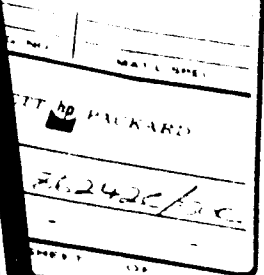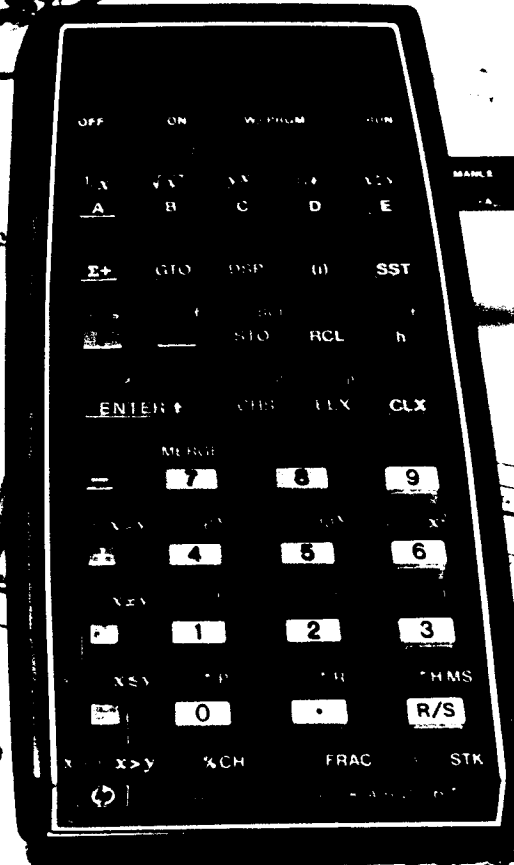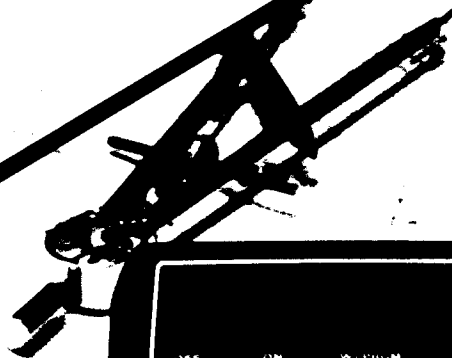# HEWLETT-PACKARD JOURNAL

# A Pair of Program-Compatible Personal Programmable Calculators

*One is a small, portable printing calculator and the other is a hand-held version without a printer. Both versions feature a "smart" magnetic card reader, and each can run the other's programs.*

by Peter D. Dickinson and William E. Egbert

T WO POWERFUL NEW fully programmable personal calculators in the tradition of the HP-65 (the first fully programmable pocket-sized scientific calculator)[1] are the pocket-sized HP-67, Fig. 1, and the HP-97, Fig. 2, a briefcase-portable calculator that has a built-in thermal printer, a buffered keyboard, and a large easy-to-read display. Both new calculators can read and record programs on magnetic cards using their built-in firmware-controlled magnetic card reader/recorders.

A novel feature of these two new calculators is that they have exactly the same programmable instruction set. Even the instructions that are used to control the thermal printer on the HP-97 can be programmed on the HP-67 and are interpreted as meaningful display commands when executed by the HP-67. Programs stored on magnetic cards are written in a code that can be properly interpreted by either an HP-67 or an HP-97. Programs written on either machine can be run on the other with appropriate key codes displayed (see Fig. 3).

## Design Philosophy

The basic design goal of the HP-67 and HP-97 development was to provide moderately priced computing tools capable of easily solving real-world problems. To achieve this goal, previous HP calculators were analyzed for possible improvement. The successful HP-65 contained many innovative features, including its magnetic card reader. Other features that have been well received by users include fully merged key codes, engineering notation, and a pause function. A selection of the best features of the HP calculator line can be found in the instruction set of the HP-67 and HP-97.

Speed and memory capacity were also considered during the design of the HP-67 and HP-97. Portable operation was desired, so battery capacity placed a

major limitation on speed. After several alternatives were considered, the low-cost, moderate-speed microprocessor used in the HP-91 and the HP-21/22/25 series[2] was chosen; it yields a system speed roughly equivalent to that of the HP-65.

