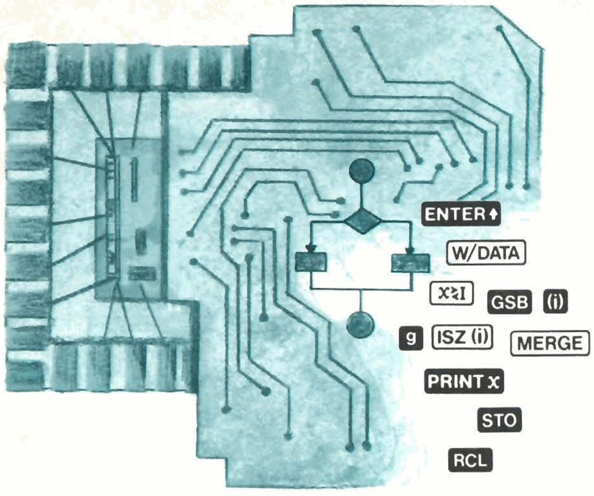


Featuring, this issue:

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.

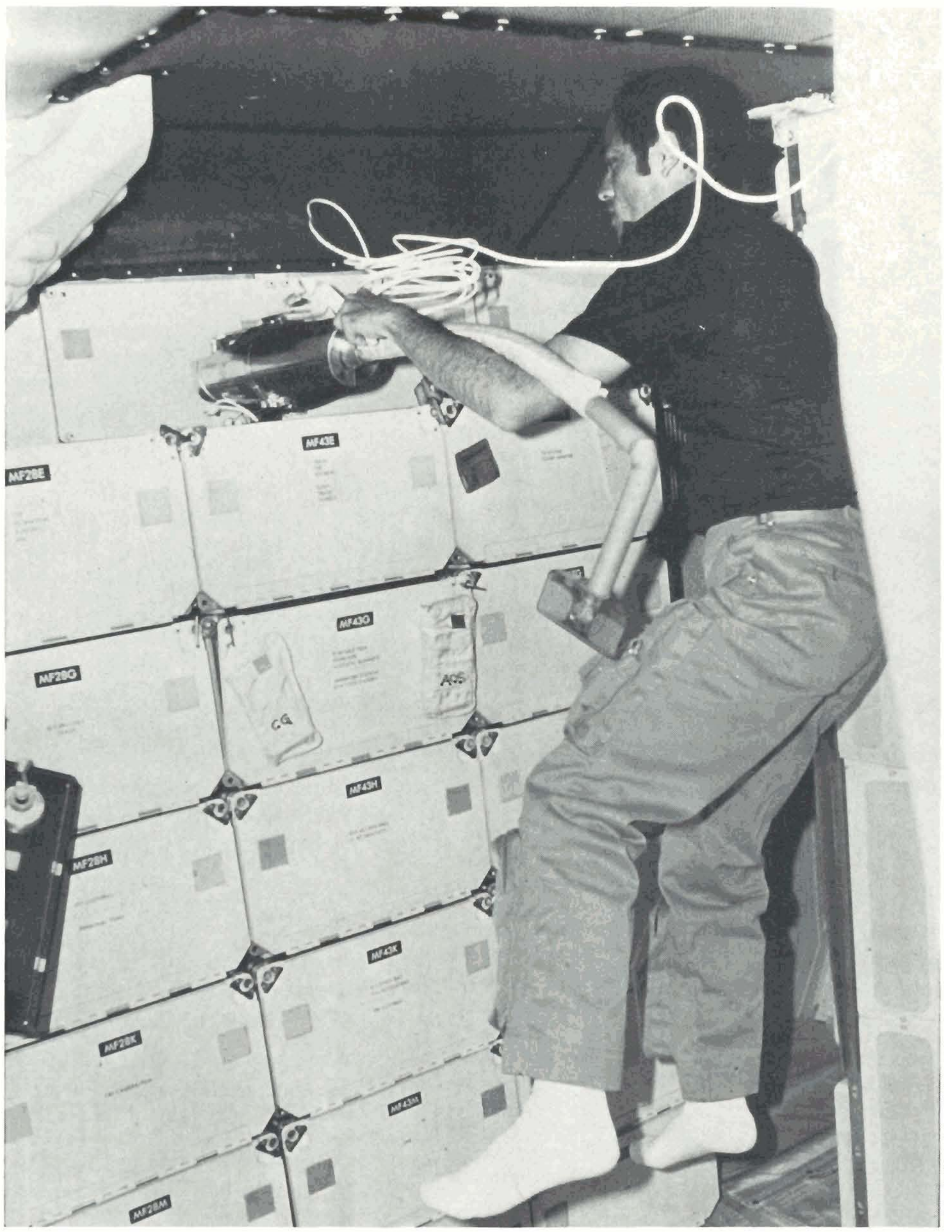
In 1980, when the National Aeronautics and Space Administration (NASA) of the U.S. began studying available calculators for use on space shuttle flights, they soon realized that the most important initial factor was large memory, an absolute necessity in order to accommodate the lengthy programs that 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

