HP-65 USERS' NEWS

HEWLETT hp PACKARD

Volume 1
Number 1

INSIDE AN HP-65

Most of you won't have seen what's inside your calculator. And it's not too likely that you'll *ever* take it apart! So we took a photograph of an HP-65 before it was fully assembled to show you the main internal parts.

On the right of the photo is the keyboard printed-circuit assembly. The dark area across the top is the light-emitting diode (LED) assembly that fits under the display window on the front of your calculator. Running across the keyboard are eight rows of key switch strips. The metal bosses on each switch strip are pressed down by the keys to make contact with printed traces underneath. The 35 keys on the keyboard are mounted directly above each "switch" and are held in place by a key retainer grid and the top case assembly.

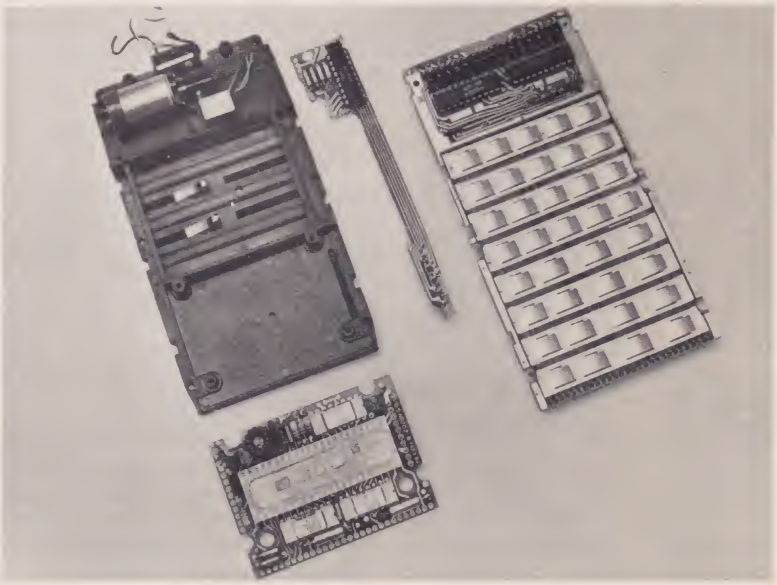
In the centre of the photo is the card reader board assembly. This is the interconnection between the read/write head, the electric motor, the battery and the logic board assembly. Notice that a printed-circuit board, not just wires, is used to connect the top and bottom levels of electronics.

On the left is the support plate assembly, with the card reader assembly mounted at the top area. You can see the motor, top left, and the read/write head to the lower right of the motor, all part of the underside of the calculator. Note the two battery connector springs in the centre of the support plate. You see these when you remove the battery pack from your HP-65.

At lower left is the logic board assembly. This contains the arithmetic logic unit and most of the other "brains" of the HP-65. This is attached to the support board directly above it in the photograph. It's hard to believe that this small assembly contains the equivalent of nearly 75,000 transistors. Just imagine the equivalent of that in vacuum tubes!

Of course, there are several other small parts and, naturally, the upper and lower case assemblies. Essentially, however, the parts shown in the photo (plus the battery pack) are what make the HP-65 the remarkable, compact device that it is.

IN THIS ISSUE



USERS' LIBRARY CORNER

As the number of program contributors grows, the HP-65 Library becomes more comprehensive. New and varied programs continue to arrive daily, and many of you are taking advantage of this wealth of time-saving programs.

After all, think of the advantages of not having to develop your own programs. First – you save time, and therefore money. There's no need to research the application. There's no need to program the problem. Neither is there the onerous task of checking the program. And, last but not least, you save the time and trouble of completely documenting everything. What could be more convenient?

Naturally, we are not suggesting that you forego the pleasure of formulating your own programs. On the other hand, why spend your time and effort doing what someone else might already have done? Therefore, prior to writing your own programs, thumb through your catalogue and program up-dates to find the applications that solve your problems... and send off your order. We are aware, of course, that we cannot yet offer programs for everyone's specific application but we urge experienced HP-65 owners to help us reach this goal by contributing new and useful programs.

TOP TEN PROGRAMS

Here are the 10 most requested programs, in their order of popularity, as we go to press:

1.	STOPWATCH	no. 00268A
2.	MOON ROCKET LANDING	no. 00287A
3.	RANDOM NUMBER GENERATOR	no. 00433A
4.	BAGELS-LOGIC GAME	no. 00242A
5.	BOOLEAN LOGIC EVALUATION	no. 00572A
6.	FAST FOURIER TRANSFORM OF 8 REAL DATA	no. 00276A
7.	BIORHYTHMS	no. 00284A
8.	ANALYSIS OF VARIANCE (TWO WAYS)	no. 00331A
9.	INVERSE FAST FOURIER TRANSFORM OF 5 COMPLEX DATA	no. 00408A
10.	OPTIMUM SCALE FOR A GRAPH	no. 00546A

WHAT YOU GET WHEN YOU ORDER A PROGRAM?

