IN expects a five-character (or less) input variable name in the ALPHA-register when it is called. For instance, if you wished to prompt for input of the variable "ABC" and store the input in register 12, the calling sequence would be:

```
01 11
02 STO 00
03 "ABC"
04 XEQ "IN"
```

IN is convenient for the user of your programs. Once a problem has been run, the user can rework the problem, keying in only the values he or she wishes to change. (Pressing R/S without keying in a value leaves the value unchanged.) This allows rapid sensitivity analysis of chosen variables.

Flag 22 is set upon return from IN if the user made an input; it is clear if the user did not make an input. You may be able to make use of this fact.

"OUT" formats and either prints or displays the value in the X-register. "OUT" consists of the steps below.

```
01♦LBL "OUT"
02 SF 21
03 "⊢="
04 ARCL X
05 AVIEW
06 RTN
```

OUT is simpler than IN. Put the value to be output in the X-register and the name of the value in the ALPHA-register. For example, the program sequence to output a value called "DEF" currently in register 11 would be:

```
01 RCL 11
02 "DEF"
03 XEQ "OUT"
```

YES or NO Question Subroutine

It is frequently desirable to ask the user a question with two possible answers. It is almost always possible to pose these questions in a *yes* or *no* context. It is usually desirable to remember the user's answer in the form of a set (yes) or clear (no) flag. The subroutine "Y/N?" aids in asking such questions:

1. It adds the characters "? Y/N" to the prompt put in the ALPHA-register prior to call. The prompt must contain 6 or less characters.
2. The subroutine prints the results of the question if a printer is present.
3. The subroutine sets or clears the flag specified by the contents of the X-register, on call. (If X=5, flag 5 is set or cleared.)
4. Similar to IN, the subroutine retains the current status of the flag if the user fails to answer the question.
5. The subroutine sets ALPHA mode and turns it off.

```
01♦LBL "Y/N?"
02 CF 23
03 ASTO L
04 "⊢? Y/N"
05 AON
06 PROMPT
07 AOFF
08 FC? 23
09 RTN
10 CF IND X
11 RDN
12 ASTO X
13 "Y"
14 ASTO Y
15 X=Y?
16 SF IND T
17 FC? 55
18 RTN
19 CLA
20 ARCL L
21 "⊢: "
22 FS? IND T
23 "⊢YES"
24 FC? IND T
25 "⊢NO"
26 AVIEW
27 RTN
```

Program Title Subroutine

A nice touch in many applications is a title on the printed output. This subroutine prints the title, double wide, and spaces appropriately.

```
28♦LBL "TITLE"
29 ADV
30 SF 12
31 FS? 55
32 PRA
33 CF 12
34 ADV
35 RTN
```

SIZE Check Subroutine

It can be very annoying to be on the last input of a long input sequence and get a "NONEXISTENT" error. This is usually the result of an incorrect SIZE. By executing this subroutine at the beginning of a program, this problem is eliminated.

```
01♦LBL "SIZE"
02 "SIZE>="
03 ARCL X
04 1
05 -
06 SF 25
07 RCL IND X
08 RTN
```

To call this subroutine, you must place the necessary SIZE in X prior to the call. The calling sequence must never be in another subroutine! The calling sequence for a SIZE of 54 is:

```
01 54
02 XEQ "SIZE?"
03 FC?C 25
04 PROMPT
```

SIZE and TITLE Combined

Since SIZE and TITLE are usually done first, one call can usually do both jobs. The combined subroutine is:

```
01♦LBL "TITLE"
02 ADV
03 SF 12
04 FC? 55
05 PRA
06 CF 12
07 ADV
08♦LBL "SIZE?"
09 "SIZE>="
10 ARCL X
11 1
12 -
13 SF 25
14 RCL IND X
15 RTN
```

The calling sequence, for a TITLE of F=MA and a SIZE of 6, is:

```
01 6
02 "F=MA"
03 XEQ "TITLE"
04 FC?C 25
05 PROMPT
```

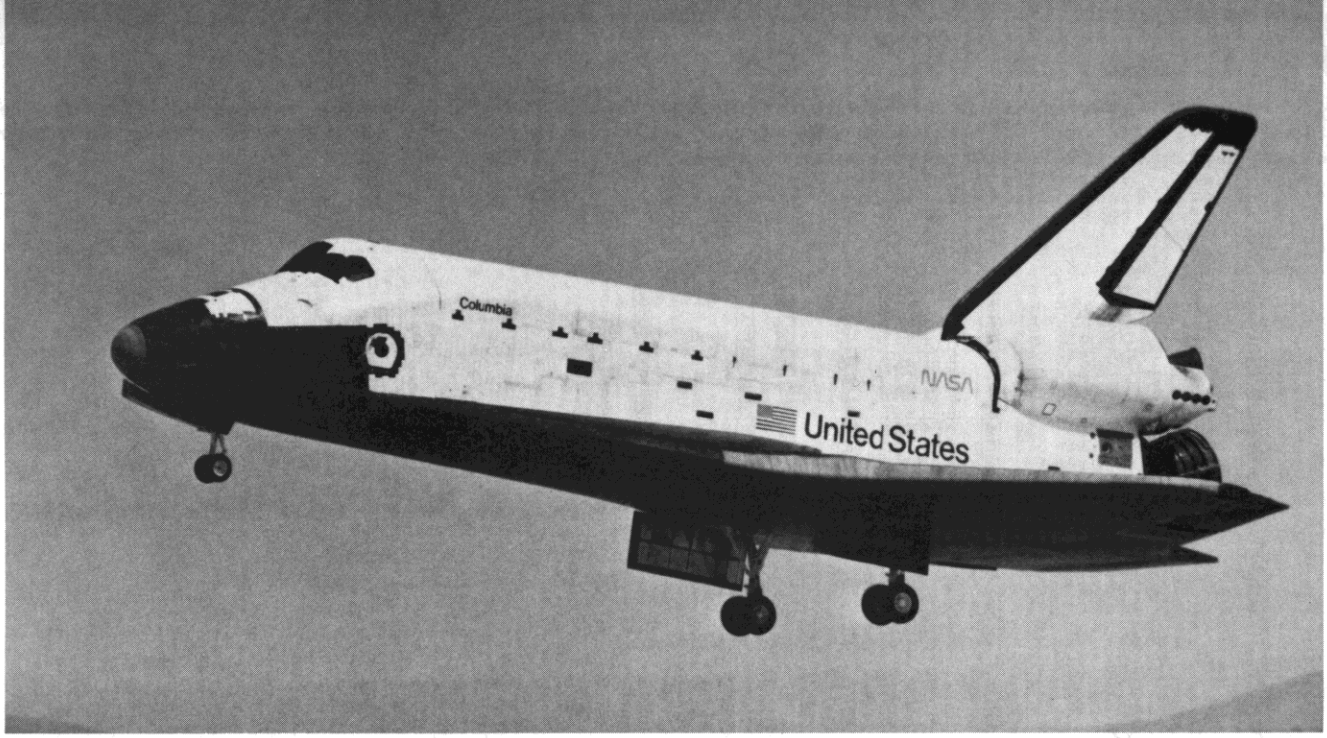
HP-41 Function List

We have had many, many requests for a list of all of the functions available on the HP-41, so here it is. Sometimes if you do not see an unusual display for a while or do not key in the steps for a while, you tend to forget how to do it, and that can be very frustrating. Now you have a complete record to fall back on when *your* memory fails you.

FUNCTION NAME	DISPLAY	KEYSTROKES
*PLUS	+	+
*MINUS	-	-
*TIMES	*	×
*DIVIDE	/	÷
*RECIPROCAL	1/x	1/x
*10^x	10↑x	■ 10^x
*ABSOLUTE VALUE	ABS	XEQ ALPHA ABS ALPHA
*ARC COSINE (COS^{-1})	ACOS	■ COS^-1
PAPER ADVANCE	ADV	XEQ ALPHA ADV ALPHA
ALPHA OFF	AOFF	XEQ ALPHA AOFF ALPHA
ALPHA ON	AON	XEQ ALPHA AON ALPHA
*ALPHA RECALL	ARCL	ALPHA ■ RCL
ALPHA SHIFT	ASHF	XEQ ALPHA ASHF ALPHA
*ARC SINE (SIN^{-1})	ASIN	■ SIN^-1
*ASSIGN	ASN	■ ASN
*ALPHA STORE	ASTO	ALPHA ■ STO
*ARC TANGENT (TAN^{-1})	ATAN	■ TAN^-1
*ALPHA VIEW	AVIEW	ALPHA ■ VIEW
*BEEP	BEEP	■ BEEP
*BACK STEP	BST	■ BST
*CATALOG	CAT	■ CATALOG
*CLEAR FLAG	CF	■ CF
*CHANGE SIGN	CHS	CHS
*CLEAR ALPHA	CLA	ALPHA ■ CLX/A
CLEAR DISPLAY	CLD	XEQ ALPHA CLD ALPHA
CLEAR PROGRAM	CLP	XEQ ALPHA CLP ALPHA
CLEAR REGISTERS	CLRG	XEQ ALPHA CLRG ALPHA
*CLEAR STATISTICAL REGISTERS	CLΣ	■ CLΣ
CLEAR STACK	CLST	XEQ ALPHA CLST ALPHA
*CLEAR X	CLX	■ CLX/A
COPY	COPY	XEQ ALPHA COPY ALPHA
*COSINE	COS	COS
DEGREES INTO RADIANS	D-R	XEQ ALPHA D ■ - R ALPHA
OCTAL TO DECIMAL	DEC	XEQ ALPHA DEC ALPHA
DEGREES MODE	DEG	XEQ ALPHA DEG ALPHA
DELETE	DEL	XEQ ALPHA DEL ALPHA
DECREMENT, SKIP IF EQUAL	DSE	XEQ ALPHA DSE ALPHA
END	END	XEQ ALPHA END ALPHA
*ENGINEERING NOTATION	ENG	■ ENG
*ENTER	ENTER↑	ENTER↑
*e^x (EXPONENTIAL FUNCTION)	E↑X	■ e^x
$e^{(x-1)}$	E↑X-1	ALPHA E ■ ENTER X ■ - ■ 1 ALPHA
FACTORIAL	FACT	XEQ ALPHA FACT ALPHA
IS FLAG CLEAR?	FC?	XEQ ALPHA FC? ALPHA
IS FLAG CLEAR? CLEAR	FC?C	XEQ ALPHA FC?C ALPHA
*FIX DISPLAY	FIX	■ FIX
FRACTIONAL PORTION	FRC	XEQ ALPHA FRC ALPHA
*IS FLAG SET?	FS?	■ FS?
IS FLAG SET? CLEAR	FS?C	XEQ ALPHA FS?C ALPHA
GRADIANS MODE	GRAD	XEQ ALPHA GRAD ALPHA
*GO TO	GTO	■ GTO
CONVERT TO HOURS, MINUTES, SECONDS	HMS	XEQ ALPHA HMS ALPHA
HOURS, MINUTES, SECONDS PLUS	HMS+	XEQ ALPHA HMS ■ + ALPHA
HOURS, MINUTES, SECONDS MINUS	HMS-	XEQ ALPHA HMS ■ - ALPHA
CONVERT TO DECIMAL HOURS	HR	XEQ ALPHA HR ALPHA
INTEGER PORTION	INT	XEQ ALPHA INT ALPHA
*INCREMENT, SKIP IF GREATER	ISG	■ ISG
*LAST X	LASTX	■ LASTx

ERRATA: The sixth entry under "Display" should be 10↑X. Also, VIEW and AVIEW can be accessed by ALPHA entries or by keyboard keys.

FUNCTION NAME	DISPLAY	KEYSTROKES
*LABEL	LBL	■ LBL
*NATURAL LOG (BASE e)	LN	LN
NATURAL LOG OF (1+x)	LN1+X	XEQ ALPHA LN ■ 1 ■ + X ALPHA
*COMMON LOG (BASE 10)	LOG	LOG
MEAN	MEAN	XEQ ALPHA MEAN ALPHA
MODULO	MOD	XEQ ALPHA MOD ALPHA
DECIMAL TO OCTAL	OCT	XEQ ALPHA OCT ALPHA
*OFF	OFF	XEQ ALPHA OFF ALPHA
*ON	ON	XEQ ALPHA ON ALPHA
*POLAR TO RECTANGULAR	P-R	■ P→R
PACK	PACK	XEQ ALPHA PACK ALPHA
*PERCENT	%	■ %
PERCENT CHANGE	%CH	XEQ ALPHA ■ G CH ALPHA
*PI	PI	■ π
PROMPT	PROMPT	XEQ ALPHA PROMPT ALPHA
PAUSE	PSE	XEQ ALPHA PSE ALPHA
ROLL UP STACK	R↑	XEQ ALPHA R ■ ENTER ALPHA
RADIANS TO DEGREES	R-D	XEQ ALPHA R ■ - D ALPHA
*RECTANGULAR TO POLAR	R-P	■ R→P
RADIANS MODE	RAD	XEQ ALPHA RAD ALPHA
*RECALL	RCL	RCL
*ROLL DOWN STACK	RDN	R↓
ROUND OFF	RND	XEQ ALPHA RND ALPHA
*RETURN	RTN	■ RTN
STANDARD DEVIATION	SDEV	XEQ ALPHA SDEV ALPHA
*SCIENTIFIC NOTATION	SCI	■ SCI
*SET FLAG	SF	■ SF
*SIGMA PLUS (STAT. REG.)	Σ+	Σ+
*SIGMA MINUS	Σ-	■ Σ-
SPECIFY SIGMA REGISTERS	ΣREG	XEQ ALPHA ■ F REG ALPHA
*SINE	SIN	SIN
SIGN	SIGN	XEQ ALPHA SIGN ALPHA
SIZE	SIZE	XEQ ALPHA SIZE ALPHA
*SQUARE ROOT	SQRT	√x
SINGLE STEP	SST	SST
*STORE PLUS	ST+	STO +
*STORE MINUS	ST-	STO -
*STORE TIMES	ST*	STO ×
*STORE DIVIDE	ST/	STO ÷
*STORE	STO	STO
*RUN/STOP	STOP	R/S
*TANGENT	TAN	TAN
TONE	TONE	XEQ ALPHA TONE ALPHA
VIEW	VIEW	XEQ ALPHA VIEW ALPHA
*IS X EQUAL TO ZERO?	X=0?	■ x=0?
IS X NOT EQUAL TO ZERO?	X≠0?	XEQ ALPHA X ■ H ■ 0? ALPHA
IS X LESS THAN ZERO?	X<0?	XEQ ALPHA X ■ I ■ 0? ALPHA
IS X LESS THAN OR EQUAL TO ZERO?	X<=0?	XEQ ALPHA X ■ I = ■ 0? ALPHA
IS X GREATER THAN ZERO?	X>0?	XEQ ALPHA X ■ J ■ 0? ALPHA
*IS X EQUAL TO Y?	X=Y?	■ x=y?
IS X NOT EQUAL TO Y?	X≠Y?	XEQ ALPHA X ■ H Y ? ALPHA
IS X LESS THAN Y?	X<Y?	XEQ ALPHA X ■ I Y ? ALPHA
*IS X LESS THAN OR EQUAL TO Y?	X<=Y?	■ x≤y?
*IS X GREATER THAN Y?	X>Y?	■ x>y?
X EXCHANGE WITH (ANY REGISTER)	X<>	XEQ ALPHA X ■ I ■ J ALPHA
*X EXCHANGE WITH Y	X<>Y	x≤y
*EXECUTE	XEQ	XEQ
*X SQUARED	X↑2	■ x^2
*Y TO THE X POWER	Y↑X	■ y^x



Space shuttle *Columbia* landing after first orbital flight. (Photo courtesy of NASA.)

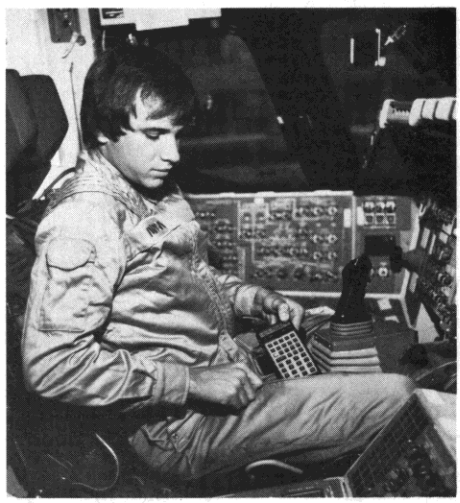
used before reentry into the earth's atmosphere to compute the shuttle's present center of gravity and the amount of fuel to be burned in each tank to reach the required center of gravity for reentry. This center of gravity program was termed "flight critical" by NASA and necessitated extensive pre-launch testing of the calculators.

The other program, the Acquisition of Signal program, ran continually in the second calculator, starting at launch, so it could display at any time the next ground station that *Columbia* could contact, when it would be in contact, the duration of that contact, and which frequency (UHF or S-band) could be used. And, thanks to Continuous Memory, the calculator did not have to be on during the whole flight.

You will be interested to know that NASA is committed to using HP-41C's in future shuttle missions, and that it plans more exotic applications. One likely program will let the HP-41C compute the "navigational" commands to be given to a mechanical arm so it can reach out and grab a nearby satellite. Another program would take, as input data, coordinates of the shuttle's big hatch and determine if it is closed.

Hewlett-Packard is very pleased that the HP-41C was chosen for this application, and we are working to support NASA's future needs. HP may produce custom ROM modules containing the special shuttle programs, and thus eliminate the need for Memory Modules.

NASA also foresees the day when astronauts will carry into orbit HP printer/-plotters that work as peripherals with the HP-41C, making hard copy immediately available.



A NASA technician in the Space Shuttle Simulator stores the HP-41C Calculator in a special pouch in the astronauts' flight suit.

HP KEY NOTES

January-April 1981 Vol. 5 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.*

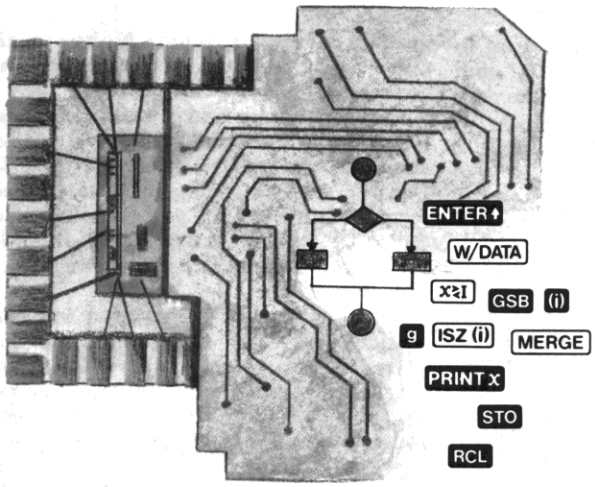
Hewlett-Packard Company
Users' Library
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 USA

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

Address Correction Requested

BULK RATE
U.S. POSTAGE
PAID
PERMIT NO. 814
PORTLAND, OR

Printed in U.S.A.



Featuring, this issue:

May-August 1981 Vol. 5 No. 2

HEWLETT PACKARD

HP Key Notes

In Five Years . . .

It is almost exactly 5 years since this division of Hewlett-Packard moved to Corvallis. On the cover of HP KEY NOTES Volume 1 Number 1, we printed a photograph of our first building. At that time, there also was another building being started.

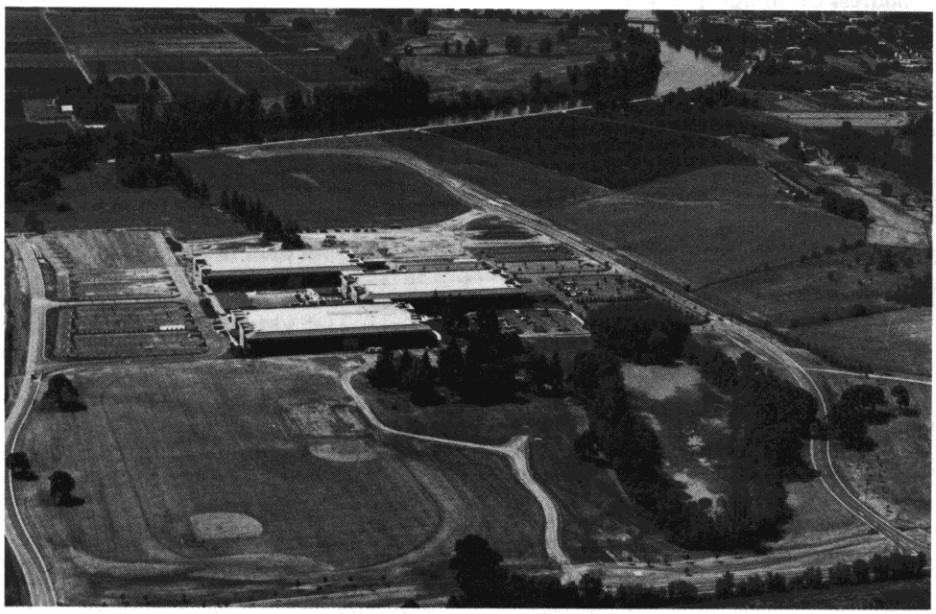
On the cover of Volume 3 Number 2 (May 1979), we printed an aerial photograph of the 140-acre Corvallis Division site. At that time, there were two completed buildings with a total of 416,000 square feet of floor space.

Since then, the HP-41 calculator system was introduced, and the highly successful new Series 80 Personal Computers were developed and introduced. So we started a third building, and it is due for completion approximately by year's end. Also, because most present conference rooms have been absorbed by the search for floor space, we have started a one-story cafeteria-conference building. And the fourth building is in the planning stage.

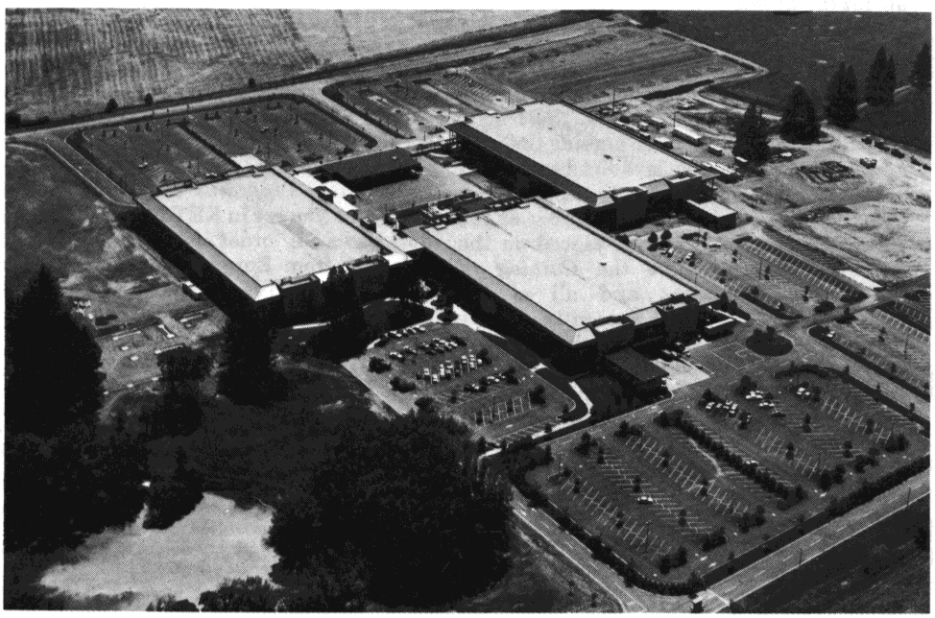
It is amazing how much floor space is required to make small, complex instruments like calculators and personal computers. But our customers thoroughly like our products and purchase them in ever-increasing numbers. Thus, to satisfy our clientele, we must expand.

In the close-up photograph, between the lake and the building on the left, you can see the preliminary foundation work for the new cafeteria-conference building.

The Corvallis site is just north of downtown Corvallis, which can be seen at the top right corner of the upper photo. The river in the background is the Willamette, and it borders the east side of the site. We are about 81 miles (130 km) south of Portland and 40 miles (64 km) north of Eugene, and it is about 53 miles (85 km) to the Pacific Ocean. Corvallis is at latitude 44°34′ north and longitude 123°17′ west.



Hewlett-Packard's Corvallis site. That's a 5-acre natural lake at lower right, and at left rear is the newest building.