After examination of hundreds of HP-65 application programs it was decided that data memory should be easy to address and efficient to use, since 17.3% of the operations in a typical program were found to be data

**Cover:** *The HP-97 and the HP-67 are fully programmable portable calculators that can run each other's programs. The HP-97 has a built-in thermal printer. The HP-67 and HP-97 are described in the article on page 2. The printer and the HP-91 Calculator that was the first to use it are described in the article on page 9.*
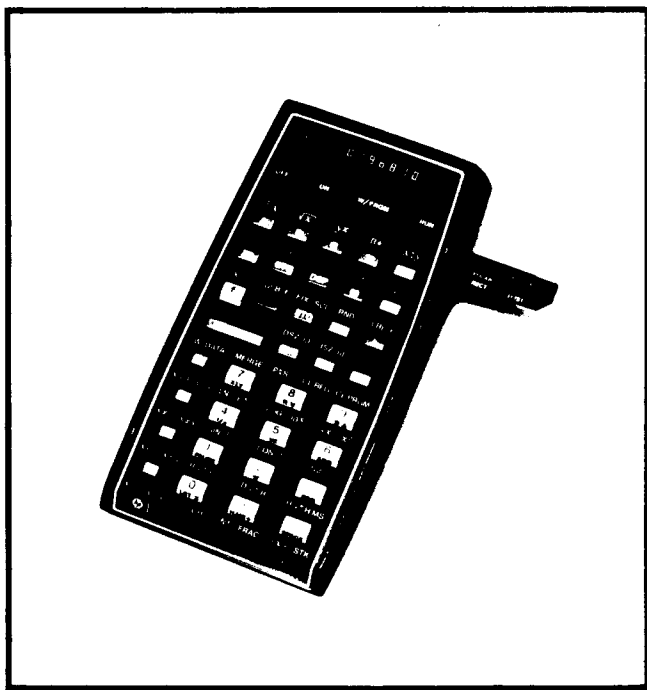
## In this Issue:

**Fig. 1.** *HP-67 is the fully programmable successor to the HP-65. Besides having more computing power than its predecessor, it is program-compatible with the HP-97 shown in Fig. 2.*

program steps and a data memory consisting of 26 data storage registers. Fig. 4 shows the register structure of the HP-67 and HP-97.

A final design goal was to provide useful new features within the framework of an RPN (reverse Polish notation) logic system. Some of these features include a "smart" card reader, data read and write capability, program merging, a round function, an improved pause, and several types of indirect addressing.

All HP-67 and HP-97 programmable features are implemented as microprogrammed firmware for the built-in microprocessor. Since these microprograms form the basic structure of the calculators, a great deal of effort went into producing efficient debugged code. To aid in program development, a cross assembler and a sophisticated simulator/debug system were developed for use on an HP 2100 minicomputer.

Final firmware was implemented in mask-programmable read-only memories. It was quickly recognized that a significant cost reduction for the HP-67 compared to the HP-65 would require improved packaging of the logic circuits. The multichip hybrid and multilayer printed circuit board that made the HP-65 possible was an expensive solution, so a new chip was developed that combined 1024 10-bit words of ROM with 16 56-bit registers on a single chip. This chip was designed to be packaged in an 8-pin plastic package, allowing all the HP-67 logic to be assembled in discrete packages on a conventional two-sided printed circuit board. Fig. 5 shows a comparison of the HP-65 and HP-67 logic boards.

store and recall operations. It was also felt that a balance should be provided between program memory and data memory. The HP-67 and HP-97 have a program memory consisting of 224 fully merged
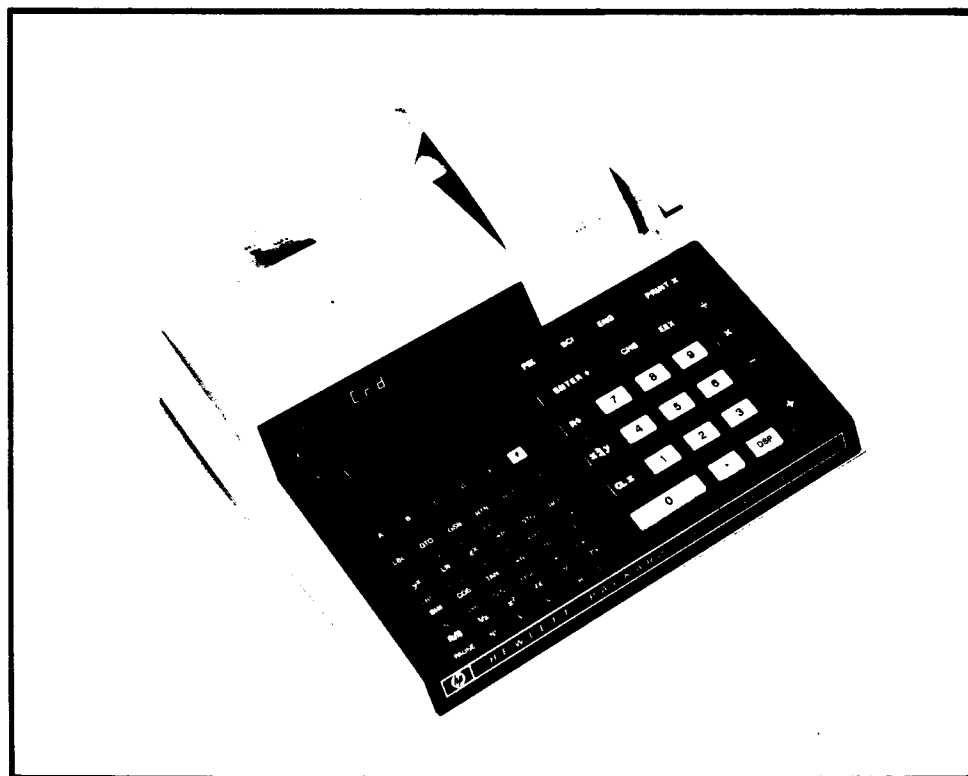