Many of you have asked this question, so here is the answer.

Primarily, you get a photo-copy of the original documentation from the submitter (see also, *When Contributing Programs*) which includes:

1. **The Program Description – with equations, variables used, operating limits and warnings, sketches (when applicable), sample problem(s), sample solution(s), and references used.**
2. **Operating Instructions–showing, step-by-step, how the data is keyed in and how answers are generated.**
3. **A Program Form – showing each step required to key in the program, including the associated program code and program comments. Register, label and flag usage are also documented here.**

With the information provided, it then takes but a few minutes to key in the program and record it on a magnetic card. Then, don't forget to snip off the upper left corner of the card so that the program cannot be accidentally erased.

WHEN CONTRIBUTING PROGRAMS

Unfortunately, we have to reject certain new program submittals. By far the main reason for rejection is *unacceptable documentation*. This means that it is (a) illegible, (b) incomplete, (c) documented in English language (d) not on original HP-65 Users' Library forms, or (e) on an unsigned *Program Submittal* sheet.

The only reason for these stringent standards is to ensure that those who order a copy of these programs will receive legible and understandable documentation. Therefore, before you submit a program to the library, please read the instructions in Part II of the Program Catalogue for details, and use the forms supplied.

While we're on the subject of contributed programs, may we make a plea for brevity? Please try to keep your program titles as short as possible. It might not always be possible but most can stand some shortening without losing anything in clarity. Use abbreviations if necessary. In fact, the title must not exceed the number of spaces provided on the *Program Submittal* form. Regarding the program abstract, please provide a brief description of the program's function but do not include any special characters or signs.

Once your program has been checked and included in the Library you will receive a letter noting the newly assigned Library Program number. And, to show our appreciation for your contribution, we will send you ten blank magnetic cards and any one free program of your choice from the Catalogue.

NEW PROGRAM TRENDS

All programs received since inception of the Library are now freely available. These are listed separately as up-dates to the first issue of the catalogue distributed last November and will be included in the next issue. As a matter of interest, here is the trend of application areas covered by incoming programs:

1. STRUCTURAL DESIGN/STATICS
2. SURVEYING
3. SPECIAL FUNCTIONS (MATH)
4. GENERAL STATISTICS/PROBABILITY
5. ASTRONOMY
6. FINANCE

These new programs can be ordered now. Please use the standard Program Order form, dated and signed, and make sure that you include a cheque or money order (payable to Hewlett-Packard S.A.). This helps us to respond to your request without delay and avoids lengthy invoicing and accounting procedures which tend to slow the process. However, those who prefer to make a Bank Transfer should remit directly to:

Swiss Credit Bank (Geneva Branch),
Hewlett-Packard S.A.
Account no. 327-360
Users' Library

Note: Please do not settle Program payments through your local Hewlett-Packard office.

PROGRAMMING TIPS

More and more HP-65 owners are finding new and ingenious ways to write and edit programs. Quite a few of you have been kind enough to send your ideas and hints to us. We, in turn, will continue to publish them in USERS' NEWS, as they arrive, so that everyone can profit from them. Here are the latest tips to reach us.

Rearranging Saves Steps

Most equations can be rearranged to save steps in memory. Consider the following (assuming x is in the x-register):

$$x^3 + 2x^2 + x + 1 =$$

STO 1 3 g y^x RCL 1 ENTER↑ × 2 × + RCL 1 + 1 +

This results in 14 steps. Now, see how rearranging the equation results in only 11 steps (and saves one register).

$$((x + 2)\,x + 1)\,x + 1 =$$

ENTER↑ ENTER↑ ENTER↑ 2 + × 1 + × 1 +

Store More Than Nine Constants

Two (or more*) constants (up to a total of 10 digits) can be stored in one register by multiplying or dividing them as necessary to make one a pure integer and the other a pure decimal fraction and by combining them as one decimal number. To get either number back, recall the composite number stored. Then f INT or f^{-1} INT, followed by multiplication or division by the appropriate power of 10, restores the original constant to the display. Look at the following example.