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

Library Corner

All of the programs highlighted in KEY NOTES are available worldwide. However, before you order any, be sure to read the paragraph below: "**Ordering Programs.**"

SUBMITTING PROGRAMS

To maintain the high quality of the programs submitted to and accepted into the Users' Library, we encourage you to closely follow the *Users' Library Contributors Guide for the HP-41, HP-67 and HP-97.* Complete and orderly documentation is essential to ensure the acceptance of a contributed program to the Library.

In the last issue of KEY NOTES (Volume 5 Number 1) we promised to publish some new guidelines for submitting programs. Well, we have done better than that; we have created a new column called "In the Key of HP." This ongoing column will address the things that we look for when we are determining the acceptability of a program that has been submitted to the Users' Library. Also in this column, we will present some extremely useful ideas and techniques that we have developed here at HP, and we will answer some of the more common questions that we encounter nearly every day.

Up to now, every program submitted to the Library had to include a magnetic card (or cards). We have a good reason for this; without a card or cards, it would take far too long to review and check all the many program submittals. Also, there is always an increased chance for errors when someone keys in handwritten keystrokes.

Since the advent of the HP 82153A Digital Wand for the HP-41, we can now accept HP-41 program submittals that have bar code instead of magnetic cards. However, **the bar code you submit with a program must be reproducible.**

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.

LIBRARY SUBSCRIPTIONS

In the United States and Canada, the fee for a one-year subscription to the Users' Library is $20.* If you live outside the U.S. and Canada, the fee is $30* because of considerably higher postage and handling charges. Upon becoming a member of the Users' Library, you will be placed on the mailing list to receive the *Catalog of Contributed Programs* and all of the updates. Plus, you will presently receive four coupons for free programs, each program valued at $6.* It doesn't take an HP-41 to realize that this is a good deal! And, as an added incentive, a free one-year subscription to HP KEY NOTES is included in your initial membership to the Users' Library, **no matter where you live in the world.**

Also, it will please you to know that once you are a member of the Users' Library, your succeeding renewal subscriptions in the U.S. and Canda are only $10,* and outside the U.S. and Canada they are only $15.*

IF YOU WERE A MEMBER ...

You would be receiving a tremendous bargain from now until December 31, 1981. During that time, the Corvallis Users' Library is offering—**FOR MEMBERS ONLY**—a 25-percent discount on all orders for six or more programs. You would also get 48-hour turnaround time on orders received by mail or by our toll-free telephone number. And you would get same-day service on orders telephoned *directly* to the Library. Plus, with the next *Catalog Addendum*, you would get—FREE—complete documentation with bar code for the HP-41 game program, "Reversi."

And there are many more benefits of Library membership, whether in Corvallis or Geneva. You should reconsider this opportunity if you have turned it down in the past. The Corvallis Library, for example, now has the equipment, facilities and a larger staff to handle all your varying needs. Plus, in the near future, the Library will offer even *more* services.

Think about it again. For the price of membership, you presently get four free programs of your choice. That, alone, is worth more than the membership fee. And in the future, we will be making more special offers *for members only*. But the greatest advantage of Library membership is being able to choose from a very large collection of software that is all ready to serve you in *your* application. The savings in time and effort are worth much more than the small membership charge.

ORDERING PROGRAMS

HP-67/97 and HP-41 programs mentioned 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 a prerecorded magnetic card (or cards). Also, for HP-41 programs, this 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 or if you are ordering from Europe, South America, or Asia, 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).

Here's a helpful hint for customers outside the U.S. We have found that your orders are handled in a more efficient and timely manner if you will send, **attached to your order**, an International Money Order, a Foreign Draft, or the equivalent. *Any of these must be in U.S. dollars, drawn on a U.S. bank*, otherwise they will be returned to you, which involves a long delay for you. Much time is wasted and orders are held up in trying to match orders and checks that are sent in separately, or written on checks for non-U.S. banks and in foreign currency. Another option for you is to use such major credit cards as American Express, VISA, or MasterCard.

Usually, orders *not* delayed by the above problems will be shipped within 48 hours after they are received in Corvallis.

** 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.*

NEW PROGRAMS

Here are some recent submittals to the Corvallis Users' Library. All of the programs 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 a Quad-RAM or the HP-41CV.

(67/97) Lunar Day Converter (#04536D)

Using the Lunar day and the hour of the moon this program calculates lunar brightness and stellar magnitude for use with the program **Astrophotography Exposure Guide** (following). It will also convert lunar brightness to stellar magnitude. When the field of view of the telecamera system is less than the diameter of the moon, this program will convert brightness and magnitude to reflect the area covered by the field of view. (211 lines, 8 pages)

(67/97) Astrophotography Exposure Guide (#04551D).

Using this program, and knowing the stellar magnitude, telecamera *f* number, film ASA, elevation angle, and filter factor, one can determine approximate exposure times, along with times corrected for reciprocity. Individual changes can be made to any of the inputted datums for easy comparisons and changes in exposure times. By bracketing exposure times to compensate for atmospheric conditions, the astronomer can quickly determine exposure times that will ensure a good picture. This

program corrects for black and white film reciprocity from 0.5 to 400 seconds, given the corrected exposure times up to 1-½ hours. (224 lines, 13 pages)

(41) Lunar Day Converter/Astrophotography Exposure Guide (#00771C)

This program combines the two preceding (67/97) programs. It automatically prompts for data and it works with the printer. *Required accessories: Three Memory Modules*. (250 lines, 839 bytes, 18 pages)

The author of the preceding three programs is **James P. Patterson** of Cambridge Bay, N.W.T. Canada

(67/97) Shape Factors Calculations (I) (#04553D)

This program finds, by direct evaluation of the formula, the shape factor for the following systems: (a) Two plane rectangular surfaces perpendicular to each other and having one common edge; (b) Two equal parallel rectangular surfaces in opposite location; (c) Two plane circular surfaces with common central normal. Inaccurate reading of graphs is avoided by using this program. (117 lines, 8 pages) (*This program will be of interest to solar engineers—Ed.*)

Author: **Jorge A. Pita**
Quito, Ecuador

(67/97) Combined Footing Analysis (#04514D)

This program analyzes a combined footing, such as that used in an A-frame foundation, giving the factor of safety against overturning, soil pressure distribution under the footing, and maximum moment in the footing slab between the piers. (695 lines, 26 pages)

Author: **Joseph S. Lucca**
Elmhurst, Illinois

(41) Plot of Two or Three Functions on One Graph (#00732C)

This program extends the capability of plotting on the HP 82143A printer, thus enabling up to three functions of x to be plotted on one graph. The program works in a way similar to the PRPLOT function of the printer prompting for function name and limits of x and y axes. *Required accessories: HP 82143A Printer, one Memory Module*. (183 lines, 350 bytes, 6 pages)

Author: **John L. Gilby**
Jagersrust, South Africa

(41) Means (Statistics) (#00844C)

This program solves for the arithmetic mean, geometric mean, harmonic mean, root mean square, standard deviation, mean deviation, coefficient of variation, and Z-statistic. All outputs are clearly labeled by use of the HP-41 alphanumeric capabilities. *Required accessories: One Memory Module*. (209 lines, 448 bytes, 9 pages)

Author: **Richard S. Altman**
Richmond, California

(41) Sieve Analysis and Soil Moisture Content (#00674C)

This program can be used to determine the cumulative percent retained, percent finer, and sample loss, given the total sample weight and the weight retained on each screen. It also can be used to determine the weight of water, the weight of dry soil, and the percent moisture, given the wet and dry sample weights and the dish weight. *Required accessories: None*. (120 lines, 283 bytes, 7 pages)

Author: **Mike E. Brazie**
Golden, California

(41) Interpolation and Numerical Integration by Akima's Local Process (#00679C)

This program applies Akima's process to finding a set of local interpolating polynomials that best fit the curve that would be drawn through a set of arbitrarily spaced points by eye, and it also integrates the result. Akima's process introduces no false extremes or inflections. *Required accessories: None*. (161 lines, 273 bytes, 7 pages)

Author: **J. F. G. Darby**
Parkville, Australia

(41) Data Input (Store) and Review (#00693C)

Numeric and/or alphanumeric data may be stored in a user-specified series of data registers. Error correction is available for the most recent datum input. Register numbers are provided in a PROMPT message. Any or all data registers may be reviewed without a printer. The review will display the register number and its contents. The user specifies the data display format. Also included is a routine that performs the same function but saves 9 registers by eliminating some of the prompts. *Required accessories: None*. (70 lines, 154 bytes, 6 pages)

Author: **Barbara G. Lawrence**
Balboa, Republic of Panama

(41) Time Sharing (#00690C)

Up to four people can use the calculator without interfering with each other's calculations. Businessmen in conference or students doing homework together will find this program useful. The stack and storage registers 1, 2, and 3 are available for each user. All that is needed when changing hands is for each new user to press their user key—A, B, C, or D. Otherwise, the calculator functions as normal. *Required accessories: one Memory Module (only required if statistics registers are used)*. (130 lines, 204 bytes, 9 pages)

Author: **Neil M. Hunter-Blair**
Bangkok, Thailand

* *U.S. dollars. See note at bottom edge of front cover. Also, overseas orders for this program must include a 10-percent postage charge. See "Ordering Programs" on page 2.*

THE CAVES

Do you sometimes feel as though the walls are closing in around you? Do you ever crave a life full of intrigue and adventure? Well, this HP-41 program will not solve your problem with the walls, but it will take you on an imaginary adventure full of excitement and intrigue.

"The Caves" is a recently submitted program that is the dedicated work of **James R. Surber** of Omaha, Nebraska. And, at $6* it is a real bargain from your Users' Library—especially if you enjoy games programming. For that paltry sum you get 30 pages of excellent documentation plus some pages of bar code. The program consists of 8 magnetic cards, 578 lines, and 1704 bytes.

Included in this bargain are illustrations and an example called "The Story of the Caves." The story begins: "Once upon a time there was a wizard who set out to make his name in the wonderful world of wizardry and thus earn a fabulous fortune. The titillating trail that awaited him was the awesome task of travelling through the world of "The Caves." And if that, alone, doesn't capture your interest, read through the following abstract.

(41) The Caves (#00900C)

"The Caves" is an adventure game in which the player moves about in a series of 67 interconnected caves to pick up nine items to total 1200 points without being "killed" by the various hazards. If you *like* games, mazes, and puzzles with magic, you will *love* "The Caves."

This is a full-memory-capacity program but, because of the size, it may be totally entered by the card reader or the Wand, even if your system has individual memory modules. (The fourth module must be in place to start the game.) The printer is optional and uses flag 13 in the program. This causes the warnings and wizard's words to be in lowercase print. "Auto off" is used if death occurs.

The game is simple to operate in that the display or the overlay gives you any necessary information. The board is easily imaged in the mind, or a list of 1 to 67 caves on paper is all that is needed to help remember hazardous caves.

The game has chance, skill, and goal orientation in it to keep the player interested and active. Hazards and valuables are interconnected to the equipment present, and it takes skill to move through the caves. Loss of equipment, non-warned hazards, and random selection of caves without warning play the role of chance in "The Caves." Goal orientation is just getting through alive and scoring.

Some ideas and parts of the program were taken from the *Games Solution Book* by Hewlett-Packard. I would also like to thank **Scott Sowders** for his help in the game "The Caves." *Required accessories: Four Memory Modules (or Quad RAM, or HP-41CV).*

Do You *Really* Know Your HP-67/97?

When KEY NOTES contained only 12 pages, it was difficult to find the necessary space to print all the longer articles we received. Now, however, it is easier to plan for such articles, and we are happy to present this one for all our HP-67/97 owners. And, although written some time ago, it is still good information.

This test is the contribution of **William M. Kolb** of Marlboro, Maryland. It was printed before in the *PPC Calculator Journal*, the newletter for the PCC Club.* Longtime readers of KEY NOTES will remember Mr. Kolb from his contributions in V2N4p10 and V1N3p4 (both out of print and no longer available).

Mr. Kolb developed this self-assessment test to help the members of his local Chapter of PPC determine if they were taking full advantage of their PPC's (personal programmable calculators). There is no scoring criteria. The purpose of the test is to teach you something about your own abilities and not those of others. The suggested answers involving programs, however, are worth studying for additional programming insight. These particular solutions were not selected for their brevity, but because they illustrate straightforward approaches using certain features of the calculator. And, although the questions are for the HP-67, nearly all of them apply to the HP-97.

HP-67 SELF ASSESSMENT TEST

Are you getting the most out of your PPC? The following self-test will help you find those areas where additional study can improve your programming ability. Allow yourself about 2 hours to finish. Do not use your calculator to check the answers until you are finished. Answers are on page 8.

1. NOP is a "dummy" instruction used as a filler step on some calculators. The HP-67 does not have a NOP key. Give an example of an HP-67 program that requires a NOP.
2. There are two ways to double a number in the X-register: (a) [2], [×] and (b) [ENTER], [+]. Other than speed, what is the principal difference?
3. If the I-register contains 15.00, what is the result of the instruction GTO (i)?
4. What are the contents of the stack registers after the following keystrokes: CLX, ENTER, 1, CHS, ENTER, CHS, 2, CHS, 3?
5. After single-stepping through the following program in RUN mode, will flag 3 be set or clear? LBLA, CF3, 0, F3?, ÷, RTN. Is there any difference between single-stepping or running this program?
6. What is the value of LSTx after executing the following: π, D←, CHS, 1, STO+0, CLX?
7. Under what flag conditions will the following program branch to LBL 0: F0?, F3?, F3?, GTO 0, R/S?
8. What stack register prints first after the command to print stack?
9. Write a four-step program to rearrange the stack from XYZT to YZXT.
10. Assume that the stack is arranged as follows: X=1, Y=2, Z=3, T=4. What does the stack contain after the following keystrokes: +, X≷Y, STO+0, ENTER, LSTx?
11. The function X<Y does not appear on the HP-67; write a program to test for X<Y.
12. Where will program execution stop when A is executed from the keyboard?

001 LBL A	011 LBL 1	021 LBL 2	031 LBL 3	041 LBL 2
002 GSB 1	012 GSB 2	022 GSB 3	032 GSB 2	042 RCL 1
003 RTN	013 RTN	023 RTN	033 RTN	043 RTN

13. If step 042 in the previous question is changed to GSB 2, where will execution stop?
14. List ten different conditions that will cause the display to show an error message.
15. Write a program that returns the integer remainder when an integer in the Y-register is divided by an integer in the X-register.
16. Does the program from the previous question consider round-off errors; i.e., the remainder when four is divided by three is one and not 0.999999999. Will the program work for DSP 0 through DSP 9 in both SCI and FIX modes? Will the program work properly for negative values of X and Y?
17. What is the result of the following instruction sequence: 90.9090, H←, H.MS←?
18. What is in the X-register after the following keystrokes: 30, ENTER, 20, +, 10, %, LSTx, +?

    (a) 5.10 (c) 25.00
    (b) 15.00 (d) Insufficient data

19. Write a program in 15 steps or less that loads register 0 through 25 with the numbers 0, −1, −2, −3, ..., −25.
20. You must load the first 8 registers with data from a magnetic card. With the exception of the I-register, no other registers are to be altered. Which keystroke sequence is correct?

    (a) 8, X≷I, W/DATA (c) 7, X≷I, MERGE
    (b) 8, STO I, MERGE (d) 7, STO I, W/DATA

21. If side two of the magnetic card is accidentally read before side one in the previous question, what are the contents of the first 8 registers? Assume that the magnetic card contains data identical to that generated in problem 19.
22. Write a program that computes the sum of the terms $2^n/n!$ from n=1 to n=100. Note that n! is equivalent to 1×2×3×4× ... × n.
23. You must merge a program called SORT with the program currently in memory. SORT is to begin at step 200 of the current program. Which is the current keystroke sequence?

    (a) 200,STO I,MERGE (c) GTO.200,MERGE
    (b) 199,STO I,MERGE (d) GTO.199,MERGE

24. How many program steps will be loaded from SORT in the previous problem if SORT contains 115 steps?
25. Describe three ways flag 3 can be set. Is there another?

* *(Founded in June 1974, 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 Calculator Journal *is a monthly publication published by PPC to disseminate user information related to applications, programs, hardware innovations, programming techniques, problems—any information related to the selection, care, use, and application of Hewlett-Packard personal programmable calculators. PPC is not sponsored, nor in any way officially santioned, by Hewlett-Packard.—Ed.)*

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.

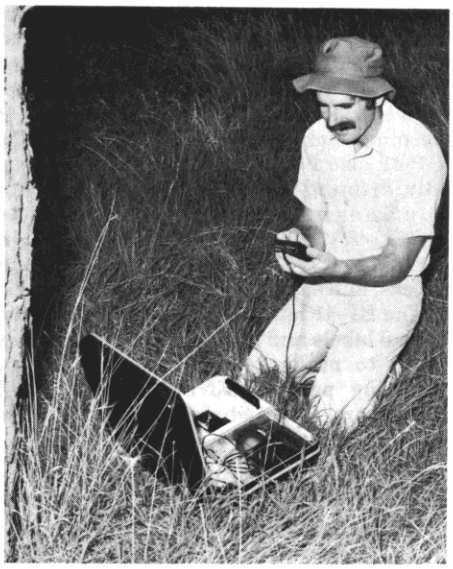
HP-41 "Bush "Computer"

No matter where you roam on planet Earth (and sometimes even in outer space!), you will find our ubiquitous HP-41 being confronted with all sorts of challenges ... challenges that it very swiftly turns into opportunities to save much time, material, and money. And in the spirit of the foregoing statement, you will find the HP-41 turning into a "bush computer" for Australia's Division of Forest Research, a part of their Commonwealth Scientific and Industrial Research Organization (CSIRO).

Just before we put the last issue "to bed," we received a letter and a photograph from **Dr. Wilfred J.B. Crane**, a Senior Research Scientist for CSIRO Forest Research. Dr. Crane also sent a press release that we think will be of interest to the many foresters who own HP programmable calculators. Here it is.

COMPUTER GOES BUSH

A computer carried on a belt could soon be saving Australian foresters, biologists, ecologists, fire fighters, and field researchers hours of calculation time, according to Dr. Wilfred Crane, CSIRO Forest Research.



Using the HP-41 "in the bush." We are not sure if that is **Dr. Crane** in the photo, but we presume it is.

Most foresters still use complex tables and slide rules to calculate such things as fire danger, tree girth, volume, growth rates, royalty and stumpage charges, fertilizer and spraying rates, thinning and cutting schedules, and pay.

"But computer technology now makes it possible to carry a hand-sized computer into the bush on one's belt, programmed and personalized to any requirement and taking only seconds to come up with answers," Dr. Crane said.

"Whether you call these things computers or calculators is a moot point, but I call them computers because they have all the facilities of a computer including a memory exceeding 2K, or two thousand bytes (bits of information); he said.

The computers are simple to operate and can be used by anyone who can read. Programming does not require special knowledge of any computer 'languages' and the instruments can be individually 'set' for a person or tasks. And the basic cost puts these 'bush computers' within reach of many foresters.

Dr. Crane believes bush computers will have many uses for foresters. "For assessing, research, log tallying, or any data collection, the machine can become a field book. Data can be entered, edited in the field, and then read out on cards if the data exceeds the capacity of the memory," he said.

Dr. Crane predicts that bush computers will be used for a wide range of purposes in the field: identifying plants, soils, and animals; measuring stream flows and estimating fire dangers; measuring trees and forests; locating roads; surveying; and even remembering radio call frequencies.

Dr. Crane uses a Hewlett-Packard HP-41CV and has written a number of programs for his work with CSIRO Forest Research. He has already had an enthusiastic response to them from colleagues working in State forest services, and hopes that more will contact him for exchange of information and ideas.

Thank you, Dr. Crane, for this contribution to KEY NOTES. We have many foresters here in the U.S. who are active programmers for their HP-65/67/97/41 calculators. Perhaps we can soon count on having your programs in the Users' Library.

Library Contest Announced

The Corvallis Users' Library is celebrating its increase in staff, facilities, and better service by announcing a **Program Submittal Contest**. Over the years, many of you have asked for or suggested such a contest, and we have finally reached the point where we can make you an offer you simply can't refuse.

Contribute your best HP-67/97/41 program(s) to the Corvallis Users' Library before December 31, 1981, and you will become eligible to win some exceptional Hewlett-Packard products. Each month, for four months from September through December, three programs will be chosen on merit by a review committee. Prizes will be awarded as follows:

September 30, 1981—3 WINNERS!
October 30, 1981—3 WINNERS!
December 31, 1981—3 WINNERS!

and each of these 12 winners will choose either an **HP-41CV Calculator** or an **HP 82143A Printer**.

On January 15, 1982, we will choose three winners for the **Grand Prizes**, and those three Grand Prizes certainly will be grand! Each of these three lucky contest winners will receive an **HP-85 Personal Computer**!

We know you will agree that this contest is worth entering. All the prizes will be awarded, and all winners will be notified by mail and announced in a future issue of KEY NOTES.

You do not need to be a Library member, and you may enter as often as you wish, but there are some rules you must follow or your submittal will not be accepted. The rules are:

- All programs must include at least the minimum documentation required by Hewlett-Packard standard Library submittal forms (and outlined in the *Contributor's Guide*). Forms from the Geneva Library will be accepted.
- All programs must be accompanied by either magnetic cards or bar code, and the bar code must be reproducible.
- All submitted programs must be in English.

Programs will be judged by the Corvallis Users' Library. They will be judged on:

1. Completeness of documentation.
2. Technical programming accuracy.
3. Usefulness (utility/technical capability).
4. Ease of use.

The decisions of the judges will be final. As programs are chosen as winners, we'll highlight them in KEY NOTES.

Here is your chance to put your calculator and your expertise to work. You have to admit that the prizes are enticing.

For submittal forms or for more information, contact the Corvallis HP-67/97/41 Users' Library at the address on the back cover. Or call them on (503) 757-2000, but remember that this is not a toll-free number.

More Software = More Solutions

There is no such thing as *too much* software. The more the better!

Recognizing this, Hewlett-Packard is implementing a program to encourage the development of software by qualified suppliers. The goal is to have an increasing flow of software from which you, the end user, can choose solutions to your problems.

Actually, similar programs have been successfully operating in Hewlett-Packard for some time, including one on the Series 80 Personal Computer line from Corvallis.

The software will be developed and must be supported by the supplier. If accepted under the program, Hewlett-Packard will help merchandise the software and make available its dealer network to do the actual selling.

One concern, of course, is quality. Even though the software products are not HP

(Continued)

supported, Hewlett-Packard wants their customers to get what they pay for. It is expected that "HP PLUS" will meet this requirement. This acronym stands for: *P*rogram for *L*ocating *U*ser-proven *S*oftware. It is a common theme for all such programs in Hewlett-Packard. What it means is that software must be sold to and used by several customers before begin accepted into the HP Software Supplier Program.

More software for more applications will be coming. But right now, HP is interested in contacting potential suppliers. Interested parties should write:

HEWLETT-PACKARD
1000 NE Circle Blvd.
Corvallis, OR 97330
Attn; Dept. 5341

Software Additions and Corrections

Here's a contribution from **John S. Chipman** of Minneapolis, Minnesota. Those of you with the *High-Level Math Users' Library Solutions* book for the HP-41 might want to add it to your book. Those of you who don't have this solutions book will find the routine useful.

The gamma-function routine listed in the *High-Level Math* book for the HP-41 is based on approximations and is valid only for x≥1. In many applications, one needs to compute γ(x) only for x equal to a positive integral multiple of 0.5 (including x=0.5). Readers of KEY NOTES might be interested in the following little routine that gives exact results for x=.5, 1, 1.5 ..., 70.5 and uses 69 bytes.

```
01♦LBL "FRGAMMA"
02 ENTER↑
03 ENTER↑
04 "ILLEGAL X"
05 X<=0?
06 GTO 01
07 INT
08 X=Y?
09 GTO 02
10 RDN
11 2
12 *
13 ENTER↑
14 INT
15 X≠Y?
16 GTO 01
17 1
18 STO 00
19 R↑
20 .5
21♦LBL 03
22 X=Y?
23 GTO 04
24 ST* 00
25 1
26 +
27 GTO 03
28♦LBL 04
29 PI
30 SQRT
31 RCL 00
32 *
33 RTN
34♦LBL 01
35 AVIEW
36 STOP
37♦LBL 02
38 1
39 -
40 FACT
41 END
```

The many applications of the HP-41's MOD function have yet to be realized. One application of MOD is helpful in converting a negative angle, or an angle greater than 360°, to its corresponding positive angle in the range from 0° to 360°. For example:

KEY STROKES	DISP
165 [CHS] [ENTER]	−165.00
360 [XEQ] [ALPHA] MOD [ALPHA]	195.00

Bob Flye of Longview, Washington, pointed out that this application of the MOD function shortens some of the programs in the *Surveying Users' Library Solutions* book. In the program "TRVAA" on page 69, lines 64 through 72 can be replaced by 360, [MOD]. In the program "AACMP" on page 72, lines 83 through 87 can be replaced with 360, [MOD]. And, in the program "AACDL" on page 73, lines 130 through 134 can be replaced with 360, [MOD].