**Fig. 2.** *HP-97 is a fully-programmable printing calculator that fits in a standard attache case. It weighs only 1.13 kg including battery pack.*

**HP-67 Program Memory**

| Step | Instruction | Keycode |
|------|-------------|---------|
| 001 | **f** LBL **A** | 31 25 11 |
| 002 | **g** x² | 32 54 |
| 003 | **h** ÷ | 35 73 |
| 004 | **x** | 71 |
| 005 | **h** RTN | 35 22 |

**HP-97 Program Memory**

| Step | Instruction | Keycode |
|------|-------------|---------|
| 001 | LBL **A** | 21 11 |
| 002 | x² | 53 |
| 003 | **f** ÷ | 16-24 |
| 004 | **x** | -35 |
| 005 | RTN | 24 |

**Fig. 3.** *HP-67 and HP-97 programs are exactly the same, but some operations are prefixed in the HP-67 and not in the HP-97. Programs recorded on magnetic cards are automatically loaded correctly for the calculator being used.*

Improved production techniques were also studied as a means of improving manufacturing efficiency and reducing cost. One of the most significant new tools developed for the HP-67 and HP-97 is an automatic on-board IC tester designed to speed up production testing and service. The programming capability of the HP-67 and HP-97 made it possible to write self-testing diagnostic programs to aid in the production process.

**New Features**

Program memory in an HP-67 or HP-97 is composed of 224 8-bit steps. Programs are stored in the 32 56-bit registers contained on two of the new data/ROM chips. To increase programming efficiency all programmable operations are stored as single 8-bit instructions. Since eight bits only allow 256 different operations, a new form of display formatting and segmented data storage were used to achieve a fully merged instruction set. The HP-67 and HP-97 use 250 of the 256 possible instructions.

Among the new features of the instruction set is a round function, which rounds the internal representation of a number to correspond with the number shown in the display. This allows rounding to a specified number of digits after the decimal point, or rounding to a number of significant digits, depending on the type of display format chosen. The round feature is particularly useful in financial programs to assure that dollars and cents are properly rounded during chain calculations.

An improved pause function allows the keyboard to become active during a pause. Data can be entered, functions executed and magnetic cards can be read during a pause, after which the program will automatically resume execution.
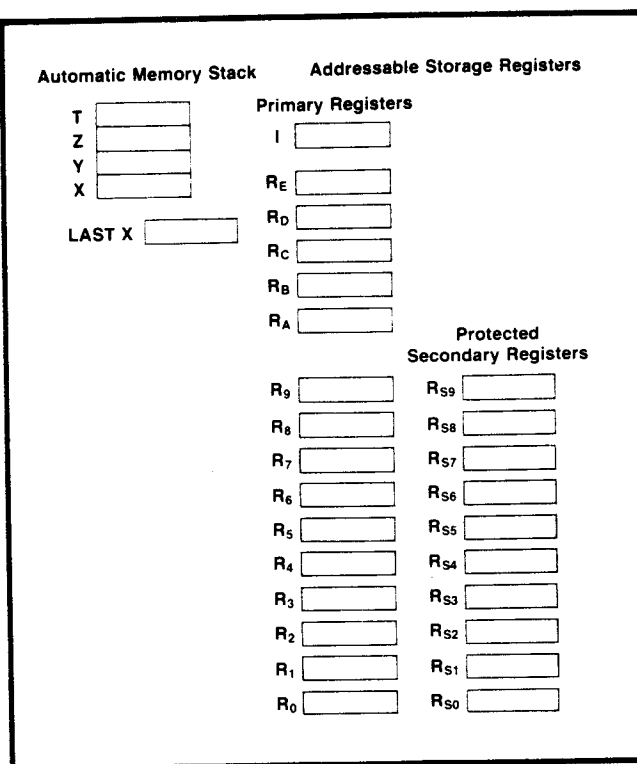


**Fig. 4.** *The HP-67 and HP-97 have a four-register operational stack and a register that retains the last number in the X-register, which is the display register. There are 26 addressable data storage registers, including the I-register, that can be used for indirect control. Protected secondary registers cannot be accessed directly with STO and RCL keys. These registers are used most often for statistical functions and programming, but they can be accessed from the keyboard using the P⇌S key.*

Program merging permits the user to add to or alter a program that is already in the calculator. The user specifies the last step of the current program to be preserved, presses **f MERGE** (HP-97), and loads a card containing the new program. The new program then replaces the current program starting with the step after the one specified.