Let's use 12.34 and 567.89 as two constants to be stored in register R_1. Multiplying the first by 100 and dividing the second by 1000 gives 1234 and 0.56789, which are added and stored as 1234.56789. Then, pressing RCL 1 f INT yields 1234, which is then divided by 100 to obtain the original 12.34. Similarly, pressing RCL 1 f^{-1} INT 1000 × yields the original 567.89.

** Using combinations of f INT and f^{-1} INT, and multiplying and dividing as shown above, you can get more than two constants in one register. You will, of course, have to use smaller numbers, and use the integer keys more than once.*

Using DSZ as a Flag

Although very handy, and sometimes necessary, flags do use a lot of steps in memory. Therefore, it often pays to use DSZ as a flag, because you then save several steps. By storing 1 in register R_8, you create your own self-clearing flag using DSZ. When the program executes DSZ, it decrements the contents of R_8, which sets it to zero. Then it tests R_8 and, because it *is* zero, skips before continuing execution (just as when testing a flag that is set). The second time the program executes DSZ, R_8 is decremented again and the program pointer continues sequentially (just as when testing a flag that is clear) because the number in R_8 is no longer zero. The following examples should clarify this function.

Example 1	*Example 2*
LBL	LBL
A	A
g	f^{-1}
DSZ	TF 1
GTO	GTO
0	0
3	3
6	6
0	0
RTN	f^{-1} } *
LBL	SF 1 } *
0	RTN
4	LBL
0	0
0	4
RTN	0
	0
	RTN

** Notice that in Example 2 there are two extra steps (f^{-1} and SF 1) to clear the flag. This is unnecessary in Example 1 because DSZ decrements to zero and is self-clearing.*

Using DSZ as an Up Counter

Generally, the DSZ key is used as a down counter, with storage register 8 initialized as a positive number n and DSZ used in a loop to count n operations. A less obvious use of DSZ is as an up counter. Initializing storage register 8 to zero and executing DSZ in a loop keeps track of the number of loops performed. Remember to use CHS to display a positive result when DSZ is used as an up counter.

Save One Memory Location

The use of a subroutine call (A, B, C, D, or E) in place of GTO saves one memory location. If this technique is used, R/S should be used instead of RTN to return control to the keyboard.

Save Time with R/S

It is often very helpful to insert R/S statements while keying in programs to allow you to see results in the display as you progress through the routine during execution. This will avoid tedious searching through the entire routine when the final results of the program are incorrect. These R/S statements can be deleted later.

How to cut down Long Numbers

In cases where you need a particular number (especially a repeating number), it may take less steps to generate the number through calculation. For example, it is more efficient to calculate a decimal fraction by integer division than to use one memory step for each digit. To illustrate, 2 ENTER↑ 3 ÷ takes less memory than keying in .6666667.

Save Two Program Steps

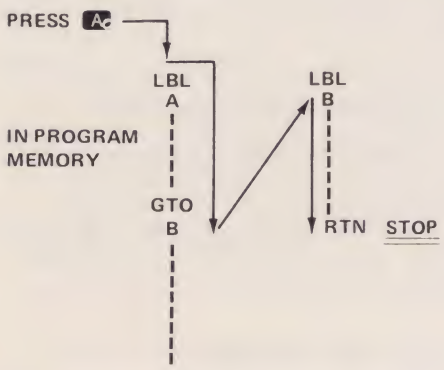
If you develop a program that insists on occupying 102 steps, and the first two steps are a label (say, LBL A), here's a way out of your dilemma. Merely delete LBL A at the beginning of your program. For example: If you press LBL A on the keyboard and it was the label you deleted at the top of memory, the program pointer will search for LBL A. When it doesn't find it, the program will start execution at the top of memory. However, there is one caution to observe before you use this programming tip. If the label you want to delete is called as a subroutine in your program, this trick cannot be used.

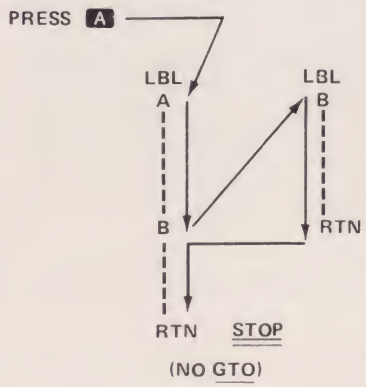
QUESTIONS – AND SOME ANSWERS

Here are some common questions received by Customer Service that also may be causing you trouble.

Question: When I call a subroutine, the program stops at the RTN of the subroutine. What's wrong?