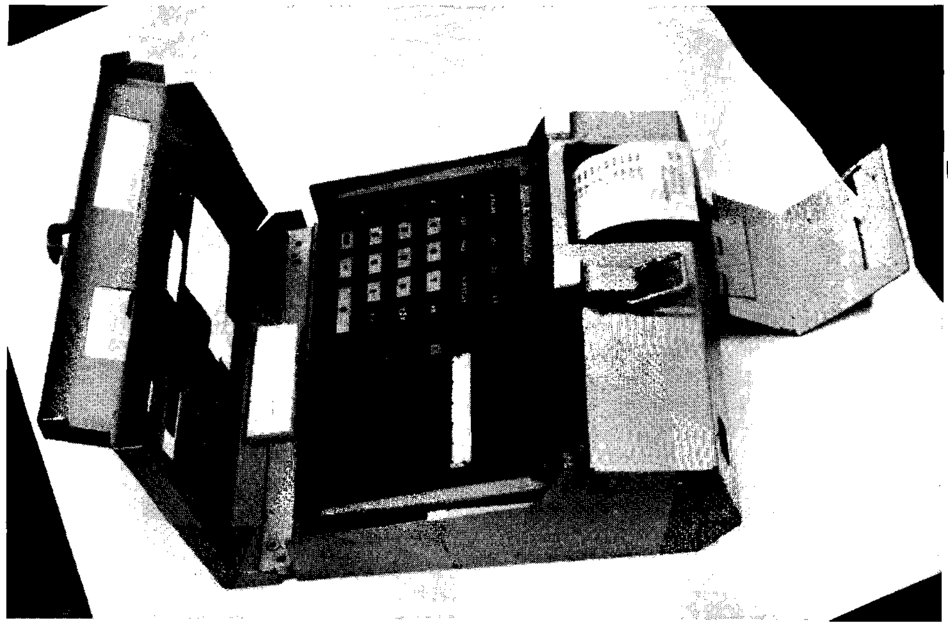
No matter where you live in the world, you know that skyrocketing inflation has caused severe increases in the costs of labor, materials, postage, and freight. And since KEY NOTES has more than doubled its mailing list in just the last year, we have to make a decision to either discontinue the newsletter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

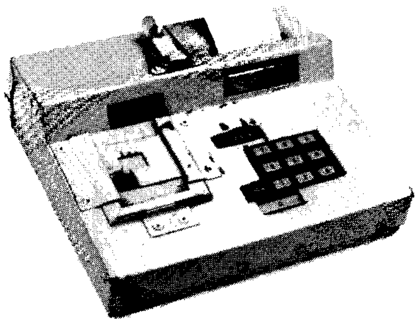
So please be patient for a while longer. We promise that you will not regret the wait.

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.

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.

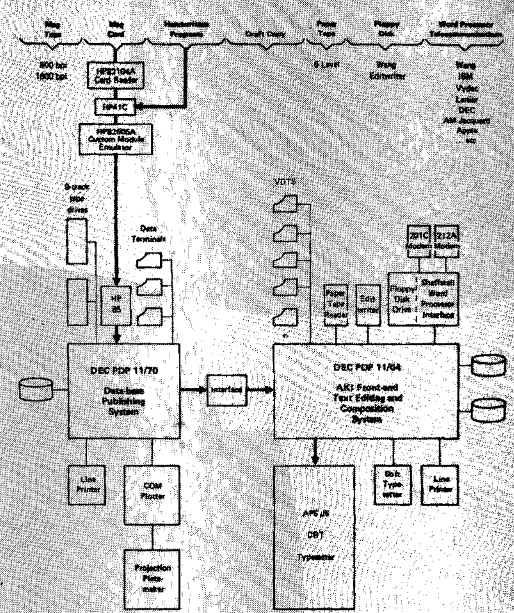
ROW 1 (1 – 5)
ROW 2 (6 – 8)
ROW 3 (9 – 10)
ROW 4 (10 – 11)

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.