An assortment of indexed and indirectly addressed operations provide computing power, interest, and challenge for the sophisticated programmer. Besides indirect store and recall of data, other indirect features include an indirect GO TO instruction, a form of indexed relative addressing, and indirect control of display formatting. To aid in editing, the HP-67 and HP-97 can go to any line number while in either run or program mode.

Other new features include three-level subroutine nesting, ten user-definable functions, advanced statistical functions, and 14 conditional branching instructions.

The thermal printer of the HP-97 can print programs, registers, the entire stack, or just the display register. A program can also be traced using the
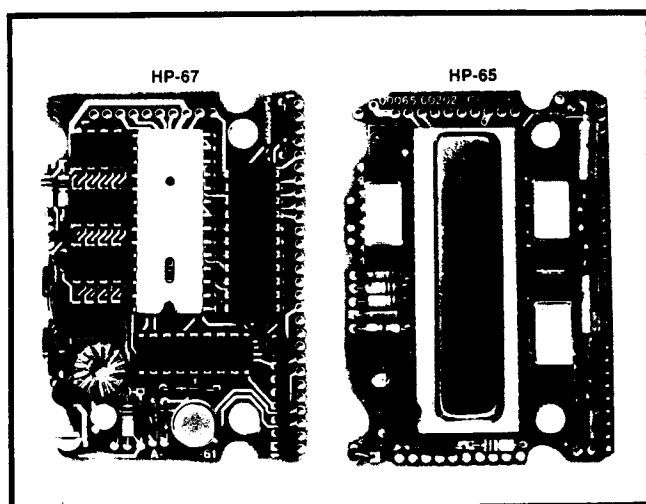
**Fig. 5.** *A new combined ROM and data-storage chip makes it possible to put all the HP-67 logic in discrete packages on a conventional two-sided printed circuit board. Here the HP-67 logic board is compared with the HP-65 logic board.*

printer, with operations and intermediate results printed out step by step to simplify program debugging. The printer itself is the same as that in the HP-91 Calculator (see article, page 9).

### I-Register for Indirect Control

Indirect operations are implemented using the I-register. The number in this register can be altered using the operations , , , and on the HP-97, or the equivalent key sequences on the HP-67. The last two operations increment or decrement the number in the I-register by one and skip an instruction if the number left in the I-register is zero.

The value contained in the I-register can also be used to *control* other operations. The (*indirect*) function combined with certain other functions allows the user to control those functions using the current number in the I-register. uses the number stored in the I-register as an *address*.

The indirect HP-97 operations that can be controlled by the I-register are:

, when the number in the I-register is 0 through 9, changes display formatting so that the number in the display contains the number of decimal places specified by the current number in the I-register.

, when the number in the I-register is 0 through 25, stores the value that is in the display in the primary or secondary storage register addressed by the current number in the I-register.

, when the number in the I-register is 0 through 25, recalls the contents of the primary or secondary storage register addressed by the current number in the I-register.

, , , , when the

number in the I-register is 0 through 25, performs storage register arithmetic upon the contents of the primary or secondary storage register addressed by the current number in the I-register.

, when the number in the I-register is 0 through 25, increments (adds 1 to) the contents of the primary or secondary storage register addressed by the current number in the I-register. In a running program, one step is skipped if the contents of the addressed register are then zero.

, when the number in the I-register is 0 through 25, decrements (subtracts 1 from) the contents of the primary or secondary storage register addressed by the current number in the I-register. In a running program, one step is skipped if the contents of the addressed register are then zero.

, when the number in the I-register is 0 or a positive 1 through 19, transfers execution of a running program sequentially downward through program memory to the next label specified by the current number in the I-register. Labels A-E and a-e are treated as 10-19 for indirect branching.

, when the number in the I-register is a negative number between −1 and −999, transfers execution of a running program *back* in program memory the number of steps specified by the current negative number in the I-register. This can be used to branch rapidly to any step number of program memory.

, when the number in the I-register is 0 through 19, transfers execution of a running program to the subroutine specified by the current number in the I-register. Like a normal subroutine, when a is encountered, execution transfers forward and continues with the step following the GSB instruction.