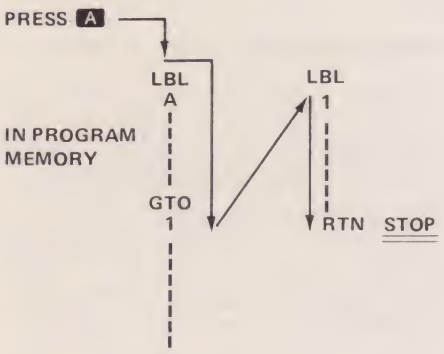
Answer: You used a GTO before the subroutine call letter. The diagrams below show the flow of the program for the two situations:

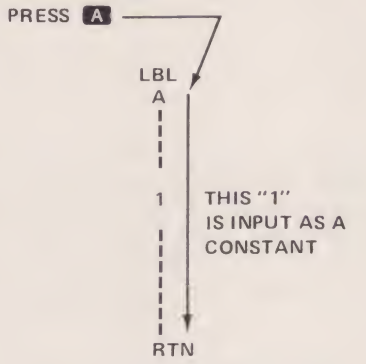




Question: Can I use letters and numbers to call a subroutine?

Answer: No. You may branch to both lettered and numbered labels, but subroutines are restricted to letters. A number without a GTO will be interpreted as a constant:





Question: In single-stepping through the Diagnostic Programs, sometimes it takes two or three presses of SST for the display to change. Why?

Answer: Some functions with prefix keys take two positions in program memory (such as f SIN), so two SST 's are required to execute the function. A merged operation that occupies only one memory location (such as g LST x) takes only one SST .

Question: Occasionally I find in memory a "41" code that I didn't put there. What is it? How did it get there?

Answer: There are two internal pointers that keep track of where you are when executing a program. The primary pointer is "on" when you're executing a main program. If you call a subroutine, the primary pointer is left pointing at the next step after the subroutine call. The secondary pointer is turned "on" to execute the subroutine. If you stop in the subroutine before you press RTN, and single-step to the call of the subroutine, you'll find the "41". This is the primary pointer; and when the subroutine is finished, it disappears and the primary pointer is turned back on.

Question: When I stop in the middle of a subroutine and restart the entire program, I get improper results. What happened?

Answer: The "41" was left in program memory from the previous execution. Therefore, if you *do* stop in a subroutine and wish to restart, you must *first* press RTN to correct the situation. The RTN clears and repositions the pointers at the top of memory.

Question: Sometimes when I press R/S , execution goes right through the first RTN it sees and stops at the second one. What should I do?

Answer: When this happens, you have used R/S to start a program that begins with a label and ends with a return. Your calculator is operating correctly; but read on.

Programs are executed from the keyboard or from within another program. And RTN has to satisfy two roles. One, it serves to stop a program called from the keyboard. Two, it serves as a signal to return to a previous program. In other words, in one case execution stops at a RTN and in the other case the RTN is a signal to transfer back to a program. The A to E keys are used, either from the keyboard or from the program, to control the status of each RTN . However, R/S does not affect the status of RTN 's. Consequently, if a series of label-return programs are in memory and execution is started with R/S , then the HP-65 will pass through every other RTN . Therefore, we do not recommend that R/S be used to execute label-return programs.

Most programming can be done without the use of R/S . For beginners, it should be considered primarily a debugging aid to be used to execute a program a portion at a time. For instance, if you execute to a R/S , the R/S in the program will stop execution. If your program is stopped by a R/S , you can restart by pressing R/S from the keyboard. The R/S will not affect the status of the RTN . A suggestion is: PAIR YOUR R/S 's. If you initiate execution with a R/S from the keyboard, then you must plan to have that execution terminated by a R/S in memory.

Note: We are aware that some of the more experienced HP-65 owners will not find much that is new to them among the tips discussed. However, we thought that more recent purchasers would appreciate the chance to profit from the findings of their "elders".

HOW FAST IS THE HP-65?

Several owners have asked us: "Just how fast is the HP-65? How long does it take to perform various cycles or functions?" Well, since there are many, many variables, it is not easy to answer the question precisely. However, we can tell you the results of some of our investigations.

Although microinstruction cycle times can vary as much as 293 to 339 milliseconds from one calculator to another, measurements on any given calculator can be used to give an idea of relative times required by various instructions. To time various functions, our Quality Assurance Department devised a method that used a looping technique and a stop-watch. Some of the more common execution times are tabulated below to give you an idea of the speed of the HP-65.