ROW 19 (113 - 118)

ROW 20 (118 - 124)

ROW 21 (125 - 126)

ROW 22 (126 - 137)

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

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.*

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."

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).

```
                    XEQ "Y DEMO"
INPUT X?            INPUT X?
Y = 9.0000                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         } Clear LAST x.
03 +         }
04 1         } Add 1 to LAST x if flag 39
05 FS? 39    } is set.
06 ST+L      }
07 2         } Add 2 to LAST x if flag 38
08 FS? 38    } is set.
09 ST+L      }
10 4         } Add 4 to LAST x if flag 37
11 FS? 37    } is set.
12 ST+L      }
13 8         } Add 8 to LAST x if flag 36
14 FS? 36    } is set.
15 ST+L      }
16 LAST X    } Recall accumulated
17 FIX 0     } 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)

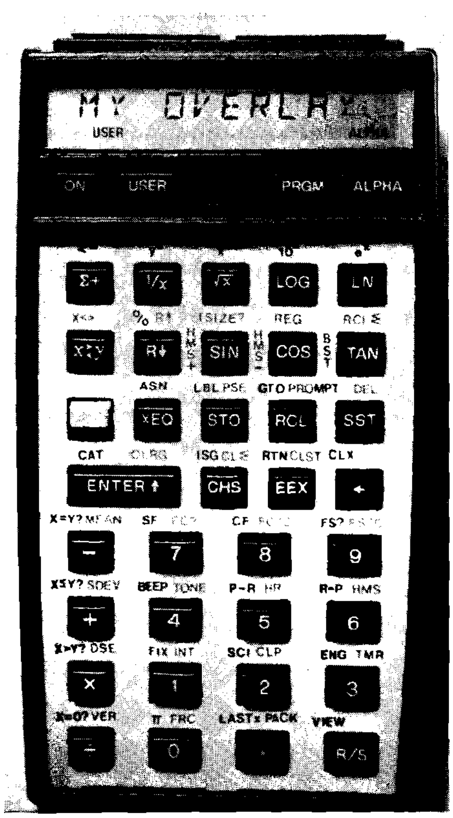
Flags 50-54. These flags always test clear and have little use in programs.

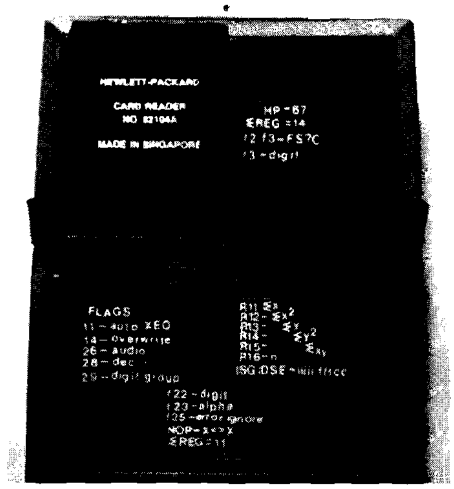
The HP-41 provides a wide variety of general-purpose and dedicated flags. Unfortunately, this brief discussion of flags could not cover all possible applications. For example, some clever programmers use flags as decision trees to provide multiple paths through a program. Three flags, for instance, could 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 irresistiblel) 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 [x] 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 (V4N5). 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 Shíblí** 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 i calculator games or to make long-lastin programs seem a bit shorter. I hope you lik this routine and find a place for it in you 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 a interesting feature you didn't mentior During operation, press R/S to stop th "show" in the display. Then press R/ again. The "eagle"—or as we call it, th goose—is back again, only "honking" thi time! Also, line 03 can be many, man things, limited only by your imaginatior —Ed.)

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

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

To find what the constant in line 17 woul be, I let "A" run for exactly one minute, the pressed R/S. I then recalled register 00 an divided it into 60. I was surprised to see 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 LBL 01 after line 04 and a GTO 01 just befor the END statement. Use "A" to start and "B at all mile posts.

There are several telephone timer pro grams in existence, but most of them ar quite long and complex so that the account for the various time and rat problems. This routine was sent to "2 Words" last September by **Peter Linlor** c Mountain View, California. While it isn't a sophisticated as others, we thought you' 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"
02 CLRG
03 CF 05
04 FIX 2
05 .01
06 STO 03
07 .009
08 STO 04
09 "LOAD AND RUN"
10 PROMPT
11♦LBL 00
12 +
13 +
14 +
15 RCL 00
16 RCL 03
17 HMS+
18 STO 00
19 FRC
20 X≠0?
21 GTO 01
22 RCL 02
23 ST+ 01
24 ISG 04
25 GTO 01
26 SF 05
27♦LBL 01
28 CLA
29 "⊦ "
30 FC? 05
31 "⊦0"
32 ARCL 00
33 "⊦ $"
34 ARCL 01
35 AVIEW
36 GTO 00
37 RTN
```