, when the number in the I-register is a negative number between −1 and −999, transfers execution of a running program *back* in program memory the number of steps specified by the current negative number in the I-register. Operation is then like a normal subroutine.

If the number in the I-register is outside the specified limits when the calculator attempts to execute one of these operations, the display will show When using the function, the calculator uses for an address only the integer portion of the number currently stored in the I-register. Thus, 25.99998785 stored in the I-register retains its full value there, but when used as address , it is read as 25 by the calculator.

Fig. 6 is an example of a program that uses the indirect GO TO and indirect store and recall of data. This program, "Follow Me," converts the HP-97 into a learn-mode four-function calculator with percent key and automatic constant.

### Intelligent Card Reader

Among the important features of the HP-67 and

5

**Follow Me**

(Fig. 6 program listing — a dense HP-97 program printout with step numbers, instruction mnemonics, and annotations including "Clear registers and set index at 24 to begin sequence", "Perform addition and put addition code of 1 in display register", "Perform subtraction and put 2 in display, then transfer to LBL 0", "Perform multiplication and put 3 in display", "Perform division and put 4 in the display", "Decrement step count. GTO function store / GTO error.", "Store function code and return operation result.", "Perform %, store display register value, and put 5 code in display", "I/O halt code of 6 put in display after storing display register value.", "Constant code of 7 put in display after display value is stored.", "If I is non zero after decrement, store code. First 24 indicating that too many operations have been attempted.", "Store constant code and", "recall constant value", "If I is non zero after dec store cd. GTO error. Store code and return display to proper status", "Store 24 in I to reset counter and store zero code in Rg for auto reset at end of sequence", "Store display value, access code after dec. put code in I, transfer to LBL corresponding to code.", "Reset to start new sequence by setting I to 24 and returning output to display.", "Perform subtraction.", "Perform addition and return to LBL E for next instruction", "Perform multiplication", "Perform division.", "Perform %", "Halt for I/O.", "Recall constant.")

REGISTERS / LABELS / FLAGS / SET STATUS table

**Fig. 6.** *A typical HP-97 program illustrating the use of indirect data storage and recall and the indirect GO TO operation. This program converts the HP-97 into a learn-mode four-function calculator.*

HP-97 is the "smart" firmware-controlled card reader. In the HP-65 a program was either recorded on or read from a single side of the card. The HP-67 and HP-97, however, allow both program and data cards to have either one or two sides of meaningful information. The smart card reader allows either side of a two-sided card to be read first and properly stored. If the other side of a card must be read, the display shows Crd to prompt the user. Data and programs from cards are automatically recognized and stored in their proper locations in the calculator. A special merge function allows two programs on separate cards to be merged together in program memory to form a single unit. Either data or program cards can be placed in the reader while a program is running and read automatically under program control. This allows up to 112 additional program steps or selected data registers to be read into the calculator and acted upon without user intervention.

To help reduce programming overhead the status of the calculator is normally initialized automatically when a program card is read. Display format, trigonometric mode, and user flags are set to the state they were in when the card was recorded.

The HP-67 and HP-97 use the card reader developed for the HP-65. Although one of the more interesting challenges was adapting the card reader to the new desk-top package, the basic mechanical design and two-track recording scheme that have previously been described remained unchanged.[1,3] In the HP-65, however, the control of the card reader was accomplished in an entirely different fashion. The microprocessor was effectively bypassed during read and write operations and programs were transferred directly between the card reader and the program memory via the interface circuit. The smart card reader in the HP-67 and HP-97 is made possible by the new way that the card reader interacts with the microprocessor under firmware control.

Fig. 7 shows the system organization of the HP-67. When a card is inserted into the card reader the motor switch is activated, setting an input flag on the card reader controller (CRC). The microprocessor, when it is not otherwise busy, interrogates the CRC via the $I_S$ line to see if a card is present. The CRC sends a pulse back to the arithmetic, control, and timing circuit (ACT) on the FLG line, indicating the presence of the card. If the PRGM/RUN swich is in the RUN position and the write data function has not been called for, the firmware sends an instruction to the CRC indicating that a read operation is about to take place and then instructs the CRC to set an output flag that turns on the motor by enabling the motor drive circuit on the read/write chip. As the motor drives the card over the head, flux reversals in the magnetic medium are sensed by the head, amplified and converted to logic levels by the read/write chip, and passed on to the CRC.