Operation	*Execution time (milliseconds)*
+	48
–	50
×	70
÷	110
g x≷y	17
g R↓	17
g R↑	17
g NOP	15
g π	56
g LST X	20
g ABS	53
g $1/^x$	136
g y^x	350 to 750
g n!	205 to 1116

HP-65 WINS AT LE MANS

As the champagne corks popped for the Matra Simca winner of last year's 24-hour race, there was cause for celebration in the HP-65 "pits" as well. One of the Matra Simca technicians carried an HP-65 in his pocket during the race to chalk up one of the first wins for the programmable calculator. Used from conception of the Matra Simca 670C and the new 680, the HP-65 was used to determine dimensions of parts and to calculate aerodynamic loads and chassis "tuning". During the race itself, the HP-65 contributed to the win by rapidly and accurately performing fuel consumption calculations and, simultaneously, keeping tabs on the "Index of Performance" classification.
Quite a performance to chalk up a first appearance "win" at such a prestigeous event!

TAKING CARE OF MAGNETIC CARDS

The magnetic cards in your HP-65 Application Pacs are strips of magnetic tape (similar to the tape in hi-fi or audio cassettes) placed on a relatively rigid plastic-coated material. Just as a magnetic tape can be erased by passing it over a strong magnet, so can magnetic cards. The closer the proximity of the magnet to the card, the less strong the magnet has to be; the further away, the stronger. To give these general statements a frame of reference, magnetometers at airports are not very strong to start with and they are far away – so they are safe. On the other hand, carrying your cards in a shirt pocket containing a magnetized screw-driver is not safe, even if the cards are in their plastic case. The part of the program on the cards closest to the screwdriver blade may be erased. A good rule of thumb to follow is to keep *any* kind of magnetized material at least 30 to 60 cm (1 to 2 feet) away from your magnetic cards.

HERE'S A BATTERY TIP

Here's a tip you can use to extend battery life when you are using your HP-65 in a long series of computations. Between actual calculations, press the decimal (·) key. Only the decimal point will be displayed. Therefore, battery drain will be exceptionally low. Then, when you want to continue calculating, merely press CLx and the display will return to 0.00.

APPLICATION PAC CORRECTIONS

If you own some of our Application Pacs, check the following corrections and mark them in your copy. If your copy is correct, you have a later, revised issue of the book.

E.E. Pac 1, EE1-02A, Page 11 (US and French versions)

The equation for MAG(Z) should read:

$$MAG[Z] = \frac{R_p[(1-\omega^2LC)^2 + R_s^2C^2\omega^2]^{1/2}}{[(1-\omega^2LC)^2 + (R_s+R_p)^2C^2\omega^2]^{1/2}}$$

Notice that Rp was missed in the original book.

E.E. Pac 1, EE1-03A, Page 14 (US and French versions)

Because of a programming error after step 80, it is possible to get the shift function stuck when card EE1-03A is used.

If your E.E. Pac 1 book is dated September 1974, the program listing is correct. If not, you have an original book, and the program listing on page 110 should be changed to read as follows:

Delete step 81 (32 f^{-1}).

In its place, add two steps:

35 07 g $x \gtrless y$
31 f

Under Notes: 2 on page 15, the keystroke sequence is now longer. Just before "SWITCH TO RUN", add another line as follows:

Press (SST) seven times, then (1).

E.E. Pac 1, EE1-14A, Page 46 (US version), Page 48 (French version)

In the schematic diagram, L_1 should be labelled L_2, and C_2 should be labelled C_3. (This error has been corrected in books dated September 1974).

In the equation for b_i on page 47, γ should be squared.

E.E. Pac 1, EE1-17A, Page 52 (US version)

The variable R should be defined as "radius of coil to centre of wire in inches."

E.E. Pac 1, EE1-18A, Page 54 (US version), Page 56 (French version)

The expression for Q yields results too large by a factor of 1000. Change the constant in the equation from 25.59 to 0.0256. Change the answer on the following page from $Q = 1.08 \times 10^5$ to $Q = 1.08 \times 10^2$. Change the program listing on page 128 as follows:

Step	*From*	*To*
67	02 2	83 .
68	05 5	00 0
69	83 .	02 2
70	05 5	05 5
71	09 9	06 6

E.E. Pac 1, EE1-08A, Page 28 (US version)

This program, *Minimum-Loss Pad Matching,* was incorrectly recorded. A35 08 ([g] [Ri]) code was accidentally substituted for a 31 ([f]) code at four steps past label E (page 116).

Other minor errata in E.E. Pac 1 are as follows:

Page 35 (US version) – Line 12 of the *User Instruction Form* should be [STO] [1] instead of [STO] [ ].