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 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.

"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
```

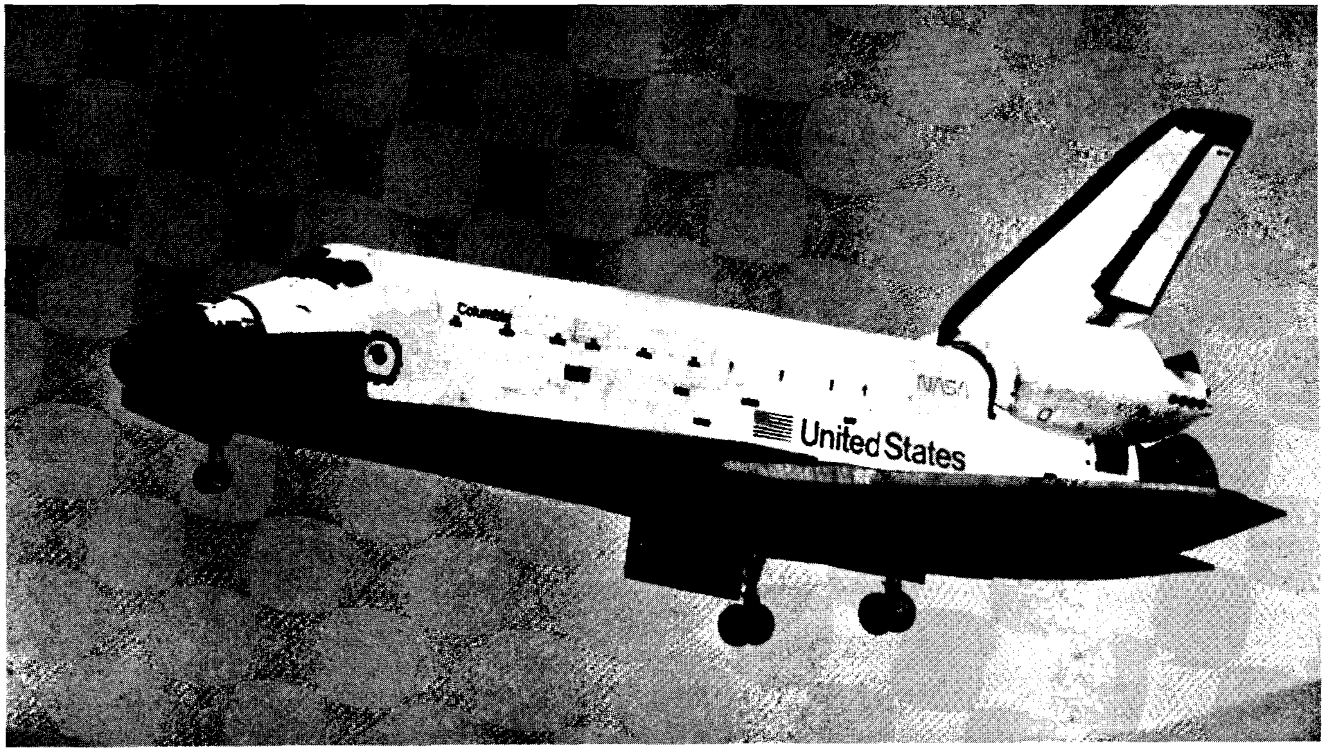
HP-41 Function List

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	1↑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.