HP-41 AVIATION PAC

An addendum card is *now* being included with the *HP-41 Aviation Pac* to print out the following three conditions. If you did not get such a card with your Pac, add this information to your book.

1. In the FLIGHT PLAN program, adequate flight parameters must be allowed for mid-flight descents. If the leg length specified is shorter than that needed to complete the descent, subsequent calculations will be incorrect. (This condition can be detected by an increasing, instead of the expected decreasing, altitude.) This warning does not apply to the final descent.
2. When using the FLIGHT PLAN program, a single-leg flight must be divided into a two-leg flight to calculate the descent heading correctly.
3. Before using the FLIGHT MANAGEMENT program, flag 21 must be cleared, unless you are using a printer.

HP-41 SECURITIES PAC

If the following addenda do not appear in your copy of the *HP-41 Securities Pac*, add the information to your book.

1. In the BONDS program, the price and yield of a 360-day bond held for exactly 6 months is incorrect. The correct price may be determined by adding the calculated price and accrued interest (X and Y registers respectively). The yield cannot be determined by keystrokes alone.
2. The calculated high and low break-even points in OPTION are incorrectly labeled; the high break-even is first, and then the low break-even.

In the Key of HP

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

In this first column, **John Laux**, a Technical Adviser in the Users' Library, discusses labels, because labels used in HP-41 programs submitted to the Users' Library are often a source of confusion. This article outlines the proper way to use labels when submitting a program to the Users' Library.

WHAT'S IN A LABEL

A label is a marker that is used to designate a useful section of a program. To access or reuse the labeled section, all you need to do is key in GTO "label" or XEQ "label" and the HP-41 will search for—and perhaps execute—that section. In many of the previous scientific programmable calculators, it was necessary to keep track of the line number associated with every entry point or subroutine in a program so that they could be used for future access. But, unfortunately, the line numbers changed whenever a line was deleted or inserted. Now, labels eliminate the inconvenience of keeping track of line numbers. Labels in the HP-41 also help the programmer by visually defining the boundaries of pertinent routines. For example, when the HP 82143A Printer prints a program in NORM or TRACE mode it prints it in blocks, and each block is a labeled routine.

The HP-41 has two major types of labels: global labels and local labels. Global labels exist to make *inter-program* jumping (jumping past an END) possible. Local labels are used when jumps are made *within* a program.

GLOBAL LABELS

A global label is defined by the function "LBL" followed by a string of up to seven ALPHA characters. The only characters that cannot be used in global labels are the decimal point (.), the comma (,), and the colon (:). The single letters A through J and a through e define local ALPHA labels (see the section: "Local Labels").

Global labels are most useful as program delimiters. A global label at line 01 is the best way to name your program; the placement of the label here is advantageous both because it is easy to find and because it is most often the best place to enter a program. Generating a meaningful global label is not difficult because it consists of up to seven ALPHA characters *of your choosing*!

Another valuable function of global labels is their use in global subroutines. Program segments that are labeled with

global labels can be called from other programs. Useful subroutines can be globally labeled to provide universal access (access from any point in program memory).

The universal accessibility of global labels means that every time a global label is called (by GTO or XEQ), the HP-41 searches the entire program memory starting at the .END. until it finds that label. (Refer to: "Additional Notes on Labels.") This can be a relatively long process. For this reason, it is most efficient to use local labels on segments of a program that are called only by their resident programs.

Another reason it is wise to be conservative when using global labels is their costliness in terms of space. The global label declaration, alone, takes four bytes of program memory, and each character in the label takes one byte. Therefore, the minimum length of a global label is five bytes and the maximum is eleven bytes. Eleven bytes is more than one-and-one-half registers! Contrast this with the fact that the minimum length of a local label is one byte and the maximum length is only two bytes. It is clear that global labels should only be used to give a routine or program universal accessibility.

An attractive feature of global labels is assignability. Once a global label has been used in program memory, that label can be assigned to a key in the same manner in which any function in the HP-41 can be assigned. The reassigned key becomes XEQ "MATRIX"—or whatever global label you have chosen. Assignability is one of the reasons that a global label is so long; the location of the key assignment is stored with the label.

LOCAL LABELS

Local labels are shorter than global labels mainly because they are defined by either a 2-digit number of by one ALPHA character. There are a total of 115 different local labels. Local labels are grouped into three categories, each category having a different use.

The first category is the short-form local label. The short-form local label gets its name from the fact that it is the shortest definable label; that is, one byte. Numeric labels 00 through 14 are short-form local labels.

The second category of local label is the long-form local label. This label gets its name from the fact that it uses 2 bytes of program memory. Numeric labels 15 through 99 are long-form local labels.

Long-form and short-form local labels each have corresponding GTO statements. The short-form GTO statement is 2 bytes long and the long-form GTO statement is 3 bytes long. On the first execution of a local short- or long-form GTO statment, the HP-41 searches the present program for the label and, upon locating the label, stores the jump distance to that label in the GTO statement. During succeeding executions of the GTO, the HP-41 does not have to search for the label because it knows exactly how far to "jump." This saves execution time! The short-form GTO statement can store jump distances of up to 112 bytes in either direction. The long-form GTO statement can store jump distances greater than the full capacity memory length of the HP-41, therefore it is not limited in this respect. Numeric XEQ's are always long-form; they require 3 bytes of program memory and they always store the jump distances to corresponding labels.

The criteria you should use for selecting short-form or long-form local GTO/LBL pairs is jump distance. If you use only long-form local labels, then your program will run fast because the jump distance will always be stored. But, if the jump distance is less than 112 bytes, you can get fast operation, and save program space, by using short-form GTO/LBL pairs. (You can determine jump distance by summing the byte requirements for each instruction between the GTO and its LBL. Appendix D in the owner's handbook lists byte requirements.)

The third category of local label is the local-ALPHA label. The letters A through J and a through e define local-ALPHA labels. These labels are identical in function to the long-form local label except that if a local-ALPHA label is used in a program, and the HP-41 is positioned to that program, then the key that corresponds to that label ([A] for LBL A) is automatically assigned to that label in USER mode. This is convenient for such things as accessing input routines or re-initializing a program without having to key in XEQ "label" or make a global key assignment. This is also convenient in that, with each new program, a new set of automatic key assignments becomes available.

IN A USERS' LIBRARY PROGRAM:

Library procedure and customer convenience demand at least one global label in a program. All submitted programs must have one or more global labels, and we prefer a global label at line 01. Global labels demand complete documentation; you must expand on why each global label was used and where the user will be after executing that global label.

Because of the difficulties associated with using superfluous local-ALPHA labels when there are an abundance of alternative local numeric labels, such unnecessary labels will not be acceptable. Local-ALPHA labels demand superb documentation; expand on why each local-ALPHA label was used and where the user will be after execution of that local-ALPHA label. Local-numeric labels do not require such extensive documentation.

Programs that require global label key assignments will not be accepted. Key assignments are a convenience feature; it should be left to the user to decide which key assignments are desirable. Manditory key assignments can hopelessly confuse people who are used to their own key assignments.

POSITIONING THE HP-41 TO A PROGRAM

A local label is accessible only when the HP-41 is positioned to the program in which that local label is found. If this program has a global label in it, then positioning is easily done by executing GTO, followed by the global label.

The catalog function is a convenient means of scanning the contents of program memory for the programs that you have written. What you see when you execute [CATALOG] [1] is a sequential listing of all of the global labels you have used in programs and all of the END's between those programs.

Occasionally, you will write a program that contains only local labels. Local labels do not appear when you execute [CATALOG] [1]; so, the only thing that would appear to indicate the presence of a program with only local labels would be the END corresponding to that program. Positioning the HP-41 to a program that contains only local labels is done by using the catalog function. Simply key in [CATALOG] [1], press [R/S] at the desired program, and clear the display by pressing [←]. Or, press [CATALOG] [1] and [R/S] immediately, then use [SST] to position the HP-41 to the desired program.

ADDITIONAL NOTES ON LABELS

1. When a global label is defined, the distance from that global label to the first preceding END is stored with it. This storage facilitates not only the running of the catalog but also the programmatic search for global labels. When a global label is sought, the first place the calculator looks is the permanent program end (.END.). At this location, the position of the preceding END or global label is stored. With this information, the calculator jumps to the END or global label, repeating the sequence until the desired label is found. Note that although this is an internal time-saving procedure, the search for global labels happens every time a global label is called; therefore, it is considerably slower in repetitive routines than the direct jump process of the local label.
2. Since the global label search is backward from the permanent program end (.END.) to the first program in memory, if two identical global labels exist in memory, then only the label closest to the end of memory will be accessed upon execution of that global label. If you wish to access the other program with the identical global label, then you may do so by positioning the HP-41 to that program using the CATALOG function as explained in the last section under "Positioning the HP-41 to a Program."

(Continued)

3. Since the setting and clearing of USER mode is programmable (SF 27 and CF 27 respectively), the setting of flag 27 should be performed at any time that the use of local-ALPHA labels are "required" by the program. Do not use local labels on segments of your programs that you do not want automatically assigned to the keyboard.
4. In contrast to the reverse search pattern for global labels, local labels are searched for from the point of call forward. This allows the flexibility to use more than one local label with the same number if one or more short forward jumps are needed.

LABELING A SUBROUTINE

Whenever a subroutine is called by its resident program, and you choose to use a global label on this subroutine to make it universally accessible, also consider placing a local label right after the global label. This extra label will use a maximum of two bytes of program memory, but, it saves bytes in the long run! The reason this local label will save bytes is that every GTO or XEQ statement that would call the global label from the resident program can call the local label instead. GTO and XEQ statements calling a local label use less memory than those calling a global label. Also, with a local GTO/LBL pair, you are taking advantage of the rapid-jumping capabilities of the HP-41 (see the section: "Local Labels"), and you are avoiding the relatively long global label search every time you call the subroutine from its resident program (see the section: "Global Labels").

Answers to HP-67 Test

Numbers in parentheses after an answer refer to page numbers in the HP-67 or HP-97 owner's handbook.

1. NOP can be used in a program when DSZ or ISZ are used purely as counters. For example: ... DSZ, NOP, RTN. Sometimes an alternative, however, is to repeat the non-branch option: ... DSZ, RTN, RTN.
2. LSTx will be different for the two cases. (67 = p.67; 97 = p.58)
3. GTO LBL a. (67 = p.240; 97 = p.215)
4. X=−23.00, Y=1.00, Z=−1.00, T=0.00. CHS does not affect stack lift.
5. Single-stepping will cause an **Error** signal; running the program will not.
6. Pi. LSTx can often be used as a temporary register. (67 = p.323; 97 = p.294)
7. When flag 0 is set and flag 3 is clear, or when flag 0 is clear and flag 3 is set.
8. The T-register. (67 = p.56-57;97 = p.47)
9. R↑, X≷Y, R↓, R↓. (67=p.54-56; 97=p.48-50)
10. X=100, Y=3.00, Z=3.00, T=4.00.
11. X≤Y, X=Y, FO? Flag 0 must be clear (any unused flag would do). If you tried X≷Y you must be sure x is in the right register before proceeding.
12. The program will stop after executing the RTN in step 013. Note that step 032 sends the program to the next occurrence of LBL 2. (67 = p. 206-207; 97 = p. 185-186)
13. The program will loop indefinitely because a RTN is never encountered.
14. See appendix C in the owner's handbook. (67=p.320-321; 97=p.291-292)
15. X≷Y, ENTER, ENTER, R↑, ÷, LSTx, X≷Y, INT, x, −
16. No program is completely debugged except for the data you've tried. Check your program with the following in SCI 9 mode: (−4/−3) R=−1; (−4/3) R=−1; (4/−3) R=1; (4/3) R−1.
17. 91.3130; time is normalized so that minutes and seconds will always be less than 60.
18. Correct answer is (b). (67=p.90-91; 97=p.82)
19. LBL A, RCL I, STO(i), 2, 5, CHS, X=Y, RTN, DSZ, GTO A. Watch your initial and final conditions so that R0=0 and R25=−25.
20. Correct answer is (c). (67=p.288; 97=p.260)
21. R0=0.00, R1=−1.00, ... R7=−7.00. Primary data will always be loaded ahead of secondary data regardless of which side is read first.
22. The following solution is based on the property that each succeeding term in the series is twice the previous term divided by n+1. LBL A, EEX, 2, STO 1, EEX, LBL 0, ENTER, +, RCL 0, EEX, +, STO 0, ÷, STO+1, DSZ, GTO o, RCL 1, RTN. The answer is 6.38906.
23. Correct answer is (d). (67=p.275; 97=p.248)
24. There are 25 steps loaded into memory from 200 to 224. (67=p.277; 97=p.250)
25. Three ways to set flag 3 are: entering any digit from the keyboard; reading a data card; and SF3. Flag 3 also can be set with magnetic card header information. When constants are stored as program steps, single-stepping will cause flag 3 to be set as well. (See also: problem 5, where SST produces an **Error** condition while RUN does not.)

Magnetic Card Erasure

We often get letters concerning the possibility of accidental erasure of a magnetic card for the HP-65, HP-67/97, or HP-41 calculators. And since so many readers ask so many questions about this subject, we decided to bring you a definitive article on the subject.

Usually, there are two forms of energy that might significantly affect the recording on a magnetic card. These are magnetic fields and heat. Dirt also can affect the recording.

DIRT

When a magnetic card does not work properly, the first thing to do is check it for dirt, body oil from your fingers, and so forth. Clean the bottom side of the card with mild soap and water or with alcohol and then try it again. Usually, it will work all right after a cleaning.

MAGNETIC FIELDS

All magnetic materials have an intrinsic property called coercivity. This property is measured in units, called Oersteds, that define the intensity of a magnetic field that is required to demagnetize or erase the material. For example, a material with a coercivity of 100 Oersteds will be erased by a magnetic field with a magnitude of 150 Oersteds, but a material with a coercivity of 300 Oersteds would not be erased by the same field. Since a magnetic card has a coercivity of approximately 265 Oersteds, erasure of information stored on the card would occur only if a field approaching that value were to penetrate into the card.

Where might a field of such intensity be found that could cause accidental erasure? The fact is, a field of this magnitude is very rare. The Earth's magnetic field has a strength of approximately 0.6 Oersted. The field from an electric hand drill under full load is on the order of 10 Oersteds at the surface of the drill case. A magnetic card pressed against the body of the drill would be in no danger of being erased.

The magnetic fields surrounding heavy power wiring can be intense directly at the surface of the wire, but at the point where contact with the wire can be made (through insulation or conduit), fields will not be capable of erasing a card. If a line cord were laid directly across a card and then short-circuited, the short-circuit field intensity could possibly erase the portion of the card in the immediate areas of the line cord. The electron beam in a television picture tube is deflected magnetically. The fields directly around the deflection yoke of a TV set could erase a card; the fields outside the TV cabinet are much too low to have any effect.

It can be stated without reservation that magnetic fields surrounding ac-operated equipment and appliances in both the home and in an industrial environment are not capable of erasing a magnetic card. The same, however, is not true of permanent magnets. Improvements in permanent

(Continued)

magnet materials in recent years have led to increased use in a wide variety of applications ranging from pencil holders to refrigerator door latches. These small magnets have surface intensities as high as 1500 Oersteds. If, for example, a card were to touch a magnetic door latch, information might be lost from the portion of the card that came in direct contact with the magnet. Magnets that are sewed into towels or drapes or those that are used to secure notes to metal bulletin boards are also of sufficient strength to cause erasure of recorded material at their area of contact. Other objects in common use would include magnetic key chains and flashlights that have magnets affixed to their cases. Even small permanent magnets could be very troublesome if it were not for one very helpful factor—the factor of distance.

A magnetic field falls off in intensity approximately as the square of the distance from the magnet. In other words, if one doubles the distance at which a field is measured, the intensity will be only one-fourth as great as at the shorter distance. From this we see that distance is what really protects a card from accidental erasure. While the field intensity of a magnet may be sufficient to erase a card placed in contact with it, a spacing of just an inch (2.54 cm) can often totally protect the card.

It can be seen that the mere presence of a magnet in the same room with the card would not be the cause of any concern, and the magnet would not be capable of doing any harm.

A magnetic field can be partially shielded by a box made of soft iron, but this would be an expensive and cumbersome way to attempt to protect a card from accidental erasure. Since a spacing of just about an inch or two (2.54 to 5.08 cm) will protect a card from even the strongest commercially available magnet, distance becomes a more certain and more easily applied method of protection.

Cards do not have the ability to "conduct" the magnetism applied by a permanent magnet from one card to another. There is no chain reaction effect.

HEAT

Excessive heat, such as that which may be generated by a fire, can damage magnetic cards. In general, however, the damage is purely physical and not magnetic in nature. High temperatures can soften or otherwise change the characteristics of the card.

Materials that normally exhibit magnetic properties tend to lose their magnetic abilities when their temperature is increased. The point at which this occurs varies with different substances. As a practical matter, one does not need to worry about exposing a card to elevated temperatures as long as those temperatures do not affect the card physically. The environments in which humans normally function will not harm a magnetic card.

OTHER FORMS OF ENERGY

The world is full of electrical and magnetic phenomena such as lightning discharges, radio transmissions, static electricity, and high-power radar beams. Of all of them, only lightning discharge could possibly cause accidental card erasure and then only if the discharge occurred within a few inches of the card. (*And it probably would do more damage to you than to the cards!!*)

A magnetic field of several thousand Oersteds may exist directly in front of a radar antenna, but the field intensity drops to below the critical level only a few yards away. However, the burning hazard to humans within this high-intensity area is so great that precautions are taken to keep the immediate area in front of the antenna well clear of personnel and other objects.

ANTI-HIJACKING DEVICES

These devices are used in airline terminals to detect concealed weapons that may be carried aboard aircraft and in stores and libraries to detect unauthorized removal of merchandise and books. There are two basic types in use.

Most of the equipment in use at various airports throughout the nation could be classed as passive devices. They are designed to detect subtle changes in the Earth's normal magnetic field. The shape of the Earth's field is distorted by metal objects, and their presence will cause a change that can be sensed by the detection unit. These units do not generate a field of their own and, therefore, cannot erase or otherwise harm magnetic cards.

A second type of detector is classified as an active device. Usually a doorway or walkway is surrounded with a moderately intense magnetic field. While such a field will not totally erase a card, it could reduce the amplitude of the recording on it. Most of the detection systems operate with a field less than 20 Oersteds, and one could carry recorded cards through such a field without affecting the recording.

Unless you are sure that the detection device is of the passive type or a low-power active type, care should be exercised when transporting recorded cards through detection stations. It would be advisable to inform the attendant operating the unit that you are carrying recorded cards. If there is any doubt as to how it will be affected, request that they not be passed through the monitor and ask instead that they be visually inspected. The extra few minutes that this might take may be a worthwhile investment to preserve a recording.

X-RAY INSPECTION

From time to time we hear of packages that have been subjected to X-ray inspection during shipment. After subjecting cards to quantities of X-radiation far in excess of what would be expected for routine package examination, no incidence of signal decay or erasure has been found.

Petroleum Fluids Pac Released

There is probably not a country in the world that has been unaffected by the so-called "energy crisis." Certainly, "oil" is now one of the best-known words in *any* language. Knowing that, Hewlett-Packard decided to contribute to helping the world energy crisis by developing an HP-41 Pac for the petroleum industry. The Pac we chose to produce is for reservoir engineers; **it is NOT a drilling Pac.** Here, then, is Hewlett-Packard's first software package designed for the petroleum industry.

The Petroleum Fluids Pac has 18 programs to calculate fluid properties for petroleum fluids (gases, oil, and water or brine), as follows:

FOR GASES:

- Z Factor
- Isothermal Compressibility
- Formation Volume Factor
- Viscosity
- Pseudocritical Temperature and Pressure From Gas Gravity
- Gas Properties From Composition

FOR OIL (Above or below the bubble point):

- Isothermal Compressibility
- Formation Volume Factor
- Viscosity
- Gas-Oil Ratio
- Bubble Point Pressure
- Two-Phase Formation Volume Factor

FOR WATER (Including salinity and gas saturation corrections):

- Isothermal Compressibility
- Formation Volume Factor
- Viscosity
- Gas-Water Ratio

OTHER PROGRAMS:

- Rock Compressibility
- Total Isothermal Compressibility

The most significant feature of this new Pac is the Unit Management System, which is a simple, powerful approach to handling both the computational and dimensional aspects of reservoir engineering problems. With this feature, when you are prompted for an input, you key in the number you want to use. If the number has units different than the expected default units, you merely spell out in the ALPHA register the units you want. It's that simple! The Pac will take care of all of the problems of dimensional consistency for you, and it will calculate results in whatever units you specify. And, best of all, you have a virtually unlimited set of units from which to choose.

Another extremely important feature of the Pac is its modularity. In addition to the 18 programs listed above, the Pac has a library of 54 subroutines for input, output, and fluid property calculations. This

(Continued)

library of subroutines allows you to easily write programs to solve difficult reservoir engineering problems (such as material balance) in a fraction of the time and program space required without the Pac.

All 18 programs, the Unit Management System, and the 54 subroutines are documented in a comprehensive 200-page manual that includes annotated program listings. Also included in this Pac is a detailed 10-page Quick Reference Guide.

But, as impressive as all of this may seem on paper, it has to be experienced to be appreciated. We spent a lot of time with petroleum industry consultants to make sure that we were going in the right direction with this important Pac. The author even attended such industry-related functions as the Offshore Technology Conference. And, last but not least, many industry professionals were queried by the author before the Pac was finalized. So drop by your nearest HP Dealer after September 1, and try a "hands-on" demonstration. And if you are in the petroleum industry, we'll be amazed if you don't take the 00041-15039 Petroleum Fluids Pac home with you.

The price? Well, because this is a larger-than-usual Pac, it is more expensive than some previous Pacs. And because prices vary because of shipping, import duties, and so on, you will have to contact the dealer in *your* area for pricing information.

Routines, Techniques, Tips, Et Cetera ...

The routines and techniques furnished in this column (formerly "25 Words") are contributed by people from all walks of life and with various levels of mathematical and programming skills. While the routines might not always 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.

In Berlin, Germany, **Diethard Skaliks** came up with this routine that works on the HP-41, the HP-67, HP-97, and many other HP scientific programmables. We have it coded here for the HP-67 or HP-97.

(67/97) Here's another version of the A/B = C/D ratio routine shown in V3N4p12b. It uses only 10 lines, and it does work quite well—as do the other versions we have printed in KEY NOTES.

Start by entering the data into the stack, A↑, C↑, D↑, B. The unknown should be entered as zero (0) in the appropriate position.

```
001  *LBL0     006    →P
002    R↓      007    R↓
003  X≠0?      008   TAN
004  GTO0      009    ÷
005    R↓      010   RTN
```

Sunshine is sometimes a scarce commodity here in Corvallis, Oregon, but in sunny Livermore, California, **Jim Carley** is keeping his mind on other commodities.

(67/97) The piece in "25 Words" on Richard Sperling's "Effective Interest Cost" (V4N1p5c) was a little confusing in that the amount and rate of interest were mixed. However, the program itself shows that *amount* is correct for entry. Using the same entry and operating instructions, the following routine will do the job in 10 steps instead of 17. The operating instructions from Richard Sperling's routine are: Input the amount borrowed [ENTER], the *amount* of interest [ENTER], and the tax rate (from your income tax bracket). Press [A] and the routine will calculate the effective after-tax cost and the effective after-tax interest rate.

Enter the interest and tax rates—this should read amount of interest and income tax rate—as whole numbers; that is, enter 12 percent as "12," and a 50-percent tax bracket—Wow!—should be entered as "50."

```
001  *LBLA     006     ÷
002     %      007   EEX
003     -      008     2
004  PRTX      009     ×
005   X≷Y      010   RTN
```

(We've had a lot of "interest" in this particular routine. In fact, I rewrote it in only six lines in V4N2p11a. Evidently Mr. Carley never saw that routine—Ed.)

This next contribution is one *mean routine!* It was sent to us from **David L. King** of Grand Junction, Colorado. We think he means to give us the means to get some weighted means.