Page 38 (US version), page 40 (French version) – In Example 1, $C_2 = 2.20\,\mu F$, not $220\,\mu F$. In Example 2, $C = 10\,\mu F$, not $1\,\mu F$.

Page 64 (US version), page 66 (French version) – The equation at the top of the page should read:

$$\Theta = \frac{1.20083 \times 10^{-8}}{v}\,\ell f$$

Page 65 (US version), page 67 (French version) – Under line 3 of the *User Instruction Form,* "Frequency" should be f, Hz instead of f, MHz. Under line 7, "Frequency" should be f, Hz instead of f, MHz.

Page 111 (US version) – Step 20 should be 35 00 g LST X instead of 35 01 g NOP.

Page 126 (US version) – Step 5 should be 42 CHS instead of 43 CHS.

Stat Pac 1, Stat 1-08A, Page 22 (US version)

In paragraph (1), delete the sentence "More than 10,000 random numbers may be generated before values are repeated."

In addition, if a different sequence of uniformly distributed pseudo random numbers is desired, choose a starting value u_0 such that $0 \leqslant U_0 \leqslant 1$ and do:

1. U_0 [STO] [1].
2. Skip step 3 and perform step 4.

Math Pac 1, Math 1-23A, Page 52 (US version), Page 54 (German version)

Add the following note. "NOTE: for the case when A, b, c, are given, keys [f] [SIN] are used to find angles B and C. The HP-65 always returns the principal values. But sometimes secondary values are required, and this program will not compute them. To compute secondary values, see Math 1-16A."

Math Pac 1, Math 1-35A and 1-37A, Pages 74 and 78 (US version), Pages 76 and 80 (German version)

Note that the register R_9 is available for temporary storage only.

Math Pac 2, Math 2-19A and Math 2-20A, Pages 42-45 (US version), Pages 44-46 (French version), Pages 46-48 (German version)

Add the following note. "NOTE: this program uses register R_9, so f (x) should not use R_9. In particular, f (x) cannot involve trigonometric functions, polar/rectangular conversions, or comparison tests.

Navigation Pac 1, NAV 1-08A, Page 24. In the argument list at the botttom of the page, $d\lambda_i$ should be λ_i.

Navigation Pac 1, NAV 1-09A, Page 28. Under Notes on page 29, add a fourth note: "4. This program gives incorrect results when computing distances due east or due west across the date line. To obtain correct results, compute up to the date line and then proceed on the other side."

Navigation Pac 1, NAV 1-11A, Page 34. Under Notes: on page 35, add a fourth note: "4. If the vertex calculated does not seem reasonable, interchange points 1 and 2 and start again."

Navigation Pac 1, NAV 1-12A, Page 38. The equations for λ_{V1} and λ_{V2} should contain the term "sgn $(|\lambda_2 - \lambda_1| - 180)$" instead of "sgn (L_{MAX})" so that the program will work correctly when the initial and final positions are on opposite sides of the date line.

Navigation Pac 1, NAV 1-25A, Page 75. The second line of the keystroke solution should read:

0 [ENTER↑] 20 [C] [R/S]
1.30 [D] [E] ——————→ 185.03

The [R/S] keystroke was left out.

Aviation Pac 1, AV-1-01A, Page 10. This program halts prematurely during the heading calculation when wind vectors are from 90° or 270°. In this case, a zero is displayed instead of the correct heading. Pressing (R/S) will cause the program to continue on to the correct result.

Rhumbline Navigation, AV 1-16A, Page 50. This program requires an additional note on page 51, under Limits and Warnings, as follows: "This program gives incorrect results when crossing the international date line due east or due west."

*Position Given Heading Speed and Time, AV 1-18A, Page 56.** This program fails to give correct results for flights due west and, in some cases, due east. The equation halfway down page 56 should read:

$$LNG_D = LNG_S - \left[\frac{DIST \sin HDG}{60 \cos LAT}\right]$$

** The book lists this program as AV 1-18A. However, the original magnetic card is marked AV 1-18B.*

HP-65 APPLICATION PAC NEWS

New Application Pacs have been developed by Hewlett-Packard to provide HP-65 users with additional software in three application areas.