The CRC loads the bits into a pair of 28-bit buffers. When one buffer is filled the CRC switches over to the other buffer and signals the ACT that data is ready to be transferred to it via the DATA line. There are 952 bits on each side of the card, and the card cannot stop and wait for the microprocessor to be ready. Therefore, once the bits start coming in, the ACT must receive, process, and store the contents of one 28-bit buffer in not more than the time it takes to load the
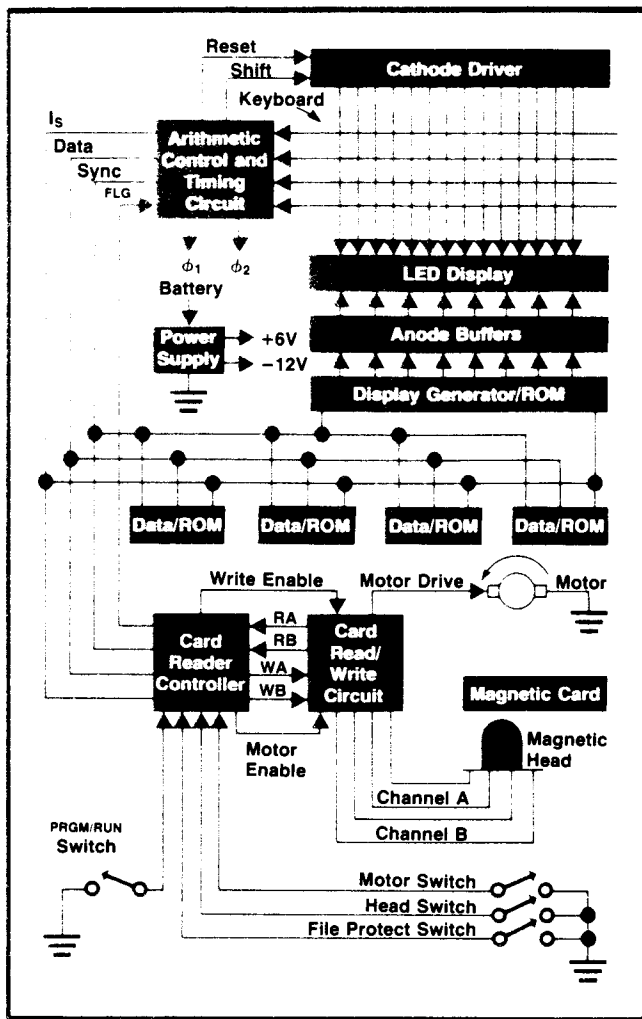
**Fig. 7.** *System organization of the HP-67. The card reader is controlled by the microprocessor so its operation is natural and foolproof.*

other 28-bit buffer, so that no bits are lost.

The card passes through the card reader at a nominal 6 centimeters per second. Since the recorded density is slightly greater than 160 bits per cm, the bits must be accepted at approximately a 1-kHz rate. But the microprocessor is capable of executing only approximately three microinstructions in the one millisecond between bits. With the CRC buffering the incoming bit stream, however, the microprocessor need only concern itself with the card reader every 28 milliseconds, on the average. It therefore has approximately $3 \times 28 = 84$ instruction times to process each 28-bit record.

The requirement that cards recorded on any HP-67 or HP-97 be readable on any other HP-67 or HP-97 made for additional considerations. First, the card reader speed can vary ±5% of nominal from one calculator to another. Second, the clock rate controlling the microinstruction execution rate can vary ±5% of nominal from one calculator to another. The firmware

had to be designed taking worst-case values into account, so a card recorded on a calculator with a slow card reader and a fast clock can be read on a calculator with a fast card reader and a slow clock without any bits being lost. This meant that the microprogrammer had to assure that each record could be processed in something less than 84 instruction times.

**Card Format**

Fig. 8 is a photograph of a magnetic program card with the recorded information made visible. The first 28-bit record on the card is the header, which carries machine status information if the card is a program card. It must be decoded for one of six card types:
- One-sided program
- First side of two-sided program
- Second side of two-sided program
- One-sided data file
- First side of two-sided data file
- Second side of two-sided data file.

The decoding of the header determines where the remainder of the information on the card should be stored and whether or not the user should be prompted to turn the card around and read it again by displaying the message Crd

The next thirty-two 28-bit records are either 112 program steps (3½ program steps per record) or 16 data registers (½ data register per record). Since the program and data storage memories are actually identical groups of 56-bit shift registers, records must be packed, two per register, prior to storing. The process of storing each record also includes adding it to a running sum of all the previously read records to form a checksum. This computed checksum is compared to the recorded checksum, which is the last record on the card, to see whether any errors occurred in the reading process. If an error did occur the firmware clears memory and notifies the user by displaying Error.
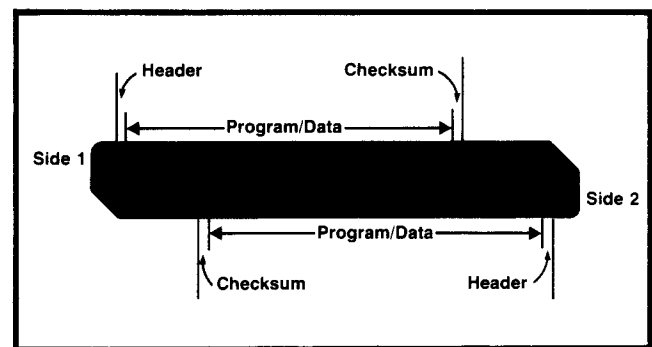