(67/97) The following routine will calculate "weighted" means on an HP-67/97. It makes use of the summation routine. To run it, enter the data sets by pressing the "straight" average number, [ENTER], the "weight," and then press [E]. After entering all data sets, press [D]. This will print the "straight" and "weighted" means and clear the registers for the next set of data.

```
001  *LBLE     010    x̄
002   ENT↑      011   X≷Y
003     R↓      012  PRTX
004     ×       013    ÷
005     R↑      014  PRTX
006   X≷Y       015  CLRG
007     Σ+      016   P≷S
008   RTN       017  CLRG
009  *LBLD      018   RTN
```

(The straight means that Mr. King refers to is actually the average of the weights. If you don't need to know the average of the weights, delete line 12—Ed.)

Next, we bring you a clever routine contributed by **Pedro Henrique Perez de Moura** of Porto Alegre, Brazil. He has found a unique use for three of the Display Format Flags.

(41) Here is a routine for decimal-to-binary conversion. It converts integer decimal numbers up to 16777215 and puts the corresponding binary number in the ALPHA register. The little "trick" in this routine is the transformation of single-digit octal numbers to three-digit binary numbers by fixing the number of decimal places displayed by the calculator indirectly to the octal digit and then testing the three display format flags 37, 38, and 39. These flags are set in a binary representation of the places displayed (see page 231 of the owner's handbook). The routine uses just one label and no memory registers. It destroys the stack, but occupies only 58 bytes. But, best of all, the conversion is fast; the worst case takes only 6 seconds.

```
01↓LBL A          17 FC? 38
02 OCT            18 "⊢0"
03 ENTER↑         19 FS? 38
04 LOG            20 "⊢1"
05 INT            21 FC? 39
06 STO Z          22 "⊢0"
07 10↑X           23 FS? 39
08 /              24 "⊢1"
09 CLA            25 FRC
10 ISG Y          26 10
11↓LBL 00         27 *
12 FIX IND X      28 DSE Y
13 FC? 37         29 GTO 00
14 "⊢0"           30 FIX 2
15 FS? 37         31 AVIEW
16 "⊢1"           32 END
```

Now, let's travel to Germany to see what **Hendrick Wolper** of Clausthal-Zellerfeld is doing with his HP-41.

(41) This short routine gets "down" the exponent that occurs in the SCI format. It is very useful for rounding-routines and similar programs.

```
01↓LBL "EXP"      06 +
02 ENTER↑         07 INT
03 ABS            08 100
04 LOG            09 -
05 100            10 END
```

While we are visiting Germany, let's stop by Bingen am Rhein and see how **Johannes P.C. Heidecker** is using his HP-41.

(41) When changing from the HP-38E/C to an HP-41, perhaps you may miss the %T function. If you think that other KEY NOTES readers are interested in this function, here is a short version of it for the HP-41.

```
01♦LBL "%T"     05 100
02 ENTER↑       06 *
03 RCL Z        07 END
04 /
```

(In case you are not familiar with the HP-38E/C, %T calculates the percent that X is of Y—Ed.)

Now, here is a routine from **Axel Harvey**, who lives in Quebec, Canada. It works well with positive input.

(41) Long Division. One may need to know the decimal expansion of a quotient to a greater number of places than a calculator can provide. This routine uses long division to print-out as many decimal digits as necessary. Enter the numerator in register Y and the denominator in X, then execute the "LONG" routine. A pause occurs after each line of output; have a finger on a neutral key (I prefer [SST]) before the pause, and execution will stop with flag 29 reset and printer buffer clear.

```
01♦LBL "LONG"     23 RCL 03
02 STO 03         24 MOD
03 X<>Y           25 10
04 STO 02         26 *
05 CLA            27 STO 02
06 PRX            28 RCL 03
07 X<>Y           29 /
08 PRX            30 INT
09 /              31 ARCL X
10 INT            32 ISG 00
11 FIX 0          33 GTO 03
12 ACX            34 ACA
13 PRBUF          35 1
14♦LBL 01         36 SKPCHR
15 CF 29          37 CLA
16 .003           38 ISG 01
17 STO 01         39 GTO 02
18♦LBL 02         40 ADV
19 .004           41 SF 29
20 STO 00         42 PSE
21♦LBL 03         43 GTO 01
22 RCL 02         44 END
```

Quebec, Canada, is a beautiful place, and it must provide a prime atmosphere for a hard-working mind! Why? Well, here's *another* contribution from **Axel Harvey**.

(41) Pseudo-Random Subroutine. Here is a process for a series of apparently random numbers, all between zero and 1. It requires two seeds, the first seed in R01 and the second in R02. The first seed may be any number greater than zero, the second any number greater than 10^{-9}. The first seed influences the resulting series considerably; a difference of a quarter of a percent in the value of the first seed can change the whole series except for the first two or three digits of the first number output. The second seed also can vary series considerably—a difference of 0.01 in absolute value can be significant—but only if it is less than 1. If it is greater than or equal to 1 it will have no influence on the series. The value 167 in line 8 may also be changed, but its effect is much more gradual; a difference of 50 or so may not make any difference for the first dozen terms of a series. I recommend that this value be kept between 100 and 200. If it is too small, there will be many pairs of adjacent numbers beginning with the same digit; if it is too big, there will be many numbers with trailing zeros.

```
01♦LBL "X"      08 167
02 RCL 01       09 MOD
03 LN           10 X<> 02
04 ENTER↑       11 MOD
05 FRC          12 FRC
06 1/X          13 STO 01
07 X↑2          14 END
```

This next technique is useful to *anyone* who programs the HP-41. It was contributed by **Andrew Zawadzki**, who lives in Birmingham, Michigan.

(41) Several times I found the need for a routine that would allow me to execute an instruction only every *second* time, while the program was running in a loop. Maybe the routine I came up with will be of interest to others. It is especially useful in viewing the ALPHA register. However, in order to *not* execute "INSTRUCTION" in the first run of the loop, flag 05 should be set before the loop starts. Of course, any other flag may be used.

```
Begin loop      (Note that FS?05 at the
    :            beginning is not necessary.
FS? 05           The two lines that follow
FC?C 05          will invert the status of flag
SF 05            05—Ed.)
FS? 05
INSTRUCTION
    :
End loop
```

Now, let's jump around the globe a little bit and see what other interesting things are being done on the HP-41. First, we land in Goteborg, Sweden, and we find that **Thomas Fange** has a contribution for us.

(41) This routine will "turn" an integer. For example, 124 will become 421, or 1981 will become 1891.

```
01♦LBL "TURN"   10 ST+ Z
02 0            11 -
03 X<>Y         12 X≠0?
04♦LBL 00       13 GTO 00
05 10           14 X<>Y
06 ST* Z        15 10
07 /            16 *
08 ENTER↑       17 END
09 FRC
```

(This routine sure would be handy during income tax time. Imagine being able to "reverse" a tax of, say, $9,862!! Of course it wouldn't be legal, but it might make one feel *better—Ed.)*

Next, we land in Jakarta, Indonesia. Here we find **John O'Shannessy**, who contributes a short routine that can save all of us some time.

(41) HP-41 owners should make a note to key in this short (shortest?) program to restore prompt execution of the two top rows of keys after running your "user" programs. Instead of ending programs with RTN or END, insert a GTO U instruction; any non-local ALPHA label may be used. And label U? It's simply: 01 LBL U, 02 END. Yes, GTO .. achieves the same result but is a nuisance, and it is usually executed *after* noting a slow response time, which can be quite substantial.

Okay, Mr. O'Shannessy, but you must "separate" it from other programs "above" it in memory. Be sure to GTO .. before you key it in—Ed.)

Meanwhile, back in Bensheim, Germany, we find that **Christoph Wetzel** is *always* on time. The routine that he has submitted reveals his secret of punctuality.

(41) After receiving your last issue of HP KEY NOTES for the first time, I was especially inspired about the "timer" routine from Graeme Leith (V4N3p.11b). But my dilemma is that I always forget important times. So I wrote the following routine that changes the HP-41 into a perfect alarm clock. It is necessary to use the function "PACK" after programming it.

```
01♦LBL "A-CLOCK"    23 HMS+
02 CF 01            24 STO 01
03 "ALARM TIME?"    25 VIEW X
04 PROMPT           26 24
05 STO 02           27 X<Y?
06 "CURRENT TIME"   28 GTO 03
07 PROMPT           29 GTO 00
08 STO 01           30♦LBL 04
09 FIX 4            31 .000091
10 RCL 02           32 GTO 05
11 X<Y?             33♦LBL 01
12 SF 01            34 FS? 01
13♦LBL 00           35 GTO 02
14 RCL 02           36 RCL 02
15 RCL 01           37 VIEW X
16 X>Y?             38 BEEP
17 GTO 01           39 GTO 01
18♦LBL 02           40♦LBL 03
19 FS? 01           41 ST- 01
20 GTO 04           42 CF 01
21 .000082          43 GTO 00
22♦LBL 05           44 END
```

("Perfect" alarm clock? Be careful; timer routines are fun to use on HP calculators but the accuracy of these routines is highly dependent on temperature, battery condi-

(Continued)

tion, and the unique characteristics of various operating situations. You can customize this routine for your HP-41 by adjusting the constants in lines 21 and 31. But don't be surprised if this "alarm clock" sounds ½-hour late on a cold morning! Also, flag 21 must be cleared if you have a printer—Ed.)

Moving on to New South Wales, Australia, we find a contribution from **David Morgan**. His is one of the many, many variations of "restricted access" routines. Another such routine appeared in V4N3p5a under the flag 11 explanation.

(41) Being a new owner of an HP-41C and all the peripherals, I thought you might want to see this small access code routine to thwart "non-HP-41" users. It can be used at "turn-on," using SF 11, or it can be attached to the beginning of another program. The owner's own unique code is inserted at line 02, and it can be ALPHA, numeric, or a combination of both. The routine can be reduced to just 20 lines by deleting lines 07, 08, 15, 16, 17, 19, 20, 24, 25, and 26.

Thanks for a great product and an entertaining newsletter; I wish it came more often.

```
01♦LBL "CODE"      16 AVIEW
02 "KEYNOTES"      17 PSE
03 ASTO Y          18 AOFF
04 "   CODE?"      19 CLD
05 AVIEW           20 CLX
06 AON             21 RTN
07 TONE 9          22 GTO "CODE"
08 PSE             23♦LBL 02
09 PSE             24 "--WRONG--"
10 ASTO X          25 AVIEW
11 X=Y?            26 BEEP
12 GTO 01          27 SF 11
13 GTO 02          28 OFF
14♦LBL 01          29 GTO "CODE"
15 "**CORRECT**"   30 END
```

(It works, Mr. Morgan, but for people with printers attached, you should have a CF 21 for line 02. Thanks for the compliment; perhaps one day we can *issue this newsletter more often—Ed.)*

If you think that HP-41's don't get to faraway places, you haven't talked to **Neil Hunter-Blair**, who sent this input from Bangkok, Thailand.

(41) If you have a card reader for your HP-41, have you ever wanted VER in a program after a WDTA or WDTAX? Non-programmable, eh? Well, try this:

With your card reader plugged in, turn on the HP-41 and press [ASN], [ALPHA], VER, [ALPHA], [LN]. Turn off the HP-41 and remove the card reader. Turn it back on and press [USER] and [PRGM], then press [LN]. In the display, the program line will be NN XROM 30,05, but when the card reader is connected the display will show NN VER.

You can enter WALL and WPRV into a program by the same method. If you have a Wand, you can enter any of these functions into a program by first removing the card reader and then scanning the function on the Paper Keyboard.

(Yes, it does work. No, it won't harm the HP-41. It wasn't in the owner's handbook because, at the time the machine was designed, those weren't intended as programmable functions. Other nonprogrammable functions can be entered into a program in the same way, but not all of them will execute when a running program encounters them—Ed.)

In the last issue (V5N1p7c), we printed an alternative to AVIEW that works with or without the printer. **Paul Cardinale** of San Francisco responded with these two shorter routines that do essentially the same thing.

(41) The routine "AV" in the right column of page 7 in Volume 5 No. 1 can be replaced by the following routine.

```
01♦LBL "AV"        07 CF 21
02 FS? 21          08 AVIEW
03 SF 15           09 FS?C 15
04 SF 21           10 SF 21
05 FS? 55          11 RTN
06 PRA             12 END
```

(Any user-controlled flag can be substituted for flag 15—Ed.)

(41) Referring to the same article referenced above, if you want flag 21 to control printing, you can use this shorter routine.

```
01♦LBL "AV"        06 AVIEW
02 FS? 55          07 FS?C 15
03 PRA             08 SF 21
04 FS?C 21         09 .END.
05 SF 15
```

This input is from **Wyman W. Trotti, Jr.** of Cayce, South Carolina. It is *another* nice routine for use in conjunction with the printer.

(41) There may be a better solution to this problem, but this one at least works. I am in the real estate business, and most of what I want to plot, using my HP-41 printer, cannot be generated by a function. The following routine stores data sequentially in registers beginning at R_{13} and, when called by PRPLOT or PRPLOTP, returns it. First, assign DATA2 to a key and use it to initialize the program and store data. As written, the program will store 50 values. Line 15 may be changed to accommodate more values. Although set for 50, you do not need to store 50 values.

Have PRPLOT call DATA to generate the plot. If the same data-set is to be plotted more than once, the program must be reset (RST) before each plot by executing DATA2 once. Data simply ignores input from PRPLOT and outputs the stored values sequentially.

```
01♦LBL "DATA2"     13 GTO A
02 SF 00           14♦LBL 01
03 XEQ 01          15 13.063
04 "RST"           16 STO 12
05 AVIEW           17 RTN
06♦LBL A           18♦LBL "DATA"
07 "N=?"           19 FS?C 00
08 PROMPT          20 XEQ 01
09 ARCL X          21 RCL IND 12
10 PRA             22 ISG 12
11 STO IND 12      23 END
12 ISG 12
```

Now, how about something for printer fans? This routine was contributed by **William J. Quinlan, Jr.** of Evanston, Illinois, and it is his first input for KEY NOTES.

(41) This routine creates any type of divider on printed tapes by simply inputting the decimal equivalent and the size of the line, which can be 0 to 23. A divider bar can be created in default by pressing R/S after the prompt.

```
01♦LBL "DV"        13 ISG Y
02 "SIZE↑CHAR"     14 GTO 01
03 PROMPT          15 PRBUF
04 X=0?            16 CLX
05 GTO 02          17 RTN
06 STO 01          18♦LBL 02
07 X<>Y            19 .023
08 .001            20 ENTER↑
09 *               21 31
10 RCL 01          22 GTO 01
11♦LBL 01          23 END
12 ACCHR
```

Lately, we have received many letters about this next subject. And since the letters were all pretty much the same, we decided to print the first one we received. It came from **Claude Roeltgen** of Mondercange, which is in the Grand Duchy of Luxembourg.

(41) I am very glad to submit some tricks I found for my HP-41. Sometimes you have to express a complex condition; for example: if (x>y? or FS?00) then XEQ 05. Normally, you would have to do:

```
01 x>y?
02 GTO 01
03 FS?00
04 GTO 01
    .
    .
nn LBL 01
nn XEQ 05
```

But you can shorten this by doing:

```
01 x≤y? (inverse of first condition)
02 FS?00
03 XEQ 05
```

In the same manner, when you have two conditions *separated* by "and," you can shorten the solution of your problem. For example: if (x=0? and FS? 05) then XEQ 10. This would give:

```
01 x=0?
02 FC?05 (inverse of second condition)
03 GTO "CONT"
04 XEQ 10
05 LBL "CONT"
```

A further useful trick when you have to invert the status of a flag (from clear to set, or set to clear) is:

```
01 FC?Cnn
02 SFnn
```

The nn means "number or numbers," as any flag number can be used.

(Flag tests work the same way. For example:

FS?aa		FC?aa
GTOnn	becomes:	FS?bb
FS?bb		GTOnn
GTOnn		

As you can see, flags are extremely useful "tools" on the HP-41—Ed.)

Here's a contribution for those of you who use HP-67/97 programs in your HP-41. It was sent to us by **Siegbert Förster** of Rosstal, Germany.

(41) Recently I tried to run, on my HP-41, a program that I have often used on my HP-97, but it did not work. So, I have discovered this translation problem.

In cases of HP-67/97 sequences like:

```
001   F0?
002   Σ+
```

the HP-41 translation is

```
01♦LBL 67    03 ΣREG 14
02 FS? 00    04 Σ+
```

which is obviously wrong.

It is simple to overcome this translation problem by switching the conditional with the ΣREG 14 instruction.

Arnold Hinrichs, who lives in Hagen, West Germany, has been doing some very practical work on his HP-41. His contribution is a unique routine that is applicable to many situations.

(41) My contribution has to do with execution times. Since the HP-41 is not a large-scale computer system, execution of self-made programs (RAM) can sometimes last a long time. But, we *can* shorten the time that the user will have to wait for an output.

Thousands of students will use their HP-41 in schools, colleges, or universities to compute values for a function that they have to draw. They will write a program that will generate some values according to the function, and then use either STOP or PROMPT to display the result. Each time R/S is pressed, it will take some time until the next value is displayed. Time is valuable in exams!

Can we make it better? We can! What about this subroutine instead of STOP or PROMPT?

```
01♦LBL "DISP"    06 FC? 03
02 FS? 02        07 AVIEW
03 STOP          08 SF 02
04 FS? 03        09 RTN
05 VIEW X        10 END
```

It requires a cleared flag 2 at start, and flag 3 must be set or clear according to the desired output. When flag 3 is clear the ALPHA register is displayed, otherwise DISP will display X.

Now the description: The main program will compute the first value, which will be ready for output in either the X-register or the ALPHA register. If XEQ DISP is used instead of STOP or PROMPT, then either X or ALPHA is displayed and flag 2 is set. While our student is busy writing down the first result, the HP-41 is busy calculating the second result. The PRGM annunciator will stay on until XEQ DISP is encountered the second time. Since flag 2 is set now (don't clear it in your program!), execution will stop at line 03. When R/S is pressed now, the next value will be displayed immediately. While our poor student is writing down the second value, his main program will compute the third value and so on. But never press R/S when the PRGM annunciator is still on!!

(Again, don't forget that flag 21 needs to be clear—Ed.)

Richard Manahan lives in Wichita Falls, Texas, with an HP-41C and an HP 82143A Printer. That he is getting along well with both of these friends is demonstrated in this next routine.

(41) While running a program that generated data that I wanted to print out on the HP 82143A Printer, I got hopelessly lost trying to get it lined up. In trying to find an easier way, I came up with the following routine.

```
01♦LBL "B1"    07 ARCL 03
02 CLA         08 ARCL 25
03 ARCL 01     09 ARCL 06
04 ARCL 25     10 ACA
05 ARCL 00     11 ADV
06 ARCL 25     12 RTN
```

Essentially, the data was stored in registers 01, 00, 03, and 06. From the keyboard I stored a "blank" in register 25. In the main program, at the end of data generation, I used an XEQ "B1" label. Since the ALPHA register has 24 spaces, the same as the printer, I found this a lot easier to use than the skip-character, etc., exercise in ACX.

One caution: This routine is sensitive to the FIX mode that you are in, and it will truncate data if it overflows. It is relatively easy to manually line-up your data in the ALPHA register and then tailor the routine around it.

We *do* think this next routine is a "neat" routine. It is a quick routine to find x^3. **Frank Wales** of Glasgow, Scotland, is the contributor of this jewel.

I don't know about you, but I think this is neat! It works just like any other monadic function in that it puts x in LASTX and it leaves the stack untouched (apart from X).

```
01♦LBL "CUBE"
02 X↑2
03 X<> L
04 ST* L
05 X<> L
06 RTN
```

For those of you who are byte-crazy (aren't we all?) and who do not mind sacrificing a few significant figures in order to save some bytes, we present this routine. It was sent to us by **Leonard D. Cordwell** of Romsey, England.

(41) A data register can store 10 digits, a decimal point, and a minus sign. If the data to be stored does not need all that accuracy, then more than one piece of information can be stored in one register. The SAVE routine I enclose is designed to store (in IND 00) two 5-digit numbers, each consisting of a 4-digit integer and one decimal place. One number (originally in the X-register) is multiplied by 10 to turn it into an integer, and the other (originally in the Y-register) is divided by 10^4 to turn it into a fraction, so that both numbers can be stored, one on each side of the decimal point. Also, since both numbers in this case are always positive, the minus sign is spare and is used to record the status of a flag. USE reverses the process to put the data back into its original form and into the registers it came from.

This routine uses 58 bytes as it stands (this can be reduced by using local labels), but it has saved 80 data registers in one of my HP-41 programs.

```
01♦LBL "SAVE"    14♦LBL "USE"
02 FIX 1         15 RCL IND 00
03 RND           16 X<0?
04 1 E1          17 SF 01
05 *             18 ABS
06 RCL Y         19 ENTER↑
07 1 E4          20 FRC
08 /             21 1 E4
09 +             22 *
10 FS? 01        23 RCL Y
11 CHS           24 INT
12 STO IND 00    25 1 E1
13 RTN           26 /
                 27 RTN
```

Tenth Anniversary Calendar Announced

Here is your chance to acquire what could become a real collector's item. Although it is hard to believe that 10 years have already passed since Hewlett-Packard introduced the remarkable HP-35 Scientific Pocket Calculator, it is true. And to celebrate this anniversary, we are producing a **Hewlett-Packard Personal Computing Products Tenth Anniversary Calendar.** It will be a limited edition. Opened, the calendar measures 18 by 30 inches (45.7 by 76.2 cm), and it has a spiral binding.

Hewlett-Packard has commissioned **Michael Cacy**, a leading U.S. artist, to create 12 original scenes that depict the exciting events and places in which HP Personal Computing Products have been used. These scenes will be illustrated in color and in mixed media. Among the events selected for the calendar are: onboard the space shuttle *Columbia;* across the Atlantic Ocean in a balloon; in the pits at a Grand Prix race; winning the America's Cup race, *twice;* at the bottom of the Pacific Ocean; and seven more exciting events.

The photograph shows a "mock-up" of the calendar before it went into production. The final calendar will have the name of the month in seven languages, and appropriate text describing the depicted event will accompany the HP product shown for each month.

Each calendar will be packed in a special flat mailing envelope to protect it during transit. Delivery will start on or about November 1, 1981, and you should allow 4 to 6 weeks for delivery. The purchase price is $5* postpaid in the U.S., Alaska, and Hawaii; elsewhere, the price is $8.50* postpaid. To order, send your name, address, and payment to:

HP CALENDAR
Attn: Darlene Johnson
Hewlett-Packard Company
1000 N.E. Circle Blvd.
Corvallis, Oregon 97330 U.S.A.

This beautiful calendar is something you will not want to miss, so don't wait too long to order *your* copy. It definitely would be a very nice Christmas present, so we will make every possible effort to get all orders in the mail as soon as possible. All overseas orders will be sent by air mail.

** 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.*

Editorial

If you haven't deciphered the coding of V4N3p10b or other such abbreviations throughout this issue, here is the solution; *Volume 4 Number 3 page 10 center column.* The letter "b" at the end indicates the center column; then "a" indicates the left column and "c" indicates the right column. It is so much easier to use this coding when referencing other issues, plus it makes it easier to index things in KEY NOTES. Especially in the future.

For **Robert A. Green** (Mississippi) and many others who wrote, "What modifications were made to the HP-41's on the space shuttle *Columbia?*", here is an answer. The small plastic "feet" on the bottom cover were removed because some plastics slowly "out-gas" and that cannot be tolerated in the shuttle. Also, the battery charger access door was glued shut as a safeguard against any possible electrical spark. And the standard thin-plastic keyboard overlays were not used because of out-gassing possibilities. NASA used clear plastic overlays that were of a different material and which were slightly thicker. That's about it. Essentially, *your* HP-41's are identical to the calculators used by NASA onboard *Columbia.*

Norman C. Samish (Houston, Texas) asked a question that most Library members would also ask: "Can you order 'Special Programs' that cost $12 and pay for them with two program coupons?" Yes, you can, Mr. Samish. You can use program coupons for *anything* available *directly* from the Library. That does NOT include *Solutions Books* or other HP products or accessories. If you have a coupon worth four programs and you want just a "Special," we will return two coupons with your purchase. In other words, you can use your program coupons like "6-dollar bills."

Mr. Samish also pointed-out an error in V5N1p14c. In the "TITLE" routine, line 04 is FC?55; it should be FS?55. That's been in print over a year, and no one ever noticed it!