Covering the Thermal and Transport aspects of Chemical Engineering, Stress Analysis, and a French version for Numerical Control programs, detailed make-up of the new Pacs and date of availability are as follows:

Chemical Engineering Pac 1
No. 00065-67050

(Thermal and Transport Science)
(April 1, 1975)

1. Ideal gas equation of state
2. Redlich – Kwong equation of state (3 cards)
3. Reversible polutropic process for an ideal gas
4. Isentropic flow for ideal gases (2 cards)
5. One dimensional normal shocks for ideal gases
6. Fluid transport numbers (Nusselt–heat & mass, Reynolds, Stanton, Lewis, Schmidt, Biot and Prandtl, 5 cards)
7. Fanning friction factor and conduit flow
8. Conservation of energy (2 cards)
9. Von Karman analogy for heat and mass transfer
10. Heat exchanger analysis (heat transfer and effectiveness for cross-flow, counter-flow, parallel-flow and parallel-counter-flow heat exchangers, 6 cards)
11. Heat transfer through composite cylinders and walls
12. Straight fin efficiency
13. Natural convection (estimate coefficients for vertical cylinders and walls, and horizontal cylinders and plates, 3 cards)
14. Black body thermal radiation (3 cards)
15. Temperature or concentration profile for a semi-infinite solid
16. Hydrocarbon combustion (2 cards)
17. Curve fitting (linear, exponential, and power curve fitting, 3 cards)
18. Unit conversion

Stress Analysis Pac 1
No. 00065-67051 (May 1, 1975)

1. Two dimensional vector operations
2. Static equilibrium of a point
3. Static equilibrium of a rigid body
4. Properties of rectangular sections
5. Properties of circular sections
6. Properties of annular sections
7. Composite section properties (2 cards)
8. Bending stress in beams or torsional shear stress in circular shafts
9. Linear or angular deformation of a shaft
10. Thin-walled pressure vessels
11. Stress in thick-walled cylinders
12. Interference fits (2 cards)
13. Mohr circle for stress
14. Soderberg's equation for fatigue
15. Circular plates with simply supported edges
16. Circular plates with fixed edges
17. Rectangular plates (simply supported and with fixed edges, 3 cards
18. Cantilever beams (3 cards)
19. Simply supported beams (3 cards)
20. Beams fixed at both ends (3 cards)
21. Beams fixed at one end and simply supported at the other (5 cards)
22. Compressive buckling
23. Eccentrically loaded columns
24. Rectangular, reinforced concrete sections
25. Bolt torque

Commande Numérique Pac 1
No. 00065-67055 (May 1, 1975)

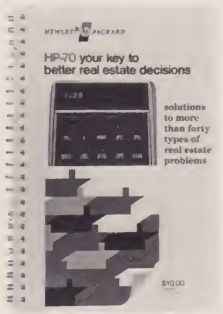
1. Point en coordonnées cartésiennes
2. Point en coordonnées polaires
3. Intersection droite-droite
4. Intersection droite-cercle
5. Intersection cercle-cercle
6. Points répartis sur une droite
7. Points répartis sur un cercle
8. Points répartis sur un cercle (données différentes)
9. Grille de points: calcul de tous les points
10. Grille de points: calcul de certains points
11. Point sur cercle défini par son angle au centre
12. Points de tangence à un cercle
13. Cercle passant par trois points
14. Cercle passant par deux points (rayon donné)
15. Décomposition en quarts de cercle
16. Translations
17. Rotations
18. Rotations puis translations
19. Translations puis rotations
20. Symétrie horizontale et verticale – symétrie point
21. Symétrie oblique
22. Conversion incrémental – absolu
23. Conversion absolu – incrémental
24. Normales à une droite
25. Normales à un cercle
26. Décalage d'un point d'une droite
27. Décalage d'un point d'un cercle
28. Trajectoire décalée (segments de droite)
29. Trajectoire décalée (mixte)
30. Interpolation circulaire
31. Cercle tangent à deux cercles (rayon donné)
32. Cercle tangent à droite-cercle (rayon donné)
33. Cercle tangent à deux droites (rayon donné)

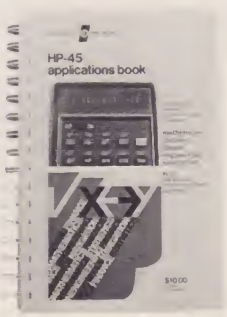
APPLICATION BOOKS FOR OTHER HP CALCULATORS

We thought that those of you who own one of our other calculator models might be interested to know that Application Books are now available covering the following areas:

HP-35: Math Pac
HP-45: Applications Book
HP-55: Mathematics Programs
HP-55: Statistics Programs
HP-70: Your Key to Better Real Estate Decisions
HP-80: Real Estate Applications

These publications are excellent aids to their respective calculators. They are designed to extend calculator usefulness by saving users' time and reducing the change of errors with clear demonstrations of the most efficient key-stroke sequences for solving the most commonly encountered complex problems. A further advantage is that users have access to a permanent record of solutions to sample cases. These Application Books can be ordered through your local Hewlett-Packard Sales Office.