**Fig. 8.** *A typical two-sided program card with the magnetic information made visible. The header tells the firmware what kind of card it is and carries machine status information. Following the header are either 112 program steps or 16 data registers. The last record on the card is a checksum.*

7

Writing a card is fundamentally the same process in reverse. Writing is attempted if a card is inserted into the card reader with the PRGM RUN switch in the PRGM position or if a write data command has been issued either from the keyboard or by the program in the RUN mode. When the card reaches the head, the head switch is activated. If the corner of the card has not been clipped to protect the card, the write protect switch will also be activitated and writing will commence. The firmware generates and transfers to the CRC the 28-bit header record for clocking out onto the card via the read/write chip. Then the appropriate program or data registers are retrieved, unpacked, and transmitted to the CRC. Each record, as it is processed, is added to the checksum, which is then recorded as the last record. If an attempt is made to write on a file-protected card, an error flag is sent from the CRC to the ACT, the firmware generates an Error display, and the card is passed without modification.

## Acknowledgments

The development of the HP-67 and HP-97 required significant contributions by many people in several HP divisions. The authors would particularly like to express their appreciation to Bob Toda and Richard Jordan for the product design of the HP-67 and HP-97, respectively; Bill Fernandez for his wide range of technical support on both products; Bill Mitchell for the HP-97 keyboard design; Chuck Dodge for the industrial design of both products; Ken Haven for coordinating the design of the plastic parts; Glenn Theodore for his comprehensive product management; Hugh Field and his applications engineers for their contributions to the product definitions and applica-

tions pacs; Dennis York for the HP-67/97 Standard Pac; Don Mackenroth for both excellent owner's handbooks; Walt Henderson and Roger Evans for their cooperative approach to production engineering of the HP-67 and HP-97 respectively; Rich Wheeler for his objectivity and cooperation in the joint development of the card reader controller; She-Chung Sun for the design of the HP-67 cathode driver chip; Dennis Harms for the high-accuracy algorithms used in both products; Jim Speaker and Bob Youden for their understanding but thorough Q.A. of both products; and Ed Heinsen, section manager for both products, who managed by objectives and who also provided technical assistance on the software simulator. ⌐

## References

1. C.C. Tung, "The 'Personal Computer': A Fully Programmable Pocket Calculator," Hewlett-Packard Journal, May 1974.
2. M.J. Cook, G.M. Fichter, and R.E. Whicker, "Inside the New Pocket Calculators," Hewlett-Packard Journal, November 1975.
3. R.B. Taggart, "Designing a Tiny Magnetic Card Reader," Hewlett-Packard Journal, May 1974.

**Peter D. Dickinson**
Now section manager at HP's Data Terminals Division, Peter Dickinson was project manager for the HP-67 and HP-97 Calculators. Peter received his bachelor's and master's degrees in electrical engineering from Cornell University in 1969 and 1970. He started working for HP on a part-time basis in 1967 and joined full-time in 1970. His research and development activities have touched on power supplies, CRT terminals, integrated circuits, and calculators; he holds one patent and has several pending on various aspects of calculators and CRT terminals. A native of the state of New Jersey, Peter now lives in Monte Sereno, California. He and his wife, who also works for HP, are expecting their first child soon. He's a licensed private pilot, plays guitar, makes stained-glass lamps and windows, dabbles in real estate, and likes to go backpacking.

**William E. Egbert**
Bill Egbert was project leader for the HP-67 and did the microprogramming for both the HP-67 and the HP-97. He also developed the motor speed control for the printer used in the HP-97 and HP-91. He holds a patent on the speed control and has another pending on the new features of the HP-67/97. Bill received his BSEE degree from Brigham Young University in 1973 and his MSEE from Stanford University in 1976. He's been with HP since 1973. Born in Fallon, Nevada, he's married, has two small children, and lives in Sunnyvale, California, but will soon be moving with his HP division to Corvallis, Oregon. He and his wife are very active in the Mormon Church. As for hobbies, Bill says, "My only hobby is turning science fiction that I read at home into science fact at work."