A letter from **C. Goldman** (London, England) informed us that there was another small error in V5N1p10c in the "CL" routine. Line 22 is REG IND T and it should be REG IND Z.

I thank both of you for taking the time to bring these errors to our attention. Often, readers notice and just fix the small errors. But it is frustrating to try a routine, find it doesn't work, and not know why. So we like to correct even the small errors.

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, 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.

By now you will have noticed that the time between V5N1 and V5N2 was considerably shorter than between past issues. Yet these two issues contained 16 pages. So you see, we are getting better in our effort to bring you lots of what you want and to get it to you on a reasonable schedule. And we have many more plans for making KEY NOTES better. However, if you don't subscribe (see page 15), you'll miss a lot of pleasant surprises in 1982 and beyond.

KEY NOTES Corrections

Thanks to letters from **William C. Tempelmeyer** of Evanston, Illinois, **Claus Primdahl Nielsen** of Bagsvaerd, Denmark, and **Hans Peter Brill** of Aachen, West Germany, you can make the following corrections to the flag table appearing in V4N3p5:

The status at turn-on for flag 37 and flag 40 should be coded M,3 rather than M,1, indicating that their status is maintained by Continuous Memory, and that "Master Clear" *sets* rather than clears these flags (display is FIX 4).

The status at turn-on for flag 47, flag 50, and flag 51 should be C rather than C,4, indicating that these flags are always clear

at turn-on but *not* that they always test clear. Flag 50 will test set in a program as long as a message resulting from a VIEW or AVIEW command is being displayed. Flag 47 will test set in a program after a PAUSE if the shift key is pressed during the PAUSE. Flag 51 will test clear in a running program, but it will test set in a program that is being single-stepped.

We also received this letter from **John L. Gafford**, of Waco, Texas.

A couple of observations about the HP-41C that may be of help to others:

First, regarding flags 22 and 23. Though implied in the instruction books and the article on flags in HP KEY NOTES V4N3, nowhere have I seen it explicitly stated that these flags, once set by a numerical (flag 22) or ALPHA (flag 23) entry, remain *set* until cleared by a specific clear command (CF, FS?C, FC?C) or until the calculator is turned off.

Second, when using AVIEW with the printer to print an ALPHA string, if the string is longer than 12 characters, program execution will be held up as the string scrolls through the ALPHA register. The PRA function in place of AVIEW will eliminate the delay, resulting in faster program execution.

I really enjoy KEY NOTES. Like a bottle of good wine, the longer it's around, the better it gets.

(I thank all of you for your sharp eyes. And special thanks are in order for that nice compliment. I know a red wine ages well; perhaps a "read" KEY NOTES also ages well?—Ed.)

KEY NOTES Subscription Plan

You were informed in the last issue that, because of skyrocketing inflation, we would soon be charging a subscription fee for HP KEY NOTES. This notice and a Subscription Order appear in this issue and will be in the next issue. The first issue in 1982, Volume 6 Number 1, will be mailed ONLY TO THOSE WHO SUBSCRIBE BEFORE THAT TIME. Here 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.

We'll bring you more details in the next issue.

* *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.*

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:

Getting Down to Earth With an HP-41 ...

Believe it or not, there are *many* people who *like* to jump out of airplanes and fall a great distance **before** deploying a parachute! While this premediatated act might not be everybody's cup of tea—so to speak—"skydiving" *is* an active and widespread sport. And you can bet that anyone who practices skydiving wants to enlist every facet of modern science to make certain (!) their fall is going to be an enjoyable experience. But what does all of this have to do with calculators?

Well, it so happens that skydivers' jumpsuits are almost as important as their parachutes. The "wings" on their suits are actually extra material between the underside of the sleeve and the body, plus extra material in the lower legs. (Take a good look at the photograph.) A suit not properly designed will not allow the wings to be self-inflating, and unwanted flapping will occur. Besides all this, suit design is important in determining the fall rate.

So along comes **Garry Carter** of Pope Valley, California, holder of three world championships and five world records in skydiving, and with him his friend **Paul Heckel**, a Systems Consultant in Los Altos, California. Do you begin to get the picture? Garry had a small business that made skydivers' jumpsuits. Paul owned an HP-41C. But to Garry, the HP-41C was only something he once saw in a magazine ad. And since he didn't have that kind of technical background, he certainly couldn't program the HP-41C.

Well, as you can imagine, Garry and Paul and the HP-41C got together, mainly because Garry couldn't afford a desktop computer even though he realized that the varied requirements for custom-made jumpsuits could be handled readily by a small computer. However, he also realized that, if he could program the HP-41C, he *could* actually have a small *handheld* computer to run his business.

In truth, it has worked out wonderfully well. Garry *did* learn to program and, using Paul's HP-41C for a start, developed an HP-41 program to calculate the critical dimensions of a jumpsuit so as to ensure that all jumpers in a team will fall at the same rate even though they differ in height and weight. This is very important in forming geometric formations that are made in the air during free fall.

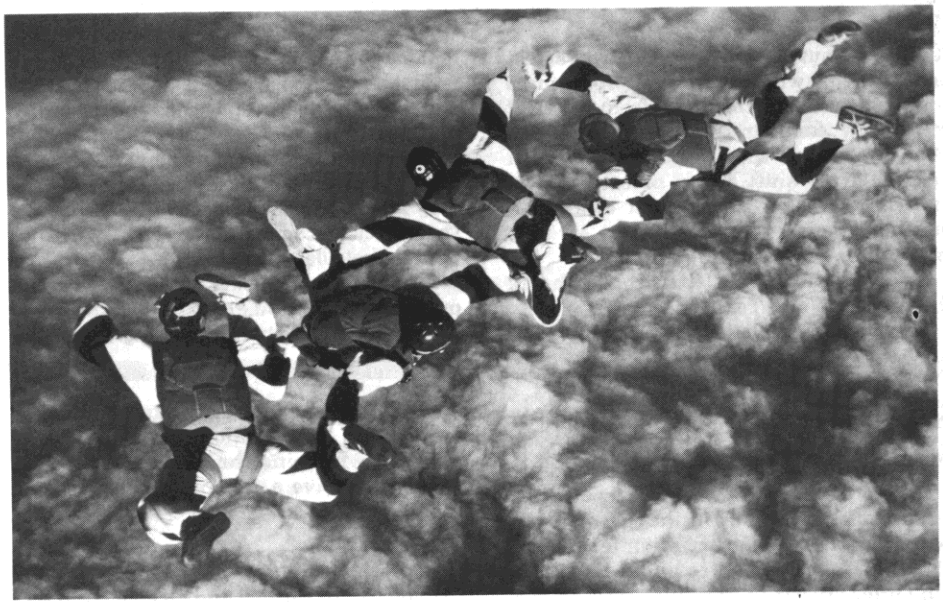
Of course, all of this happened some time ago. Today, Garry has his own HP-41C, card reader, and printer, and he has a thriving business. He even purchased an HP-41CV as a "portable computer" that can be taken out in the field to solve problems on the spot—or to make extra sales!

Garry recently came back from the U.S. Nationals (skydivers meet) and informed us that the U.S. Army "Golden Knights" skydiving team won first place in the competition. And you guessed it: they were wearing jumpsuits designed and manufactured by Garry Carter.

With the HP-41's, Garry now can easily custom-tailor a jumpsuit for anyone—even by mail-order. He can even design it to allow an "extra-slow" (!) fall rate of only (!) 92 miles per hour (148 km/h) or an "extra-fast" fall rate of up to 106 mph (170 km/h). All you have to do is furnish the parameters of height, weight, and gear weight and such measurements as waist, arm length, inseam length, and so forth. You can even specify colors and choose from various materials.

Today, Garry has a successful and larger business, and he *does* now own a computer system to run the business. But he still uses the HP-41 for suit measurements and still finds it an invaluable "field" computer.

The next time you watch some skydivers in flight, whether in person or on television, we'll bet that you think of Garry Carter and the HP-41's that made a good business profitable for him. And since you are reading this, *you* probably own an HP-41 and now know that just about *anything* can be done with that remarkable little machine. And if you still argue that it isn't a computer, maybe you should talk to Garry Carter.



Photograph courtesy of 4th Dimension. This team, informally known as 4D, consists of **Paul Stromberg**, **Jerry Swovelin**, **John Downing**, **Brian Johnson**, and **Bob Buehrer**. They placed third at the recent U.S. Nationals meet and set a new world record with Garry Carter's flight suits, which they are wearing in this photograph. There are five names above and only four skydivers in the photo? Well, that's because Bob Buehrer is taking the photo!

HP KEY NOTES

May-August 1981 Vol. 5 No. 2

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.*

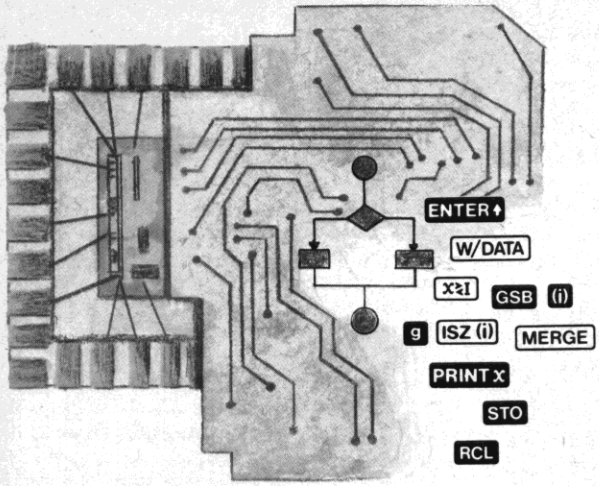
Hewlett-Packard Company
Users' Library
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 USA

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.

BULK RATE
U.S. POSTAGE
PAID
PERMIT NO. 814
PORTLAND, OR

Printed in U.S.A.



Featuring, this issue:

HEWLETT PACKARD

September—December 1981 Vol. 5 No. 3

HP Key Notes

Is this your last issue?? Better read page 15!!

New "Slim-Line" Calculators Introduced

On September 16, 1981, Hewlett-Packard released two new "slim-line" programmable calculators: the HP-11C Programmable Scientific Calculator and the HP-12C Programmable Financial Calculator. Both feature a liquid crystal display (LCD), Continuous Memory, a horizontal keyboard design, very low power requirements, and many built-in keyboard functions and programming tools.

The HP-11C Programmable Scientific Calculator has lots of memory—up to 203 program lines. And memory allocation is automatic! There are 5 user-definable keys; 5 single-character program labels; 10 numeric program labels; 4 levels of subroutines; 8 conditional tests; 2 flags; and 21 data storage registers (maximum). The HP-11C features a full set of math, engineering, and statistical functions; it also includes permutations, hyperbolics, and a random number generator. Editing capabilities include the ability to insert and delete individual lines of a program.

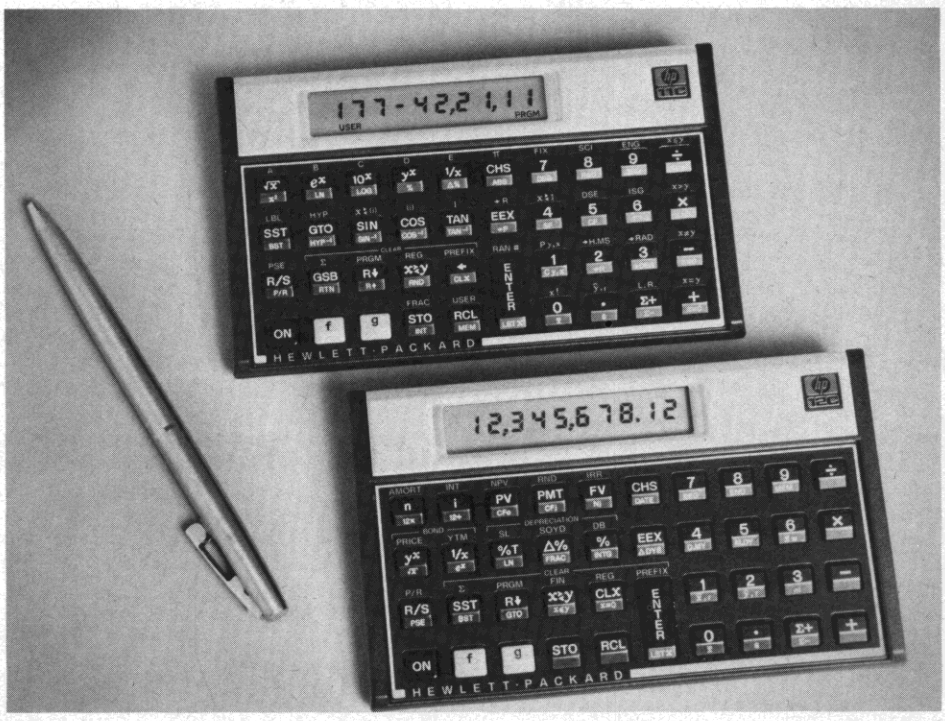
But there is much more than space will allow. And it's all in a handsome, yet very rugged, package you can carry in a shirt or coat pocket.

The HP-12C Programmable Financial Calculator has a maximum memory of 99 program lines and, like the HP-11C, has automatic memory allocation. There are 5 financial registers and a maximum of 20 data storage registers. It also has two conditional tests and unconditional branching.

The HP-12C features a full set of financial and business functions. Included are compound interest, amortization, discounted cash-flow analysis with net present value and internal rate of return, bonds and annuities calculations, three kinds of depreciation schedules, *and a feature that only HP offers*: odd-days interest, which lets you amortize, over the life of a loan, the interest accrued during the initial "odd days" period. Futhermore, this amazing pocket tool includes powerful statistical analysis and calender functions-all preprogrammed and available at the touch of a key.

Both new slim line calculators have another asset: an optional, accessory *Solutions Handbook* that has been made available to supplement the calculator owner's handbook by providing a variety of applications that would be useful in the "real" world. For the HP-11C this accessory book includes such applications as mathematics, statistics, electrical engineering, mechanical engineering, chemistry, economic analysis, surveying, and even games. For the HP-12C, this accessory book includes real estate,lending,savings, investment analysis, personal finance, Canadian mortgages, and some miscellaneous applications.

You have been asking for this type of product, and now it is here—*just in time for Christmas*. But a few words and one pictrue just can't begin to describe this new, powerful, rugged, handsome slim-line contribution to the art of programmable personal calculators. See your local HP dealer soon, before stocks are depleted. Even the price will be a pleasant surprise!



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

Library Corner

All of the programs highlighted in KEY NOTES are available worldwide. However, before you order any be sure to read the paragraph below: **"Ordering Programs."**

GOOD NEWS FROM YOUR LIBRARY

The Library, with the help of five additional part-time employees, has eliminated its backlog! Some of our customers, however, probably noticed a few deliveries still a bit tardy. That was not Murphy's Law prevailing. The culprit was a group of broken postage meter machines that delayed some shipments for five days. But those machines have been repaired and, along with our additional help, we are now meeting our goal of a two-day turnaround on orders and coupons.

For those orders that are needed immediately, make a direct call to the Users' Library, 503-757-2000, extension 3371 (NOT TOLL-FREE and ONLY for purchase-order and credit-card orders), and your order will be shipped that same day!! And if you are a member of the Users' Library, there is also still time to take advantage of the 25-percent discount on orders of six or more Users' Library programs (offer expires December 31, 1981).

Our heartiest congratulations to the first winners of the **Program Submittal Contest**! These winners and their programs are highlighted elsewhere in this issue. Don't miss these outstanding programs. And there will be more monthly winners, plus three talented authors who will win three **HP-85 Personal Computers**! You'll read all about it in the February 1982 issue—if you subscribe to KEY NOTES (see page 15). So don't miss this super opportunity to win some fabulous prizes ... submit your best programs to the Users' Library today.

Next year marks the **Tenth Anniversary** of Hewlett-Packard Personal Computing Products, and your Library has exciting plans for a year of celebration. A subscription has *always* been an excellent buy, and now it even includes a free 1982 subscription for KEY NOTES—a $5* value. And there's more! All subscribers will have a new option at their next renewal date: a $20* renewal that will include the coupon for four free programs, or the traditional $10* renewal, as always. Plus, a *Programmer's Reference Guide,* replacing the present *Contributor's Guide,* is under development and will be available with the January 1982 *Catalog of Contributed Programs.* This new guide will provide invaluable information for programmers of all experience levels. So if you are not a Library member, join now so you won't miss a beat of the excitement during our **Tenth Anniversary** year.

Finally, for those who have been eagerly anticipating the revision for program #00320C, Linear Programming Using the Simplex Algorithm, it is here! (See "New Programs.") Our thanks to **Laurence Esterhuizen**, who diligently revised his program while studying for his Masters degree in Johannesburg, South Africa.

The Users' Library is here as a service to HP-67/97/41 programmable calculator owners. We want to meet your needs, so let us know how we might serve you better.

SUBMITTING PROGRAMS

To maintain the high quality of the programs submitted to and accepted into 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 contributed program to the Library.

We also encourage you to always read the ongoing HP KEY NOTES column "In the Key of HP." This column addresses the things that we look for when we determine whether or not to accept a submitted program into the Users' Library.

Up to now, every program submitted to the Library had to include a magnetic card (or cards). We have a good reason for this; without a card or cards, it would take far too long to review and check all the many program submittals. Also, there is always an increased chance for errors when someone keys in handwritten keystrokes.

Since the advent of the HP 82153A Digital Wand for the HP-41, we can now accept HP-41 program submittals that have bar code instead of magnetic cards. However, **the bar code you submit with a program must be reproducible.**

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.

LIBRARY SUBSCRIPTIONS

In the United States and Canada, the fee for a one-year subscription to the Users' Library is $20.* If you live outside the U.S. and Canada, the fee is $30* because of considerably higher postage and handling charges. Upon becoming a member of the Users' Library, you will be placed on the mailing list to receive the *Catalog of Contributed Programs* and all of the updates. Plus, you will presently receive four coupons for free programs, each program valued at $6. It doesn't take an HP-41 to realize that this is a good deal! And, as an added incentive, if you join the Users' Library now, your initial membership will include a free subscription to HP KEY NOTES ($5 value) for the year of 1982, **no matter where you live in the world.**

Also, it will please you to know that once you are a member of the Users' Library, your succeeding renewal subscriptions in the U.S. and Canada are only $10,* and outside the U.S. and Canada they are only $15.*

With all the figures tallied, it's easy to see that you have everything to gain by becoming a member of the Users' Library.

IF YOU WERE A MEMBER...

You would be recieving a tremendous bargain from now until December 31, 1981. During that time, the Corvallis Users' Library is offering—**FOR MEMBERS ONLY**—a 25-percent discount on all orders for six or more programs. You would also get 48-hour turnaround time on orders received by mail or by our toll-free telephone number. And you would get same-day service on orders telephoned *directly* to the Library (503-757-2000, NOT TOLL FREE, and ask for extension 3371). Members of the Users' Library recently received the *Catalog Addendum* which, along with all of the most recent program submittals, contained (FREE) complete documentation and bar code for the HP-41 game program, "Reversi."

Plus, there are many more benefits of Library membership, whether in Corvallis or Geneva. You should reconsider this opportunity if you have turned it down in the past. The Corvallis Library, for example, now has the equipment, facilities, and a larger staff to handle all your varying needs. Plus, in the near future, the Library will offer even *more* services.

Think about it again. For the price of membership, you presently get four free programs of your choice. That, alone, is worth more than the membership fee. And in the future, we will be making more special offers *for members only.* But the greatest advantage of Library membership is being able to choose from a very large collection of software that is all ready to serve you in your application. The savings in time and effort are worth much more than the small membership charge.

ORDERING PROGRAMS

HP-67/97 and HP-41 programs mentioned 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 a prerecorded magnetic card or cards. Also, for HP-41 programs, this 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 or if you are ordering from Europe, South America, or Asia, 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.**

* *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.*

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).

Here's a helpful hint for customers outside the U.S. We have found that your orders are handled in a more efficient and timely manner if you will send, **attached to your order**, an International Money Order, a Foreign Draft, or the equivalent. *Any of these must be in U.S. dollars, drawn on a U.S. bank*, otherwise they will be returned to you, which involves a long delay for you. Much time is wasted and orders are held up in trying to match orders and checks that are sent in separately, or written on checks for non-U.S. banks and in foreign currency. Another option for you is to use such major credit cards as American Express, VISA, or MasterCard.

Usually, orders *not* delayed by the above problems will be shipped within 48 hours after they are recieved in Corvallis.

NEW PROGRAMS

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.

****

(67/97) ST36 A,B,C,or D GDR Comp-Girder Design (Composite or Noncomposite) (#04630D)

This program serves as an aid in the design or investigation of composite or noncomposite girders as a portion of a bridge design system covering highway and railway bridges.

The program is divided into two parts: design and investigation, either of which may be used independent of the other. Composite girders are designed, then confirmed by stress investigation, from initial assumptions for girder web and slab proportions. This requires analytical data on the distribution of moments at a section between initial dead load carried by the girder (unshared construction), on the superimposed dead load (considering creep of concrete), and on the live load applied to the composite section. (643 lines, 18 pages)

Author: **John A. Delong**
Cincinnati, Ohio

****

(41) Bearing Frequencies (#01095C)

This program calculates the vibration frequencies typically generated by rolling element bearings. It is applicable to Timken© bearing data as well as ball and needle bearings. Many of the calculated frequencies may not be observed in a vibration signature. The program is for a rotating shaft and stationary housing.

One reference used in developing this program was an ASME Paper (No. 79-Det-14) by **J.I. Taylor** titled, "Identification of Bearing Defects by Spectral Analysis." A copy of this paper is included in the documentation. *Required accessories: Two Memory Modules and Printer.* (752 bytes plus 27 data registers, 24 pages)

Author: **James Moorehead**
Pittsburgh, Pennsylvania

****

(41) Navpac for Yachtsman (#00982C)

A program designed for yachtsman, it combines the HP-41 NAVPAC Module, with its splendid features and long term almanac, into routines giving Line of Position (L.O.P), Most Probable Position (M.P.P), or longitude when latitude is known.

Also, with this program, you can calculate a fix from two L.O.P.s, the Time of Meridian Passage for any body, the time the body rises or sets and its azimuth at rise, plus longitude can be found at sunrise or sunset (including correction for atmospheric pressure and temperature). And the list goes on.

All formulas are included. *Required accessories: HP-41 NAVPAC Module, 3 Memory Modules.* (1018 bytes, 23 pages)

Author: **Louis Valier**
Honolulu, Hawaii

****

(41) Linear Programming Using the Simplex Algorithm (#00320C)

We featured this program in KEY NOTES some time ago (V4N2), but since then it has been completely revised by the author.

The program is used to solve classical, maximum or minumum problems in linear programming using the Simplex or Z-Phase approach. The user formulates the problem according to the inequality constraints and depending upon the objective function; the program does the rest. All of the rules and conventions are thoroughly explained in the documentation. It can handle up to 10 constraints and/or 10 variables.

The program performs pivotal transformations and outputs the optimum solution. The user options include a printout of the matrix after each iteration, editing capabilities, and automatic size check. In addition, the program operates in a "user-friendly" manner and it is quite fast for a program of this extent. *Required accessories: 2 Memory Modules (minimum).* (896 bytes, 19 pages)

Author: **Laurence A. Esterhuizen**
Johannesburg, South Africa

****

(41) Athletic Endurance Equation (#01023C)

The endurance equation is used to estimate the time required to run any distance from 1 to 26.2 miles, based upon known performance at a single distance. It is also used to compare an individual performance with "world class" standards. This program may also be used to compute the equivalent time requirements for swimming, bicycling, rollerskating, and race walking as alternative exercise. *Required accessories: 1 Memory Module.* (672 bytes, 8 pages)

Author: **S. Elwynn Taylor**
Ames, Iowa

****

(41) Step Test of Aerobic Capacity; U.S. Forest Service Method (#01024C)

This program determines the aerobic capacity for men and women of all ages according to age, weight, and pulse rate following a 5-minute "step" test. The program follows the method used by the U.S. Forest Service to measure physical fitness and to predict ability of men and women to sustain arduous work (such as fire line duty). The program also estimates the distance a subject is capable of "running" during a 15-minute period, based on the ability to take in, transport, and utilize oxygen. *Required accessories: 1 Memory Module.* (861 bytes, 12 pages)

Author: **S. Elwynn Taylor**
Ames, Iowa

****

SOME SPECIAL PROGRAMS

Occasionally, programs submitted to the Library are put in a category of "Special Program," by virtue of their length, their value in application, etc. The two programs that follow are such "special" programs. These programs are available from the Corvallis Library and carry the 10-percent handling charge to overseas locations. See "Ordering Programs" before you order.