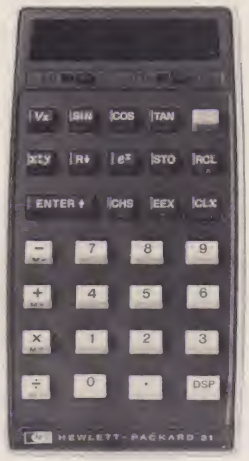
TWO NEW SCIENTIFIC CALCULATORS INTRODUCED BY HEWLETT-PACKARD

Continuing its role of pioneer and innovator, HP has introduced two new and improved scientific pocket calculators.

MORE FEATURES, MORE VALUE FROM THE HP-55

Hewlett-Packard engineers have designed a host of additional features into the new HP-55. Run through this impressive list of capabilities to see how the HP-55 sets new value standards for pocket calculators.

- 86 KEYBOARD FUNCTIONS! More than any other scientific pocket calculator.
- 20 STORAGE REGISTERS! More than any other scientific pocket calculator.
- 100 HOUR DIGITAL ELECTRONIC TIMER! Indicates hours, minutes, seconds, and hundredths of a second. Times and stores up to 10 separate events at a time. And all with 0.01 % accuracy!
- 7 ENGLISH/METRIC CONVERSIONS! Convert back and forth with a few keystrokes.
- 49-STEP KEYSTROKE MEMORY! Programming power at your fingertips. Solves repetitive or iterative problems with just a few keystrokes. The display even indicates program line numbers!
- 2 PROGRAM REVIEW KEYS! "Back Step" and "Single Step" keys let you review the entire keystroke memory. Or go to any line in memory by pressing the "Go To" key and the number of the line. Now you can position the memory to exactly where you want it.
- 4-REGISTER STACK! Retains as many as four intermediate data in sequence. At the proper time, these data are automatically entered into the calculation being performed, so you don't have to enter them manually.
- LAST X REGISTER! Automatically stores the last input argument of a calculation, for instant recall to correct an error or to enter the number into another calculation.
- 3 ANGULAR MODES! Calculate in degrees, radians, or grads.
- 4 STATISTICAL CALCULATIONS! With just a few keystrokes you now can calculate Summations, Means, Standard Deviation, and Linear Regression.
- PLUS DOZENS MORE! There's never been a pocket calculator like the HP-55.



HP-21: SMALLER SIZE BUT ADDED FUNCTIONS

A little bit extra for everyone but down another step in size, that sums up the HP-21.

Just launched at an incredibly low price, the HP-21 high quality electronic slide rule is specially designed to solve a variety of complex scientific problems. This is what you get:

- TRIGONOMETRIC FUNCTIONS:
  Sin x . Arc sin x
  Tan x . Cos x . Arc cos x
  Arc tan x . Use in either Degrees or Radians Mode. Rectangular Polar Coordinate Conversion
- LOGARITHMIC FUNCTIONS:
  Log x . 10^x . Ln x . e^x
- OTHER FUNCTIONS:
  Addition, Subtraction, Multiplication or Division in Serial, Mixed Serial, Chain or Mixed Chain Calculations. Full Four Function Register Arithmetic.
  Y^x . $1/x$. $\sqrt{x}$. π
- DISPLAY:
  10 Significant Digits (8 + 2 digit exponent displayed in Scientific Notation) with trailing zeros suppressed
  Fixed Decimal Notation with Automatic Overflow and Underflow into Scientific Notation
  Scientific Notation with Dynamic Range of 10^{99} to 10^{-99}
  Automatic Decimal Point Positioning and Selective Round-off
  Indicators for improper operations (ERROR in display) and low battery condition (lighted decimal points)
  Light-emitting diode (LED) display recessed for better contrast in harsh lighting
- MEMORY:
  Four-register Operational Stack (automatic memory)
  Stack Roll Down
  x, y Register Interchange
  Separate Addressable Memory with full Register Arithmetic.

And, of course, both machines embody the quality, experience, ruggedness and after-sales service you can rely on from Hewlett-Packard. For further information, please contact your local Hewlett-Packard Sales Office.

HP-65 Users' News is published periodically by

HEWLETT-PACKARD S.A.,
HP-65 Users' Library – Europe
7, rue du Bois-du-Lan,
P.O. Box 349
1217 MEYRIN 1
Switzerland.

Readers' comments or contributions will be most welcome. Don't hesitate to drop a line to the editor at the above address.