****

(67/97) Curve Fitting (#67000-99955)

A beautiful compilation of curve fitting programs for the HP-67 and HP-97. This collection of programs is the answer to all of your curve fitting needs. There are 15 separate curve fitting programs included in this packet, all of them submitted by one author: **Ernest A. Taylor, Jr.**, of Decatur, Alabama. You can use this program and your HP-67 or HP-97 to fit experimental data to an optimum curve. Choose a linear fit, a parabolic fit an exponential, logarithmic, or hyperbolic fit. This special set of programs will fit data to any of these curves, and more, along either the X or Y axis.

$12* buys you this extensive set of curve fitting programs, including 75 pages of thorough documentation (5 pages on each program), and 15 magnetic cards. The value of having all of these curve fitting routines at your fingertips is immeasurable. Here's the abstract:

A set of 15 separate programs that can be used to fit any data to an appropriate curve.

(Continued)

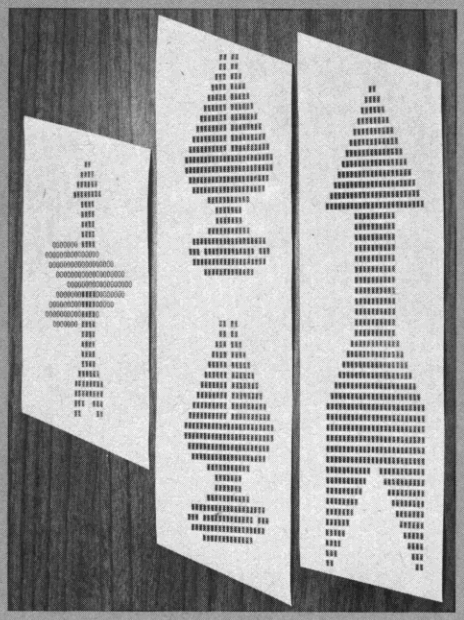
The set includes linear, exponential, logarithmic, parabolic, and hyperbolic fitting and variations of each. All of the equations used for the programming are included with the documentation along with complete instructions and example problems.

****

(41) Decorative Patterns and Letter Banner Printing (#67000-99956)

Is the nameplate on your desk, or on the door of your office lacking in originality? Take another look at it. Is it "run-of-the-mill," ho hum, out-and-out boring? Then you are in for a taste of spectacular innovation!

Imagine the thrill that your next visitors (expecting to see that same old nameplate) would experience if they were greeted by an HP-41 printout of your name on a large-size banner printout! And imagine the additional thrill if this printout was highlighted with all kinds of fabulous decorations. If the thought of all this gets you squirming with excitement, then this is the program for you.



The program is a work of art in itself. It allows your HP-41 and a printer to become a bonafide signmaker. It allows you to not only print any large size letter, A to Z, or any digit 0 to 9, but also patterns, characters, and decorations. The program fills the entire memory of your HP-41CV or HP-41 with a Quad Module, and it includes 73 pages of documentation and 20 magnetic cards, all for only $12*.

The author of this novel masterpiece is **C. Lamar Williams**, who lives in San Jose, California. Here is the abstract:

With this program you can print large size letters, A through Z, numbers 0 to 9, eight special symbols, and seventeen large size patterns of arbitrary length. Instructions are provided for the construction of a user's own patterns. Pattern capabilities include zig-zags, chain, sawtooth, happy or sad faces, hearts, arrow piercing a heart, and an arrow of optional length. Print any desired phrases. The printouts are ideal for party decorations, border decorations for posters, nameplates, and *lengthy* love letters.

First Library Contest Winners Announced

In the last issue (V5N2) we announced the Users' Library **Program Submittal Contest**. The contest rules and details are repeated elsewhere in this issue for anyone who missed them and would still like to participate. But hurry! The contest ends on December 31, 1981.

As you know, *three* programs were to be chosen on merit by September 30. That has been done, but there are *four* winners, as you will discover below. Notice that, because of their length and contribution, we have put all three winning programs in the "Special Program" category, and they are priced accordingly. Here, then, are the winning programs, in numerical order.

(41) Sound System Performance #67000-99957 (Price: $10*)

Employing readily available base data, this program computes the most relevant performance criteria for supplied equipment in a given acoustic space. A technique is used to apply the effects of occupancy required to render a marginally performing system effective in an otherwise unsuitable setting. The program is primarily useful to audio contractors and for commercial, institutional, industrial, or entertainers' sound systems. *Required accessories: three Memory Modules.* (595 lines, 1690 bytes, 34 pages)

Author: **Thomas G. Boulaine**
Buffalo, New York

(Congratulations, Mr. Boulaine! This is a most impressive program. Excellent description, complete flow diagram, relative equations, very useful sample problems and solutions, good references and drawings... I can see why your program won. Audio engineers should thank you for this tremendous tool of the trade—Ed.)

And who is **Thomas G. Boulaine**? He is a Sales Engineer for an audio contractor, and—believe it or not—this is his first "real" program. He's had an HP-41 for about a year and has written some very small "handy" programs, but this one started as a big task, and he ran it and worked on it and smoothed it until it accomplished what he desired of it. He has studied Engineering and Physics at the University of New York City at Buffalo, but has no degree—as yet. Mr. Boulaine is 40 years old, is married, and has one daughter. Since he already has an HP-41, he chose the printer as a prize.

(41) Superbeam #67000-99958 (Price: $10*)

This program calculates reactions, shear and bending moments at any point, locates points of zero shear, and plots shear and bending moment diagrams for any beam with two reactions, even—if desired—for a beam with cantilever spans at its ends. An addressing routine packs all load data into the lowest-numbered available registers, thus maximizing register usage. With three Memory Modules, up to 22 point loads or moments, or 11 trapezoidal or 15 uniform distributed loads or combinations may be accomodated by this program. *Required accessories: three Memory Modules and HP 82143A Printer.* (666 lines, 1603 Bytes, 29 pages)

Author: **Steven F. Dusterwald**
Las Vegas, Nevada

(Congratulations, Mr. Dusterwald! Anyone who analyses beams and knows about all those tedious shear computations will realize what a terrific contribution you have made to the art. A beautiful description, astonishingly complete flow charts, excellent sample problems, and solutions... no wonder you won. You told me during our short telephone interview that you considered this your pièce de résistance, *and I have to agree—Ed.)*

And who is **Steven F. Dusterwald**? He is a Structural Engineer for a structural engineering firm. He's written approximately 12 or 13 programs, all concerned with common structural problems. He is now bitten by the programming bug and thinks enthusiastically about authoring *more* programs. He also likes the plotting capability of the printer. Mr. Dusterwald has a Bachelor of Civil Engineering Degree from Cooper Union of New York City. He is 30 years old, is married, and has a 3-year-old son who uses Mr. Dusterwald's HP-33E (claims it's his, in fact!). And when asked if he thought his program would win, he replied: "Well, I made a real commitment, that's why I even typed the whole thing. Then I started to think it *was* a winner, but also thought about all those Ph.D.'s in mathematics, etc." Well, **now** you can believe you won, Mr. Dusterwald, as you use your brand-new HP-41CV prize!

(41) Acid-Based Factors for Blood and Brain Interstitial Fluid #67000-99959 (Price: $10*)

This program calculates pH, hydrogen ion concentration, carbon dioxide partial pressure, and bicarbonate concentration in arterial and venous blood, and in brain interstitial fluid for mammals, at body temperature of 37° Celsius. These calcula-

* *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.*

tions can be made in any order. Once data have been entered for any calculation, they need not be entered again for other calculations requiring the same information. *Required accessories: three Memory Modules.* (699 lines, 1645 bytes, 23 pages)
Authors: **Thomas Adams, Ph.D.**
S. Richard Heisey, D.Sc.
Michigan State University

(Congratulations, Doctors Adams and Heisey! This is a splendidly documented program, especially considering the complex subject. Very complete samples, all symbols defined, an excellent program summary, lots of flow diagrams...yessir, a winner for sure. Anyone interested in this field surely could use this fine program—Ed.)

And who are **Thomas Adams** and **S. Richard Heisey**? Well, first, they are the winners of a nice new HP-41CV. But primarily they are doctors who are in the Department of Physiology at Michigan State University in East Lansing, Michigan. They have collaborated on many programs and have submitted quite a few to the Users' Library. This winning program is the result of trying to find easy solutions to some awesome and time-consuming equations. (Refer also to V4N2p4b.) Dr Heisey's background includes Johns Hopkins University (Baltimore, Maryland) and Harvard University (Cambridge, Massachusetts). Dr. Adams attended the University of Washington in Seattle, Washington. Both are married, and both have two children.

In the February 1982 issue (V6N1) we will announce the three winners for October, for November, and for December, plus the three **Grand Prize** winners of three marvelous **HP-85 Personal Computers**. Be sure to either join the Users' Library or subscribe to KEY NOTES (see page 15) so you don't miss the next exciting issue.

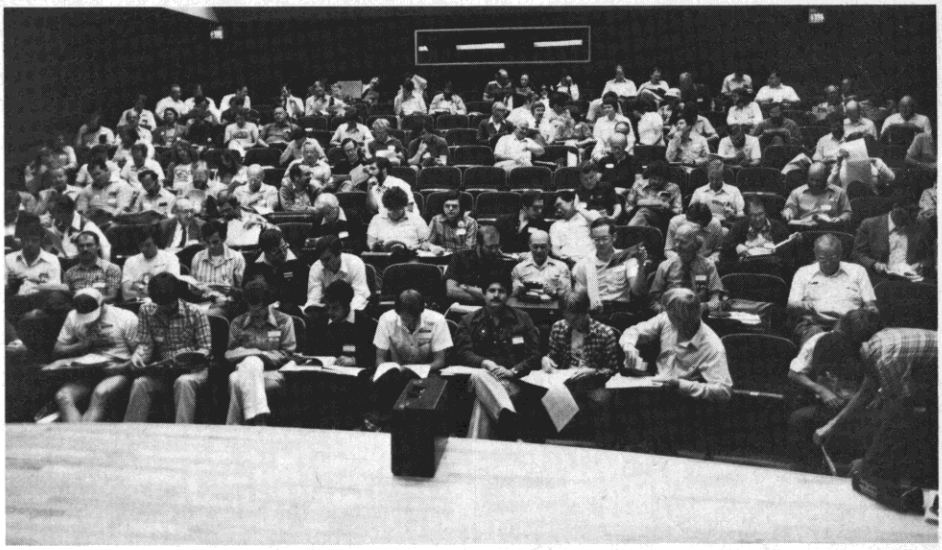


Photo courtesy of John Burkhardt (PPC).

PPC Conference Held Here

On August 21 and 22, the Columbia Chapter of PPC, the international calculator users club (page 10), sponsored the PPC Northwest Conference here in Corvallis, Oregon. For about 170 loyal PPC members, it was quite a momentous occasion, because they had the chance to fulfill a long-term dream: to see where their calculator was "born."

On Friday afternoon, August 21, we took 169 PPC members on tours through the Corvallis Division of Hewlett-Packard and then treated them to a picnic in our picnic grove just east of the lake on our property. It was an excellent chance for HP and PPC people to meet each other, and there is no doubt that both parties at the picnic had a very good time.

On Saturday, August 22, the formal PPC Northwest Conference took place at the beautiful Oregon State University Cultural and Conference Center here in Corvallis. Seventeen speakers, half from HP and half from PPC headed up sessions on everything from unique wand and printer applications to Synthetic Programming and KEY NOTES. At the end of the day, the PPC members had a rare chance to toss questions at some of the HP managers and personnel who made up a general panel. The photo shown here was taken from the stage during the Conference and shows most of the people who attended.

Naturally, there was a large contingent of members from the Pacific Northwest and from the Southern Calfironia area. But there were also members from Pennsylvania, Maryland, New York, Virginia, Illinois, and many other states. PPC members take these things very seriously—as you can see!

On behalf of Hewlett-Packard, your editor and his new assistant (**Ted Wadman**, Technical Editor), heartily thank all the PPC members who came to visit HP and who made the Conference a memorable occasion. We were delighted to meet all of you and look forward to the spring 1982 Conference which, rumor has it, will be in Philadelphia.

Toward More Secure (and Colorful) Bar Code

Standard bar code can be easily reproduced on any good photocopy machine. This is a real advantage if you want to cheaply duplicate copies for business associates. However, if the copies are of proprietary programs that you sell, then this ability becomes a major disadvantage.

Now there's a better way! Imagine black bar code printed on a bright red background. The wand "sees" no difference between red and white and happily reads the code. In contrast, the color combination drives most copiers "crazy."

How is this possible? The key is the frequency of the light emitted by the wand. It looks for maximum reflectance of light (approximately 700 nanometers) which it receives off of a white background. Of course, the right shade of red also provides maximum reflectance.

So far, HP knows of only one copier that does a good job of reproducing bar code that has a red background. Most copiers reproduce red very well and, therefore, have trouble providing any contrast between black and red. The end result is typically a muddy gray.

So far we have four shades of red that work. (You can even color-code your programs as long as you like red!) These colors, as specified by standard printers' inks, are:

1. PMS #199
2. PMS #485
3. PMS #185
4. PMS Warm Red

These inks are listed in order of worst reproduction (1) to best (4), but none have so far produced copy, on any copier that HP has available, good enough for the wand to read.

Any good printing company can reverse the process and end up with just black on white bar code. The use of the following statement will preclude that possibility.

> All rights reserved; not to be reproduced without the written permission of _____

Just put it on every page of bar code.

These are preliminary findings. The investigation continues, and you will receive more information in future issues of KEY NOTES.

If you have any questions or feedback, the person in the Corvallis Division to contact is: **Jack Peters** (503) 757-2000, ext. 2207 (not a toll-free call).

Library Contest Announced

Here are the rules and details of the popular Users' Library Contest. We are repeating this article, which appeared in V5N2, for the benefit of those who don't have the last issue. Also, those who read this article in the last issue, but didn't realize the potential benefits of participating in this contest, will read this article with a new interest. The winners for September have been announced and each has chosen and received a valuable prize. We have featured these winners, and their programs, in "Library Corner."

Contribute your best HP-67/97/41 program(s) to the Corvallis Users' Library before December 31, 1981, and you will become eligible to win some exceptional Hewlett-Packard products. Each month, for four months from September through December, three programs will be chosen on merit by a review committee. Prizes will be awarded as follows:

September 30, 1981—3 WINNERS!
October 30, 1981—3 WINNERS!
November 30, 1981—3 WINNERS!
December 31, 1981—3 WINNERS!

and each of these 12 winners will choose either an **HP-41CV Calculator** or an **HP 82143A Printer**.

On January 15, 1982, we will choose three winners for the **Grand Prizes**, and those three Grand Prizes certainly will be grand! Each of these three lucky contest winners will receive an **HP-85 Personal Computer**!

We know you will agree that this contest is worth entering. All the prizes will be awarded, and all winners will be notified by mail and announced in a future issue of KEY NOTES.

You do not need to be a Library member, and you may enter as often as you wish, but there are some rules you must follow or your submittal will not be accepted. The rules are:

- All programs must include at least the minimum documentation required by Hewlett-Packard standard Library submittal forms (and outlined in the *Contributor's Guide*). Forms from the Geneva Library will be accepted.
- All programs must be accompanied by either magnetic cards or bar code, and the bar code must be reproducible.
- All submitted programs must be in English.

Programs will be judged by the Corvallis Users' Library. They will be judged on:

1. Completeness of documentation.
2. Technical programming accuracy.
3. Usefulness (utility/technical capability).
4. Ease of use.

The decisions of the judges will be final. As programs are chosen as winners, we'll highlight them in KEY NOTES.

Here is your chance to put your calculator and your expertise to work. You have to admit that the prizes are enticing.

For submittal forms or for more information, contact the Corvallis HP-67/97/41 Users' Library at the address on the back cover. Or call them on (503) 757-2000, but remember that this is not a toll-free number.

In the Key of HP

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

In this issue, **John Loux,** a Technical Advisor in the Users' Library, clarifies some of the least understood HP-41 functions.

FOGGY FUNCTIONS

In the process of accumulating over 1000 HP-41 programs into the Users' Library, we have come to learn some things about how our authors perceive the programmable functions of the calculator. Certain standard (catalog 3) functions seem to be enshrouded in a cloud of mystery through which some users are unable to peer. To help alleviate this mystery, the following article has been compiled. In it, several potentially ambiguous functions are technically defined and some potential uses are highlighted.

Enter Function

Every Hewlett-Packard calculator owner knows that the function of the [ENTER↑] key is to load values into the stack. But in a program it is not always necessary to use [ENTER↑]. One obvious case is the use of [RCL]. The program lines: [RCL] 01; [RCL] 02; [RCL] 03; [RCL] 04; load the stack from T to X with the values from registers 01 to 04, respectively. Another, perhaps less obvious case, might be the program lines: 45; [RCL] 00. These lines load the number 45 into Y and the value from register 00 into X. The function **RCL** enables the stack lift so that the value 45 is pushed into the stack. Stack lift will be enabled when ANY function (except **ENTER**, **CL*x***, Σ+, Σ−) follows numeric entry either in or out of a running program. (Refer to appendix c in the owner's handbook.)

In a sequence of program lines such as: 45; ENTER; 3 E5; it is possible, but not desirable, to delete the ENTER statement. Many enthusiastic users have the impression that by deleting this statement they are conserving one byte of program space; actually, the calculator must maintain at least one byte between the two numbers in order to recognize them as separate entries. If the ENTER is deleted, an unpackable, null byte takes its place. No matter how the two numeric program lines are placed one after the other (switching in and out of ALPHA mode, etc.), they will always have an unpackable null byte between them. Because the space occupied by the null is the same as that occupied by ENTER (one byte), because execution of the null is not significantly faster than ENTER, and because the absence of ENTER causes confusion for inexperienced users, the Users' Library prefers that all submitted programs use a more standard method for loading the stack.

Often it is possible to eliminate the ENTER statement by rearranging program lines. For example, the sequence: SF 21; 1.4; ENTER; 25; can be shortened by one byte in this manner: 1.4; SF 21; 25.

Trigonometric Functions and Modes

The standard trigonometric functions **SIN, COS, TAN,** and **P-R,** and their inverses **ASIN, ACOS, ATAN,** and **R-P** require little explanation on their own but, when used programmatically, certain considerations must be taken into account. First, does the current trigonometric mode-setting matter to the proper functioning of the program? In many cases, it does not. All that is required is that the input values correspond to the current angular mode so that the output is correct. Or, in some cases it is possible for the program to alter its process based on the current trigonometric mode by testing flags 42 and 43, the gradians mode flag and radians mode flag, respectively. At other times, a program may require a certain mode in order to function properly. In such a case, the program should set the required mode. In any case, the user's options and programmatic mode changes should be well documented. As far as the Library is concerned, lack of documentation of this sort is undesirable in a submitted program.

Another consideration is execution time. Trigonometric functions are very time consuming, and their unnecessary use should be avoided. For example, if for some application the sine and cosine of an angle are both needed, the sequence [SIN], [LASTX], [COS] can be replaced by 1, [P-R], saving both time and space.

The final concern is, as always, space. In this example, 1 byte replaces 5: replace [x^2]; [$x \gtrless y$]; [x^2]; [+]; [$\sqrt{x}$] with [R-P] to find a vector magnitude. Even leaving the [$\sqrt{x}$] off of the first key sequence and adding it to the second results in a space savings.

Extended Precision

There are two functions in the HP-41 that allow extended precision in order to limit the calculator's inherent rounding error.

They are **E↑X−1** (e^x-1) and **LN1+X** (ln[1+x]). These functions are commonly used in formulas that model growth and decay, such as in compounding interest or radioactive decay problems. The functions allow the generation of a number that has greater precision than would the independent calculator keystrokes.

For example, if you key in the number 5.2345 E-05, execute [e^x], and then subtract 1, the calculator will return a result that is accurate to five significant figures (set display to SCI 9 to view your result). If you key in that same number and execute **E↑X−1**, the calculator will return a result that is accurate to 10 significant figures.

The extended precision functions give greater accuracy in the least significant digits because they eliminate the rounding caused by either adding or subtracting 1. The functions also save execution time and program space.

Modulo Function

Modulo (**MOD**) is often termed the "remainder" function because it effectively returns the remainder of the integer division of two numbers of the same sign. This does not mean that the two numbers must be integers or of the same sign before **MOD** can be used, but the significance of the result is slightly harder to understand.

Modulo is most useful in determining whether or not an input is within a given range or in modifying it so that it is. A number can be determined to be within a certain range if RANGE **MOD** X=0. If X is greater than RANGE, the value RANGE is returned. If X is less than -RANGE, the number returned will be a negative number less than -RANGE.

If a function works the same on integer multiples of a number as it does on the number itself (as is the case with the trigonometric functions sine, cosine, and tangent) but other functions in the program require numbers within a certain range, X mod RANGE will return a number within RANGE. For example: a hypothetical program requires the user to input the number of degrees an object has moved around a circular path. The input value is 2000 degrees (the object has moved several times around the circle). The routine: 2000; ENTER; 360; **MOD** returns the actual distance the object has moved, in degrees in the positive direction, away from its initial position; i.e., 200 degrees. In other words, the procedure will always return a number that is within the range 0 to 360 degrees, or one revolution about the circle.

In both of these applications, number pairs of opposite sign and non-integer numbers are valid inputs.

Sign Function

SIGN on the HP-41 has two major functions: (1) determining the mathematical sign of the number in X (+, −), and (2) detecting alphanumeric data. Determining the sign of a number with this function may not seem very useful (especially since we have the comparison functions [X<0?] and [X>0?]), and indeed would not be, if that were as far as it went. The usefulness of **SIGN** lies in its generation of the unit multipliers 1 and -1. With these results, the signs of other data can be altered without changing their magnitudes. One area of application is in navigation, where a series of coordinates may be entered and manipulated in such a fashion that their signs are lost. The original points may then be recalled, the sign taken, and the output corrected.

SIGN is also the only function in the HP-41 that allows you to test X for ALPHA data without generating an error, because **SIGN** operating on ALPHA data returns a zero. A simple ALPHA test might be: **SIGN**; [X=0?]; [GTO] "A LBL"; [CLx]; [LASTX]. With this test, only the LASTX register is affected if no ALPHA is detected.

Indices

Users' Library program authors have a good grasp of the use and usefulness of the functions **ISG** and **DSE** when they use them, except for two features. The first is the default increment value in the index number. Many authors waste two or more bytes of program space by keying in the cc portion of the number iiiii.fffcc when cc=01. These two digits may be left off because 01 is default. For example, to count upward from 6 to 100 by 1 (using **ISG**), the common index representation is 6.10001, while the preferred representation is 6.1 . This saves the program 5 bytes and performs the same function in the same amount of time.

The second problem comes when the author desires the incrementing or decrementing function without the "skip." This can be remedied by using cc [STO] [+] nn or something similar, but this disrupts the stack. An alternate method would be to follow the incrementing function with an inoperative function so that the skip, if performed, will not skip anything of importance. There are no truly inoperative functions in the HP-41, but there are a few non-destructive functions. Two commonly used functions are **X<>X** and **CLD**. While **X<>X** is relatively fast and causes no destruction of data, it consumes two bytes of program space. **CLD** is very fast and takes only one byte of RAM, but it functions to clear the display of any information left from a previous **VIEW** or **AVIEW**. The choice between these or other functions is, of course, up the author of the program, but since the function will be executed seldom if at all, the preferable alternative may be **CLD**.

Size Function

SIZE as a function is not difficult for the program author to understand. It is used to allocate data registers greater in number than the number of data registers the program uses. However, the use of **SIZE** is a bit more difficult for the program user. For instance, when the user has more than one program in RAM at one moment, for which program should the data register allocation be set? The one that uses the greatest number of data registers, of course. But this brings us to the real problem: authors don't always *exactly* indicate the **SIZE** requirement.

The minimum **SIZE** allocation for a program is critical in many instances, and is very difficult for the user to ascertain on his/her own initiative. When documenting a Users' Library program, there are three important places in the documentation where the required **SIZE** should be given.

(1) On the Status page.
The Status page is a short appendix of pertinent program information, and the minimum **SIZE** requirement figures prominently there.

(2) The User Instruction page.
The User Instruction page is a general but complete description of how the program can and should be used. Knowing the **SIZE** at this point can mean the difference between smoothly running the program and leafing through the rest of the documentation for the necessary information.

(3) The Example Problem.
Here, reference to **SIZE** is seldom found and it is an important place in which to see it. When trying to bake bread, it is very troublesome to have to look elsewhere than the recipe page to find the amount of flour needed. So it is with the **SIZE**. There is no better place to list it than on the page from which it will be implemented.

It is critically important to indicate whether the program has variable **SIZE** requirements. When the **SIZE** depends on the amount of data to be processed, a formula must be given to describe the dependence. Something on the order of "SIZE= 2*(the number of points) + 16" is adequate. Some example **SIZE**'s and their limitations should also be given.

Routines that determine the necessary **SIZE** and prompt you if the current **SIZE** is too small are well worth the program space they use. Such routines should not, however, be called as subroutines, because if **SIZE** is executed when one or more subroutines are pending return, the calculator will forget where to return to. The subroutine return stack is cleared when **SIZE** is executed.

When dealing with data registers, above all, do not assume that your program is the only one in calculator memory. Data registers are shared by all programs, and RAM space is limited, so use only the number of registers that your program absolutely needs; do not skip registers or indicate a minimum **SIZE** larger than that which your program absolutely needs.

Clearing Registers

It must be emphasized that data registers are community property. DO NOT EXECUTE **CLRG** UNLESS IT IS ABSOLUTELY NECESSARY. At the very least, give users the option to decide whether or not they want to clear them. However, you will find that most of your programs

(Continued)

initially store values in the registers they use, so having them clear is not a necessary prerequisite.

One favorable alternative to **CLRG** when blocks of data registers must be cleared is **CLΣ**. **CLΣ** clears a block of six registers at one time and may be positioned to any block of data registers from 00 to **SIZE-7** with the **ΣREG** function. Since the six register blocks can be overlapped and **ΣREG** can take an indirect parameter, any number from six to all of the allocated data registers can be cleared relatively conveniently. This process is surprisingly fast.

Displays

At this point in time, no set of HP-41 functions has been more misunderstood than the display functions. This group includes **AVIEW**, **CLD**, **PROMPT**, and **VIEW** and, of these, **AVIEW** and **VIEW** have been the most ambiguous.

PROMPT is relatively straightforward in its implementation. When the function is encountered in a running program, program execution halts and the contents of the ALPHA register are displayed. This is useful because it signals the user that input is required at a point. The resulting display has no effect on any user-accessable aspect of the calculator except for the functioning of the back-arrow key. When such a display is present, be it the result of **PROMPT**, **VIEW**, or **AVIEW**, pressing the back-arrow key does not clear the X-register, but rather it clears the display so that the X-register may be seen. All other functions will work properly, clearing the display as well when they are executed.

When encountered in a program, **PROMPT** will have one of two effects on a printer. If flag 21 (the printer enable flag) is set and the printer is in NORM mode, the display will be printed left-justified. If flag 21 is clear and/or the printer is in MAN mode, **PROMPT** will have no effect on the printer.

AVIEW and **VIEW** function identically except that **AVIEW** displays the contents of the ALPHA register while **VIEW** displays the contents of a specified data or stack register. These two functions work three different ways, based on the states of flags 21 (printer enable) and 55 (the printer existence flag).

When flag 55 is clear (i.e., a printer is not attached), and flag 21 is clear, and one of these two display functions is encountered in a running program, the display is filled with the contents of the specified register. Program execution does not halt, and the display will not be cleared until some disruptive display function is encountered or a new display is generated. In this mode three things can happen: status messages can be displayed without halting the program; displays may precede pauses (**PSE**) so that the duration of the display will be significant or data may be entered; and custom card reader input prompts may be generated. (Refer to your card-reader owner's handbook.)

When flag 55 is set and flag 21 is clear (i.e., the printer is disabled) and one of the two display functions is encountered again, the display is filled with the contents of the specified register, and program execution is *not* interrupted and the printer (being disabled) does *not* print. This mode is desirable for displaying messages that the programmer does not wish to be recorded.

When both flags 55 and 21 are set, and the printer is turned on, and one of the two display functions is encountered in a running program , then the display is filled with the contents of the specified register, and program execution is *not* interrupted and the displayed information is printed left justified. This is the ideal printer-oriented output mode. In this mode, long streams of data may be output to the printer for the user to review in his/her leisure. In this way, the user is not required to constantly monitor the calculator, and in exceptional cases where exceedingly long execution times are required by a program, the calculator and printer may be left for long periods (with the printer plugged in, of course). If the printer is turned off, **VIEW** and **AVIEW** perform as if flag 55 were clear.

If flag 55 is clear (printer is not present), flag 21 is set and one of the two display functions is encountered, the display is filled with the contents of the specified register, and program execution is halted. This is the mode encountered in a printer-oriented program when it is run without a printer attached. Program execution halts so that the displays that would normally be sent to the printer can be viewed and recorded by the user. In this mode, **AVIEW** works identically to **PROMPT**.

CLD functions to clear the display from a running program. The program annunciator ("flying-goose") is returned to the display.

The major problem with the use of these functions is the misconception that the status of flag 21 always matches the status of flag 55. THIS IS TRUE ONLY AT POWER ON. At any other time flag 21 may be cleared manually or by a running program. Because of this and because Users' Library programs may be run in a variety of configurations, flag 21 invariably must be manipulated or tested by the program. It is best that a program be flexible so that it can be run conveniently with or without a printer, but that is up to the author. The Library would prefer, however, that some attempt be made to control the output mode so that the program will run at least consistently, if not in the most convenient fashion, for each configuration.

Program Interrupts

There are several ways to stop a program for input, output, or simply to end execution, but several of them are ill-used and, evidently, "ill-understood."

PROMPT has been described earlier in the section on displays, so we shall not repeat the information here. **VIEW** and **AVIEW** also have been previously covered, but it is important for you to remember that they can be used to interrupt a running program. Because of their functional differences, these three display/interrupt functions can serve at least two different purposes. Since it is generally undesirable for input prompts to be printed, and it is always undesirable for input prompts to be printed when program execution halts so that the data may be entered, **PROMPT** is the obvious functional choice. In contrast, because of their flexibility in conjunction with the use of a printer, **VIEW** and **AVIEW** are ideal output functions.

STOP is a relatively easy-to-understand function; it does just what its name indicates: stops the program. But as an input/output function it leaves much to be desired. Because of its inability to generate a display other than the unlabeled X-register, it is a poor form of program interaction. The idea of preceding **STOP** with a display function may come to mind, but then **STOP** becomes useless (in most cases) because of the ability of all display functions to stop the program on their own. For this reason, **STOP** should be used predominantly in those situations where an unconditional halt is desired.

PSE is a function that gives the user time to do something. When **PSE** follows a non-interrupting display function, the information remains in the display with a pause in program execution of approximately one second. This is helpful if many display altering functions are to be executed in sequence so as to increase the time that each display is visible.

The pause function offers the added feature of allowing numeric input during the time that the program is interrupted. This feature is useful in speeding data input by reducing the necessary number of keystrokes. It also has the drawback of not allowing sufficient time to error-check the input value; therefore, the use of some other data-reviewing routine is almost essential.

RTN is most useful in signalling the end of a subroutine but can also be used as an interrupt. When **RTN** is encountered in a running program, one of two things will happen. If the portion of the program in which the **RTN** resides has been called via an **XEQ** command, the program pointer will return to the step immediately following that **XEQ**; otherwise, the program will halt in a manner identical to that of **STOP**. The functionality of the return command seems to be well understood by Users' Library program authors so we will not discuss it further.

END and **.END.** are identical in their functions and virtually identical in function to the **RTN** statement. Both **END**'s can be used to signal the end of a program, but **END** must be used on all but the last program in memory, and **.END.** can be used nowhere else. If either **END** is encountered in a running program, one of two things will happen. If the **END** was encountered as the terminator of a subroutine called by an **XEQ**, program execution will continue, starting with the state-

ment immediately following that **XEQ**. In all other cases, program execution will halt, and the program pointer will be positioned to step 00 of the executing program. These functions of **END** are particularly useful in that: (1) whole programs can be called as subroutines needing no special terminator; (2) the last subroutine in a program need not be terminated with a **RTN** before the **END**; and (3) if the program execution terminates with an **END** (a desirable occurence), the program may be restarted at line 01 by simply pressing [R/S].

A Users' Library program that turns itself off without user intervention can be unnerving at the least and destructive at the worst. **OFF** is not a common feature in contributed programs and it shouldn't be. Certain states are destroyed at power down (such things as the states of **ON** and flags 11 through 26 are altered). If you contribute a program that uses the function **OFF**, you must demonstrate in your documentation that there is no other, more convenient, method before we will accept your program into the Library.

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.

CALCULATOR TIPS & ROUTINES Especially for the HP-41C/41CV is a new book edited by **John S. Dearing** of Corvallis, Oregon. It contains a large collection of "tips" and "routines" and is thus more of a reference work than a text. However, how many times have you wished for all those routines and ideas from KEY NOTES, from HP handbooks, from programs, and from the PPC CALCULATOR JOURNAL* to be in one volume, and indexed so you could easily find them? Well, wish no more, because John Dearing has been busily adding to his collection of routines, etc., for several years and finally tackled the Herculean task of putting all of it in one book. Getting permission from all the various authors, alone, was quite a job. But the important thing is that he *did* do it, and so now all of us are the benefactors of his determination.

This book is written with the assumption that the user has already studied the owner's manual for his/her calcualtor and peripherals. It is not a "how-to-program" text; however, by carefully studying the routines, you should be able to improve your programming ability. Certainly the tips presented here will allow you to use your calculator more efficiently.

Some of the material in *Calculator Tips & Routines* includes "Synthetic Programming," but such routines are marked to prevent consternation to the beginner or to anyone not familiar with such material. And for those who don't know what we mean by Synthetic Programming, here's a definition: Synthetic Programming is the art of HP-41 programming using "new" HP-41 functions and text characters not described in the *HP-41C/41CV Owner's Handbook and Programming Guide.* The "new" functions are created in program memory, or are assigned to user keys, by "exotic" editing of standard functions. The techniques will work on any HP-41 and do not involve any modification of the calculator. (Refer to V4N3p8.)

This book also includes a Foreword by **Dr. William C. Wickes,** who wrote the book, *Synthetic Programming on the HP-41C.*** He discusses the applications of Synthetic Programming and the techniques of creating synthetic instructions.

The Introduction to this book was contributed by none other than **Richard J. Nelson,** the founder of PPC* and editor and publisher of the *PPC CALCULATOR JOURNAL.** Mr. Nelson writes about the advent of the "personal computer" and the problems that have evolved in getting information about programming to the user community.

And, finally, here is the Table of Contents:

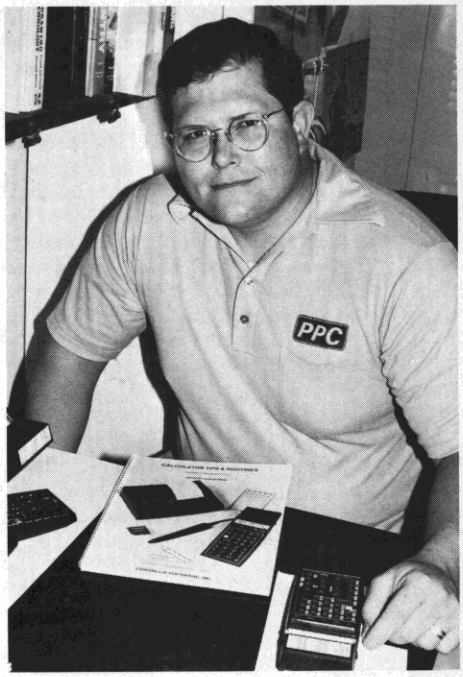
1. Basic Functions and Operations
2. Programming Tips
3. Initialization and Prompting
4. Display
5. Alpha Manipulations
6. Flags and Tones
7. Stack Operations
8. Memory and Curtain
9. Data Registers
10. Block Operations
11 Matrices and Data Processing
12. Sorting
13. Random Numbers
14. Fractions and Rounding
15. Arithmetic and Algebra
16. Geometry, Trigonometry, and Calculus
17. Base Conversions
18. Unit Conversions and Shortcuts
19. Statistics and Probability
20. Time and Date
21. Card Reader and Wand
22. Printer
23. Banners
24. Interchangeable Solutions
25. Synthetic Load Bytes
26. Reference

The book closes with a 5½-page Index plus an Author Index of those who contributed to this book.

There you have it: 136 pages; 8½ by 11-inch (21.6 by 28 cm) format, spiral bound; $15 postpaid to Canada, U.S.A., and Mexico; $20 airmail postpaid elsewhere. To prevent delay, make sure your payment is a check or money order drawn on a U.S. bank. Mail your orders to:

Corvallis Software, Inc.
P.O. Box 1412
Corvallis, Oregon 97339 U.S.A.

The author of this new book was born in Pomona, California, in 1947; graduated from Toledo High School in Toledo, Oregon, in 1965; and attended Oregon State



University in Corvallis, Oregon. In 1975 he married **Peggy Calkins,** and we suspect that, had it not been for her faith and perseverance, this book would not exist.

Mr. Dearing bought his first calculator, an HP-25C, in June 1977 (it's now used by Peggy). One of his first thoughts was, "Will I ever be able to use all 49 steps of program memory?" How things have changed since then! His next calcualtor was the HP-19C, and he discovered the value of having a printer.

Soon after this, Mr. Dearing joined PPC,* and a whole new world of calculator programming became visible to him. But translating HP-67/97 material to the HP-19C required careful study, so he started collecting and filing tips and routines for easy reference.

When the first shipment of HP-41C's arrived at the Portland State University Bookstore, Mr. Dearing couldn't resist, and he soon translated all his HP-19C material to the HP-41C. Again, after acquiring the card reader and printer, he realized the value of adding plotting and graphic functions to his earlier collection. And he also discovered KEY NOTES and more and more tips and routines.

In the spring of 1980, the Columbia Chapter of PPC* was organized in Portland, Oregon, and Mr. Dearing joined and met some local calculator enthusiasts. They, and others around the world, encouraged him to keep working on his collection of tips and routines. By early 1981, he decided to publish the collection and started to contact Hewlett-Packard and all the contributors. The huge mass of material slowly took form, and there you have it: the birth of a book.

Congratulations, John Dearing, and our special thanks to all the many people all over the world who made this book possible.

(Continued)

And if you're still looking for a Christmas present for a calculator enthusiast, maybe you just found it. It also may be tax-deductible!

**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 Calculator Journal *is a monthly publication published by PPC to disseminate user information related to applications, programs, hardware innovations, programming techniques, problems—any information related to the selection, care, use, and application of Hewlett-Packard personal programmable calculators. 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 × 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.

***The book* Synthetic Programming on the HP-41C *was reviewed in V4N3p8. Since that time, Dr. Wickes has moved to Oregon (he now works for Hewlett-Packard!), and the address for his book has changed to:*

Larken Publications
4517 NW Queens Ave.
Corvallis, Oregon 97330 U.S.A.

If you have used the old address, you will still get the book, but the new address will bring a faster response. Also, the new price for the book is $11.00 postpaid, by surface mail, anywhere. For airmail, add: U.S., Mexico, and Canada $1.00; for Europe and South America $2.00; for elsewhere $3.00.

Back Issues of KEY NOTES

Since KEY NOTES is soon going to be on a paid-subscription basis, many people have asked about whether or not we will make back issues available to those who do not have them. At the present time, we mail whatever is available, but supplies are at an all-time low ebb. So we are going to investigate the possibility of reprinting back issues, and we will give you a definite answer about that in the next issue (V6N1). It is certain that back issues will cost slightly more than issues run in mass printings, but we will do our best to make an equitable and reasonable price structure so that you can complete your collection of KEY NOTES.

Where Can I Buy Calculator Books?

That is a question that shows up in the mail a lot more often than we would like to see it. Usually, when a book is reviewed in KEY NOTES, it is a new book and still not readily available in local bookstores. Also, it takes a bit longer to distribute books to, say, Timbuktu than to New York City. So sometimes it is just a matter of patience. Our advice is to follow whatever information we print in KEY NOTES, then try your local bookstore or HP Dealer.

If those avenues prove hopeless, you might try the new service listed below. They have committed to carrying most of the calculator books that have appeared in KEY NOTES, and a lot more besides. They probably can even furnish books for HP Dealers. For example, they already stock the book reviwed in this issue. And they will mail their "Mail Store Book Catalog" of some 35 to 40 books to you if you request it. Their address is:

EduCALC Book Store
27963 Cabot Road
South Laguna, CA 92677

Software Additions and Corrections

HP-41 MATH PAC

Because of two small, rare occurrences, we have begun including an addendum card in the HP-41 Math Pac. Most of you will rarely, if ever, encounter these two cases, but just in case you do, you should add the following notes to your Math Pac handbook.

The "TRANS" program works very well when transforming coordinates from one system to a rotated or translated system. However, you must clear flag 01 *before* transforming points from the rotated or translated system to the original system. This applies to both two- and three-dimensional cases.

The "POLY" program might calculate erroneous roots for fourth- and fifth-order polynomial equations. Although this error will rarely, if ever, occur, you can use the program to evaluate the accuracy of a root. Here's how: After the roots are obtained, enter one real root and key-in [XEQ] [ALPHA] A [ALPHA] or press [A] in USER mode. If the value shown in the display is very close to zero, you have an accurate result.

A Word About the HP-41 Aviation Pac

One of our software engineers has discovered several constraints that can affect the "PLAN" program in our Aviation Pac (#00041-15018), and we think you should know about them. These constraints were not known when the Pac handbook was written and, therefore, are not pointed out in the text.

When using the "PLAN" program, always be sure to use at least two legs; one-leg flights will be computed incorrectly if there are winds aloft or if climb true air speed does not match the descent true air speed.

Also, be sure that the final descent takes place in only the final leg, with no part of the climb in that leg. If inadequate leg length or rate of descent parameters are specified, then mid-flight climbs or descents will be incorrectly computed.

Please remember that this program is for pre-flight planning on the ground, not in the air. Also, air conditions can change (such as winds), so use the program only as an aid and only in conjunction with your own computation and equipment.

If you have any questions or desire additional information on the Aviation Pac, please let us know. Address your questions to:

Hewlett-Packard Company
Customer Support
1000 N.E. Circle Boulevard
Corvallis, OR 97330

The DIGEST is back!

Have you missed the *HP Personal Calculator DIGEST*? Well it has been quite a while since we last produced an issue. But you will be happy to know that Volume 8 is scheduled for a December release, and you will be able to get one at the calculator sales counter of your nearest HP Dealer. **Also, all future issues will be available only through HP Dealers.** If you live outside the U.S. you can check with your nearest HP Sales Office for a dealer who carries the *DIGEST*; it will **NOT** be available from Corvallis.

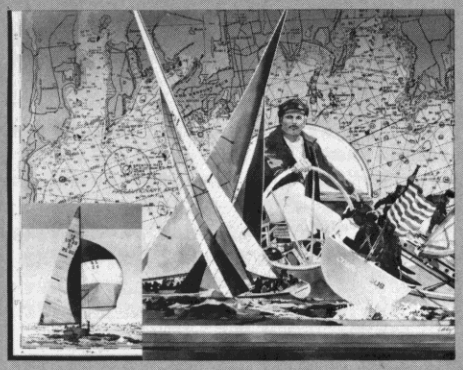
The new Volume 8 *DIGEST* contains feature articles on the HP-41C and the space shuttle *Columbia*, and an "inside" view of our new programmable slim-line calculators, the HP-11C and HP-12C.

Oh, by the way; it's now called the *HP Personal Computing DIGEST*, since the advent of the HP Series 80 Personal Computers.

Tenth Anniversary Calendar Ready

If you missed the announcement of our special **Hewlett-Packard Personal Computing Products Tenth Anniversary Calendar** in the last issue, NOW is the time to order what could easily become a real collector's item. It will be a limited edition. Opened, the calendar measures 18 by 30 inches (45.7 by 76.2 cm), and it has a spiral binding.

In the last issue, we printed a photo of the space shuttle scene, and here we have a photo depicting the "winning the America's Cup race" scene.



Below, we have reprinted the calendar-ordering details from the last KEY NOTES, in case you missed them. Since then, we've also learned that our HP Dealers will be carrying these special Tenth Anniversary calendars, probably with the dealer's imprint on them. So if you don't want to order from us, check one out at your local HP Dealer. We'll bet that once you see one, you'll surely want one.

Hewlett-Packard has commissioned **Michael Cacy**, a leading U.S. artist, to create 12 original scenes that depict the exciting events and places in which HP Personal Computing Products have been used. These scenes will be illustrated in color and in mixed media. Among the events selected for the calendar are: onboard the space shuttle *Columbia;* across the Atlantic Ocean in a balloon; in the pits at a Grand Prix race; winning the America's Cup race, *twice;* at the bottom of the Pacific Ocean; and seven more exciting events.

The final calendar will have the name of the month in seven languages, and appropriate text describing the depicted event will accompany the HP product shown for each month.

Each calendar will be packed in a special flat mailing envelope to protect it during transit. Delivery will start on or about November 1, 1981, and you should allow 4 to 6 weeks for delivery. The purchase price is $5* postpaid in the U.S., Alaska, and Hawaii; elsewhere, the price is $8.50* postpaid. To order, send your name, address, and payment to:

HP CALENDAR
Attn: Darlene Johnson
Hewlett-Packard Company
1000 N.E. Circle Blvd.
Corvallis, Oregon 97330 U.S.A.

This beautiful calendar is something you will not want to miss, so don't wait too long to order *your* copy. It definitely would be a very nice Christmas present, so we will make every possible effort to get all orders in the mail as soon as possible. All overseas orders will be sent by air mail.

** 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.*

Editorial

LAST...BUT NOT LEAST

The word "last" very seldom engenders good thoughts, unless...perhaps you just became a millionaire and it is your *last* day to work. Or, perhaps you just completed the last step in a spectacular calcualtor program that will make you rich and famous. However, loyal readers, the *last* to which I refer is that this is the last *free* issue of KEY NOTES. But don't let that fact cause you any dismay. KEY NOTES is still here, and that is what counts. It will get better, just as you have seen it progress in the past seven years. Just don't forget to read page 15 and then send us a check, join the Library, or inform your local HP Sales Office; then you won't have to worry about *last* issues, and you can think ahead about highly rewarding, informative *new* issues. I guarantee that you will remember V6N1 as a high-water mark in the mainstream of programmable calcualtors. And, significantly, it will be Hewlett-Packard's Tenth Anniversary in the calculator business.

OUR TENTH ANNIVERSARY

Speaking of tenth anniversaries...it is hard to believe that the remarkable HP-35 was released on January 4, 1972. It caused quite a sensational change in our "computational" lives just 10 short years ago. And you can believe that we are not going to allow this auspicious event to pass unnoticed. With such "firsts" as the HP-35 HP-80, HP-65, HP-41C, and...(fooled you!), we are going to build our celebration around "Ten Years of Innovation," and we'll tell you all about it in V6N1. We've even researched the very first sale of an HP-35 and have found both it and the original buyer. Both the owner and the HP-35 are somewhat older, but we're happy to report that *both* still work very well!

DON'T SEND $25!!

Some of you are mailing checks to us in the amount of $25, which represents a domestic one-year subscription to the Corvallis Library and a one-year subscription to KEY NOTES (see page 15). **Please do not send $25 checks!** *You get KEY NOTES free in 1982 if you join the Library;* thus, you should send only $20 for *both* services. Plus—and I almost hate to tell you—it costs more than $5 to return the overpayment to you. Please read page 15 before you mail a check, okay?

Also relevant to the subject of subscriptions is "addresses." We have always found that you get better and more positive mail delivery at your residence, not at your place of work. Of course it is up to you, but we'd prefer you use your home address on subscriptions. Also, be sure to notify us of any change of address.

**U.S. dollars. See note at bottom edge of cover.*

KEY NOTES FOR CHRISTMAS

Are you one of those people who seem to have everything? Does your family or a friend have trouble finding a Christmas present for you? Well, then, how about a KEY NOTES subscription, or a Library subscription, or even a "renewal" as an excellent Christmas gift? There is even the choice of quite a few price levels:

$5*—KEY NOTES Subscription (1 year)
$10*—U.S./Canada Renewal (1 year)
$15*—Overseas Renewal (1 year)
$20*—New U.S./Canada Library Subscription (1 year)
$30*—New Overseas Library Subscription (1 year)
(See page 15 for more details.)

Maybe you should leave KEY NOTES on the coffee or dinner table, open to this page—or page 15? Well, it's a thought, anyway ... plus—and this *could* be a *large* plus—KEY NOTES and Library subscriptions may be tax deductible on your Income Tax. It would pay you to look into that.

Also excellent to receive at Christmas are all those accessories you can add to your HP-67/97 or HP-41 systems. Your nearest HP Dealer should be well-stocked at this time of year, but don't wait too long!

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, 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.

BEST WISHES TO ALL

It is always a pleasure at this time of the year to thank all of you for your support of KEY NOTES, for your ardent participation, for your steady patience, and for all your nice comments about this newsletter. On behalf of Hewlett-Packard, I wish all of you all over the world a happy, safe, and joyous holiday season; a prosperous New Year; many happy hours of productive programming; and that you get a subscription to KEY NOTES for Christmas.

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.*

****

Some people *claim* that a strange creature resides in *Loch-Ness*, Ireland. Well, we *have confirmed* the existence of an HP-67 in *Nussloch*, Germany, and its owner is **Roland Waldi**. Mr. Waldi contributed this first routine.

(67/97) I'm going to introduce a German word, "Quersumme," simply because I don't know the English translation. Quersumme means "sum of the digits of a number," and most people will be familiar with it from rules such as: the Quersumme of every number divisible by 3 is divisible by 3 itself (with a zero remainder), and the equivalent rule for divisibility by 9. Here I want to present a small program to compute the Quersumme of a given integer.

```
001  *LBLA          011   LSTX
002     0           012    INT
003   STO1          013   X≠0?
004    R↓           014   GTO1
005  *LBL1          015   10ˣ
006     1           016   10ˣ
007     0           017   RCL1
008     ÷           018    x
009   FRC           019   RTN
010  ST+1
```

Now you can, for example, let the calculator find out what numbers are equal to the Quersumme of their square. The only solutions are 1 and 9. The only numbers with Quersumme (n † 3) = n are 1, 8, 17, 26, and 27.

****

Roland Waldi also has some neat hints for us to use on just about any HP scientific calculator. This is a convenient list to refer to when programming algebraic equations.

(67/97/41) Here are some special root functions of one argument.

Function	Keystrokes
$\sqrt{1+x^2}$	ENTER, 1, R-P
$\sqrt{1-x^2}$	SIN⁻¹, COS
$1/\sqrt{1+x^2}$	TAN⁻¹, COS
$x/\sqrt{1+x^2}$	TAN⁻¹, SIN
$x/\sqrt{1-x^2}$	SIN, TAN
$\sqrt{1-x^2}/x$	COS⁻¹, TAN
2 * log(x)	X†2, LOG
0.5 * log(x)	√x, LOG

These last two are similar for LN.

****

This next routine is a significant routine for those of you who do a lot of programming on the "Big" computers. **David Whyatt** sent it to us from Eastwood, South Australia.

(67/97) My HP-67 gets a fair bit of use preparing data for computer studies. This routine is added to all my data preparation programs to ensure that the answer is always in F5.X format (4 significant figures), be it positive or negative. Should the answer be too large to fit the F5 format, a row of zeros is displayed, while, if it is too small, the calculator displays it in SCI format.

```
001  *LBL9          020  *LBL8
002   STO9          021   RCL9
003    CF3          022    ABS
004     1           023   RCL8
005   STO8          024   X≤Y?
006     4           025    RTN
007   STOI          026   DSZI
008   RCL9          027    SPC
009   X>0?          028   DSPi
010   ISZI          029   RCL9
011   GSB8          030    SF3
012  *LBL7          031    RTN
013    F3?          032  *LBL6
014    RTN          033     1
015   DSZI          034     0
016   GTO6          035   STx8
017     0           036   GSB8
018   DSP9          037   GTO7
019    RTN          038    R/S
```

The routine can be modified to suit any other F format by changing step 006 (i.e., for F8.X use 7 in step 006).

One reason for sending this contribution is to see if any other reader can come up with a shorter or simpler routine. As most of the data I prepare is less than ABS(1.00) the program has been written to start with small figures and work up, thus saving time.

****

In the big sky country of Montana, right near the Continental Divide, lies the town of Anaconda. **John Craig** lives in this town and he owns an HP-67, on which, he developed this next routine.

(67/97) Twice now, I've seen, in KEY NOTES, routines that give y MOD x, but only for numbers that have the same sign. (GCD routine from **John S. Prigge** in V2N4 and a routine of mine in V3N1.) So, here's a routine for y MOD x that works for all cases (except x=0 of course). Thanks again for all the good ideas in KEY NOTES.

```
001  *LBLA          007   X<0?
002     ÷           008   ISZI
003   LSTX          009   X≠0?
004   X≷Y           010   X≷I
005   FRC           011    x
006   STOI          012    RTN
```

Daniel Q. Dye, Jr. of Denver, Colorado, contributed this next routine. It is a shorter version of a routine that we published a while ago.

(67/97) Oliver J. Olsen's register data scaling routine (V2N4p6) can be simplified by means of indirect register arithmetic. To use the improved routine, initialize the I-register with the highest register to be scaled and enter the scaling constant into the X-register.

```
001  *LBLB
002    ST÷i          005   ST÷0
003    DSZI          006    RTN
004    GTOB          007    R/S
```

****

Now, from Pittsburgh, Pennsylvania, comes a routine by **Vic Schmidt**. This is a useful routine for those of you who frequently encounter 3 × 3 space.

(67/97) Here is a fairly short routine for the HP-67/97 which multiplies two 3 × 3 matrices: R = A × B. The elements of matrix A are to be stored in primary registers 0–8 in increasing order (1,1; 1,2; 1,3; ... 3,3). The elements of matrix B are to be stored in the secondary registers (0–8) in the same manner. The resultant matrix, R, replaces B in the secondary registers, and A is preserved in the primary registers. Registers A and B are used for scratch, leaving registers 9, C, E, and D and secondary register 9 untouched. Run time is roughly 30 seconds.

```
001  *LBLE          029     0
002     7           030   GSBa
003   STOI          031   GSBa
004   GSBa          032   GSBa
005   GSBa          033    RTN
006   GSBa          034  *LBLd
007    RTN          035   STOi
008  *LBLa          036   DSZI
009     0           037   DSZI
010   GSBa          038   DSZI
011   STOA          039    R↓
012   GSBe          040    RTN
013   STOB          041  *LBLa
014   GSBe          042   X≷Y
015   GSBd          043   X≷I
016   RCLB          044   RCLi
017   GSBd          045   ISZI
018   RCLA          046   X≷Y
019   GSBd          047     3
020   ISZI          048     +
021    RTN          049   X≷I
022  *LBLe          050   X≷Y
023    R↓           051   RCLi
024   X≷I           052    x
025     9           053    R↑
026     -           054    +
027   X≷I           055    RTN
028  *LBLa
```

Only 55 program lines! There are three warnings, however. First, the subroutines must be placed in memory in the order shown,

since the LBLa is used three times (uses fewer labels this way). Second, note that the matrix is stored in registers 0–8 instead of 1–9 as in the Standard Pac matrix operations program (it turned out to be more efficient this way). Finally, there are three levels of subroutines used here, so the main routine (LBL E) cannot be called as a true subroutine. However, adding three more program lines and additional use of register 9 gives you a pseudosubroutine that is called by the sequence:

```
n
GTO E
LBL n
```

where n is any digit 0–9. The first seven lines of the multiplication routine are replaced by:

```
001  *LBLE
002   STO9      007   GSBa
003     7       008   RCL9
004   STOI      009   STOI
005   GSBa      010   GTO;
006   GSBa      011    R/S
```

and the now the pseudosubroutine may be called from up to ten different places in the master program.

****

With winter approaching, some people's thoughts turn to temperature. Since the United States continues to express temperature in Farenheit, while most of the world uses Celsius, the winter traveler can become frustrated. In Largo, Florida, **Alan Marcus** teamed up with his HP-41 to put an end to the frustrations of the temperature-conscious traveler.

(41) The following program displays a dual temperature conversion. The program, when executed, prompts for temperature to be converted and displays the dual answers—appropriately labeled. Additional data entries followed by R/S skip around the prompt.

```
01♦LBL "TEMP"      16 +
02 FS?C 22         17 STO 03
03 GTO 01          18 RCL 00
04 FIX 1           19 LASTX
05 "F"             20 -
06 ASTO 01         21 1.8
07 "C  "           22 /
08 ASTO 02         23 CLA
09 "TEMP?"         24 ARCL X
10 PROMPT          25 ARCL 02
11♦LBL 01          26 ARCL 03
12 STO 00          27 ARCL 01
13 1.8             28 CF 22
14 *               29 AVIEW
15 32              30 .END.
```

****

Now, **Julius Zechmeister** from Gopfritz, Austria has this time-saving hint for us.

(41) This is a trick to find local-ALPHA labels quicker in a long program.

```
      ⋮
193 LBL C
      ⋮
  PROMPT
      ⋮
226 LBL C
227 GTO C
      ⋮
  END
```

I often have to press XEQ C when the program stops at the PROMPT. By placing lines 226 and 227 fairly close below the prompt, I have decreased the time that the calculator takes to search for LBL C. After the first execution of the program, the GTO C at line 227 stores the jump distance to line 193 so the calculator does not have to search through each line of the program.

****

In the last issue of KEY NOTES (V5N2), we printed a "CUBE" routine that was similar to the other monadic functions in that it put x in LAST X, x↑3, in X, and preserved the rest of the stack. This is another "CUBE" routine that came to us from **Bob Flye** of Longview, Washington. It is not a monadic function but it preserves X, Y, Z, and T, and it demonstrates the advantages of the VIEW function.

```
01♦LBL "CUBE"      04 VIEW L
02 X↑2             05 SQRT
03 ST* L           06 .END.
```

****

Next, we'll travel to Stockholm, the capital of Sweden, where we get a seasonal idea from **Jens Sucksdorff**.

(41) When dealing with cyclical data series like seasonal data, it is very useful to see in what season a data entry occurs. To do so one very useful function is MOD, but as this one gives 0 as an answer each time the period is divisible by the number of seasons, it is necessary to use one of the following routines:

```
01 MOD          or    01 MOD
02 X=0?               02 X=0?
03 X<>L               03 RCL L
```

depending on whether the stack or the LAST X-register is to be maintained. To use these routines, enter the period in Y and the number of seasons in X. If the number of seasons is 4, an example output is:

Period No.	Season
1	1
2	2
3	3
4	4
5	1
6	2
:	:

****

We know there is at least one person in Borgheim, Norway, who makes efficient use of the stack in his HP-41. **Brent Tranberg** sent us this short, fast routine as an example of his skill.

(41) A very short program that solves for two real roots of a quadratic equation:

```
01♦LBL "SOLVE"
02 X<> Z           09 X↑2
03 ST/ Z           10 R↑
04 /               11 -
05 -2              12 SQRT
06 /               13 ST- Z
07 ENTER↑          14 +
08 ENTER↑          15 .END.
```

To use this routine, key in [A], [ENTER], [B], [ENTER], [C], [XEQ] [ALPHA] SOLVE [ALPHA]. The display will then contain x1. Press [X≷Y] to see x2.

(With a test and branch after line 11 and about 10 more bytes of programming—depending on the ALPHA messages—this routine also could solve for imaginary roots—Ed.)

****

Chadron, Nebraska, is the home of **Warren R. Block**, who sent in this cute and useful routine. Mr. Block recommends the routine for game programs, but it would serve to spice up any program.

(41) Here is a subroutine that is useful in game programs. It uses the DSE function to flash an ALPHA message on the display.

```
01♦LBL "FLASH"
02 CF 21           07 AVIEW
03 10              08 DSE 00
04 STO 00          09 GTO 00
05♦LBL 00          10 RTN
06 "***FLASH****"  11 .END.
```

Line 03 is the counter value and line 04 is the counter register. A tone instruction after line 07 causes a beeping sound, but slows the routine considerably.

****

It isn't often that we see an 8-byte program cut down to 3 bytes, but everything is possible. In the last issue (V5N2), we printed a %T routine (the percent that x is of y) that was short and sweet; excluding the LBL and END it was 8 bytes. Anyway, **Greg Smith** of Phoenix, Arizona, sent us his version, which lops 5 bytes off of the previous version.

(41) This is in regard to the %T routine submitted by **Mr. Heidecker** (V5N2p10c). I agree wholeheartedly that %T is a desirable function and know of a shorter way to implement it. I came across the method some time ago and cannot now remember where. The routine is as follows:

```
01♦LBL "%T"
02 1/X
03 %
04 1/X
05 .END.
```

(Continued)

The core (lines 02-04) will run as a program or can be used manually on many calculators.

(A lot of people submitted their versions of this routine. This one was the shortest and quickest. However, we appreciate the variety and thank all who contributed—Ed.)

****

Our next stop is S'Gravenwezel, Belgium, where we find that **Vally Lambrechts** is making good use of his HP-41 and card reader.

(41 with card reader) I have written a small routine to call a subroutine even when it is not in the calculator's memory. The program's name has to be in ALPHA. If the required program is in the calculator's memory it is simply executed, but if it is not, the calculator will ask for the card with the program. Then it checks to see if the correct program has been inserted and, if not, it will ask again. This routine has to be packed and have an END or it will erase itself. The last limit is that the ALPHA label may only consist of a maximum of six letters instead of seven. This routine uses about 36 bytes, and it preserves X, Y, Z, and T, and it clears flag 25.

```
01♦LBL "GOSUB"
02 ASTO L          08 "INSERT"
03♦LBL 01          09 ARCL L
04 SF 25           10 AVIEW
05 XEQ IND L       11 XROM 30,04
06 FS?C 25         12 GTO 01
07 RTN             13 .END.
```

(This is a valuable application for the HP-41 and card reader combination. It is a good basic program to help manage your "files," or card programs. Bigger computer systems have extensive file management programs. As you develop more and more routines along these lines, the HP-41 steps closer and closer to "the big league" systems—Ed.)

****

A few issues back we declared a moratorium on MOP routines and clear-register routines. Nevertheless, we still receive—daily—a pile of letters containing routines titled MOPX, CLRGX...etc. All we can say is that we love to see them but chances are slim that they'll be published. **Jim Balisalisa** from Riyadh, Saudi Arabia, sent us this MOP routine. The logic is clean, and Y, Z, T, and L are preserved.

(41) I have enjoyed reading KEY NOTES, especially "Routines,xxx." For clearing registers within two limits, has anybody tried the following?

```
01♦LBL "MOP"
02♦LBL 01          05 ISG X
03 STO IND X       06 GTO 01
04 ST- IND X       07 .END.
```

The limits of registers to be cleared are keyed into X in the ISG/DSE format (i.e., iii.fffcc) and follows its rules (see owner's handbook; pg. 163). The program can clear the registers within the specified limits either one after the other or every other, every third, etc., depending on "cc."

****

Here's an input from **Scott Newell** of Fort Smith, Arkansas. It is a shorter version of a routine that we printed in a previous issue. By the way, while you are keying in this program, keep in mind that Mr. Newell is 9 years old.

(41) IN KEY NOTES dated November 1979 (V3N4), **Craig Pearce's** program named VEIWSZE uses 15 lines but mine uses 13 lines. I enjoy KEY NOTES very much.

```
01♦LBL "SEESZE"
02 SF 25           08 FS? 25
03 0.31601         09 GTO 00
04 STO X           10 "SIZE="
05♦LBL 00          11 ARCL X
06 ISG X           12 AVIEW
07 VIEW IND X      13 .END.
```

(Thanks for the nice compliment Scott. We can't help but respect a contribution like this from a young man like yourself—Ed.)

****

Next, we have an interesting application that was submitted by **Fred Scheifele** of Delran, New Jersey.

(41) On the lighter side of programming, I have come up with a crude random word generator. The program is crude—not the words! However, it is based on the natural frequency of occurence of each letter of the alphabet. The simple random number used (LBL 00) was supplied in the applications programs book with my first programmable calculator—an HP-25. Some data must be pre-stored in the data registers. A seed for the random number generator must be in register 28. The cumulative frequencies for the letters A to Z must be stored in registers 02-27 (see list below). The 26 letters must be alpha-numerically stored in registers 29-54.

```
01♦LBL "RW"
02 "LGTH?"         21♦LBL 05
03 PROMPT          22 RCL 01
04 STO 00          23 28
05 CLA             24 +
06♦LBL 01          25 ARCL IND X
07 1.025           26 DSE 00
08 STO 01          27 GTO 01
09 XEQ 00          28 AVIEW
10♦LBL 02          29 STOP
11 RCL 28          30 GTO "RW"
12 RCL 01          31♦LBL 00
13 1               32 RCL 28
14 +               33 PI
15 RDN             34 +
16 RCL IND T       35 5
17 X>Y?            36 Y↑X
18 GTO 05          37 FRC
19 ISG 01          38 STO 28
20 GTO 02          39 .END.
```

Cumulative Frequencies for use with Random Word Generator above.

REGISTERS	VALUE
02	0.0774
03	0.0915
04	0.1237
05	0.1631
06	0.2916
07	0.3191
08	0.3367
09	0.3828
10	0.4526
11	0.4542
12	0.4582
13	0.4946
14	0.5215
15	0.5938
16	0.6711
17	0.6951
18	0.6967
19	0.7690
20	0.8319
21	0.9259
22	0.9539
23	0.9639
24	0.9779
25	0.9817
26	0.9990
27	1.0000

****

Fred Lipshultz is a physicist who lives in Storrs, Connecticut. Along with his interest in Physics, **Fred Lipshultz** is intrigued with the MOD function on the HP-41.

(41) I have developed a routine that converts a number in Base 10 to any Base n, where n can be an integer 2, 3, ... 9, or 10 itself! This is the routine; "GEN2" (for general base conversion 2). I expect the recursion relations used in this routine are known by mathematicians but this is the first time I have seen them.

```
01♦LBL "GEN2"      22 X=0?
02 "BASE?"         23 GTO 02
03 PROMPT          24 STO Y
04 STO 04          25 RCL 02
05 10              26 /
06 -               27 RCL 03
07 CHS             28 *
08 STO 05          29 RCL 05
09♦LBL A           30 *
10 RCL 04          31 ST+ 01
11 STO 02          32 RCL 04
12 1               33 ST* 02
13 STO 03          34 CLX
14 "↑NO"           35 10
15 PROMPT          36 ST* 03
16 STO 01          37 RDN
17♦LBL 01          38 RDN
18 RCL X           39 GTO 01
19 RCL 02          40♦LBL 02
20 MOD             41 VIEW 01
21 -               42 .END.
```

I think it is safe to say that the MOD function is one of the most useful functions on the HP-41, and that the above routine would not be possible without it.

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.)

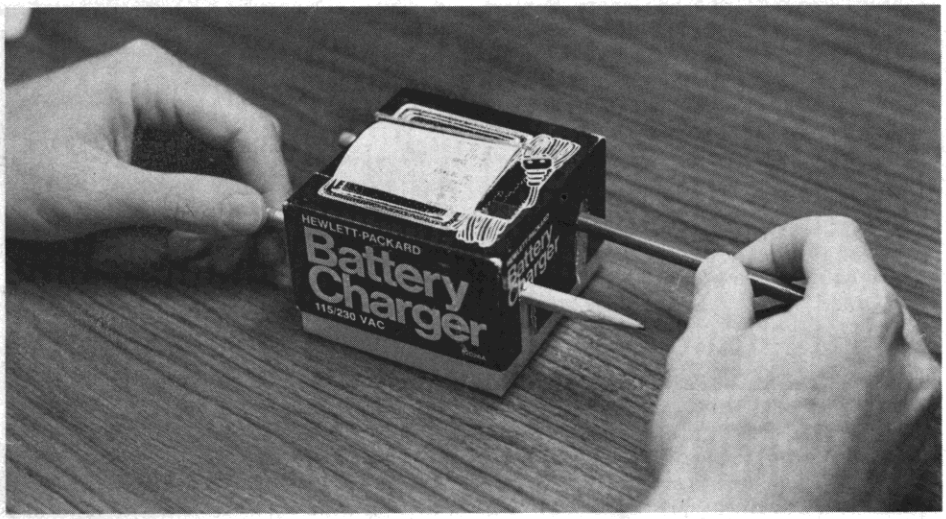
Your name, complete address, and postal code:

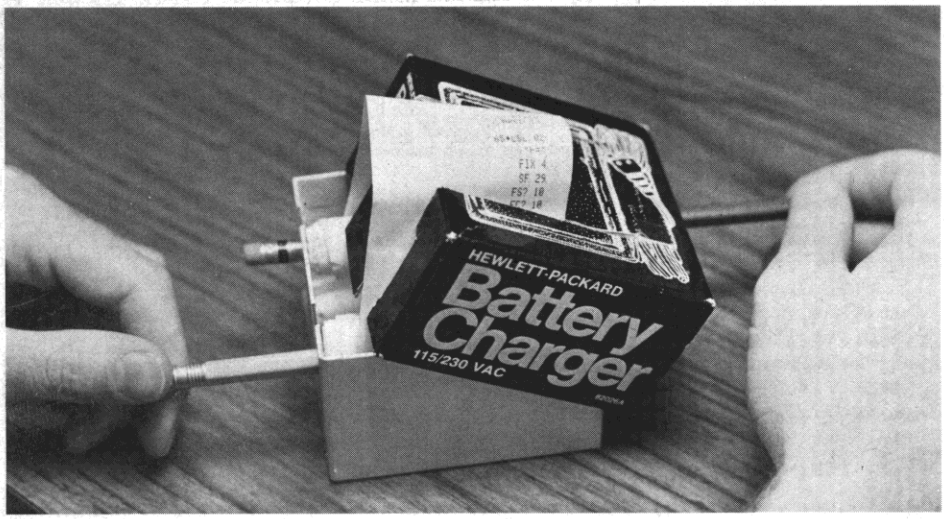
Those Inventive Calculator Users...

Calculator users tend to be an inventive bunch of people. Give them a challenge and they'll respond with a new idea. Here are some that we thought you would enjoy—maybe even use yourself.

ROLL YOUR OWN

Anyone who uses a calculator printer and creates a long, long printout knows that it isn't the easiest thing in the world to handle. Take **Tom Harper**, for example; Tom works in Customer Support (at Corvallis Division) and often has to juggle the telephone, a long printout, and maybe even a pen or pencil. As you can see in the photos, his inventive mind, two pencils, and the cardboard box for one of our battery rechargers solved one problem for him. Crude as it may look, it works—and works well.



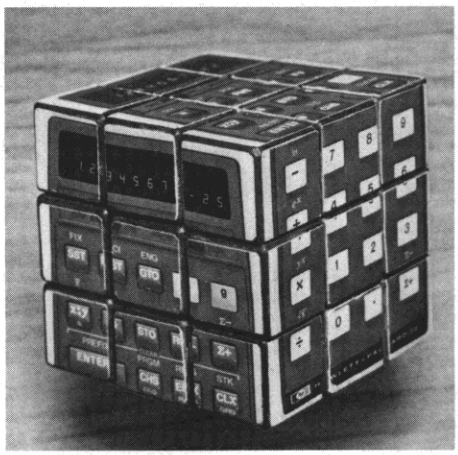


KOLB'S KUBE

Anyone can own a multi-colored Rubik's cube, which is a new fad that's sweeping around the entire world. But, as far as we know, there is only *one* "Kolb's Kube," and it resides in a desk here at Corvallis Division. That desk belongs to **Dr. William C. Wickes**, author of *Synthetic Programming on the HP-41C,* and formerly a member of the Washington (DC) Chapter of PPC. **Bill Kolb**, also a member of that Chapter, made the cube as a going-away present for Dr. Wickes when he joined Hewlett-Packard.

You cannot imagine the work that went into this project. First, Bill Kolb had to dig through his collection of old HP sales literature to find three calculator photographs that would fit onto the surfaces of a standard Rubik's cube. He finally found photos of the HP-21, HP-22, and HP-25 that were enough alike to fit, and then he had to carefully cut and fit pieces to fit the individual cube surfaces, to cover the multi-colored original surfaces.

It is truly a work of art and a splendid keepsake. It wasn't easy to photograph, but we assure you that it is not easy to solve as a puzzle, either! First, you have to remember what each of those three calculators looks like, and there are no colors to help you. It is utterly fascinating.



HP KEY NOTES

September—December 1981 Vol. 5 No. 3

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.

0001298918
M DEZEARN
EXXON RESEARCH & ENGINEERING
BOX 4255
BAYTOWN TX 77520