# Portable Scientific Calculator Has Built-In Printer

*It's the first of a new generation of small, portable, printing calculators that operate on rechargeable batteries or line current.*

**by Bernard E. Musch and Robert B. Taggart**

N THE FOUR YEARS SINCE the introduction of personal portable computing to the engineering and scientific community, calculator users have become increasingly more demanding in the capabilities and features they require. This has led to a succession of progressively more powerful and less expensive calculators with improved performance in both functions (firmware) and hardware. The HP-91 (Fig. 1) is the first of a new series of HP products that follow this general trend. Two others are described in the article beginning on page 2.

Not only will the HP-91 solve the user's engineering and scientific problems, but it will solve them more accurately than many earlier calculators, and it will, if the user wishes, write problems and answers down for him. There is nothing conceptually novel about a printing calculator. Long before electronic calculators, mechanical adding machines provided output on paper tape. What is new in the HP-91 is the thermal printing technology that provides a small quiet printer with 5.7-cm-wide tape, the desk-top-sized keyboard for ease of touch entry, and a large, easy-to-read display all in a 23×20×6-cm package that fits easily into a standard attache case and weighs only 1.13 kg including a self-contained battery pack.

## What It Prints

One of the most important decisions that had to be made in functionally implementing the HP-91 was what information to print on the tape and in what format. Providing meaningful printout on a reverse Polish calculator did present some interesting challenges. Since we could easily get 20 characters on the tape, and since the longest possible number is 15
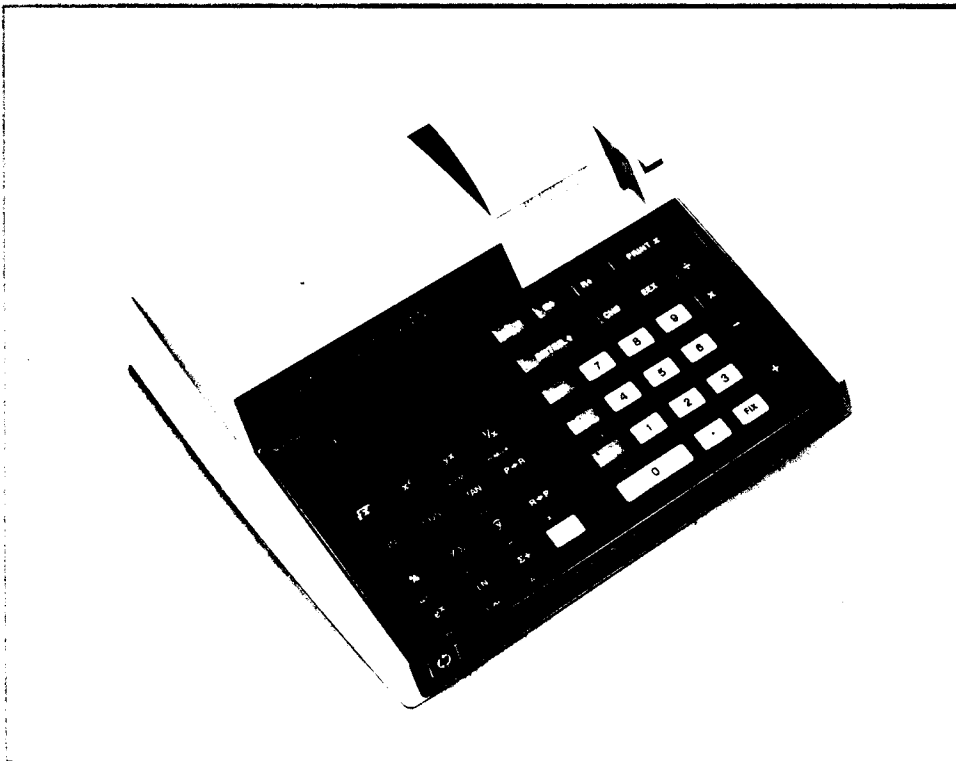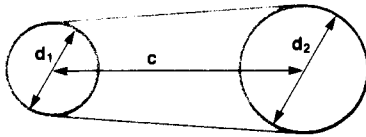


**Fig. 1.** *HP-91 Scientific Printing Calculator fits in a standard attache case and weighs only 1.13 kg including battery pack.*

**Problem:**
Compute the length (L) of a belt connecting two pulleys of known diameters ($d_1$ and $d_2$) and center-to-center separation (c).



$$L = \pi d_2 + (d_2 - d_1)(\tan\psi - 4)$$
$$\psi = \cos^{-1}\frac{d_2 - d_1}{2c}$$

**HP-91 Solution:**

Assume $d_1$ = 150 mm
$d_2$ = 250 mm
c = 650 mm

Mode Switch: RAD
Display Mode: FIX 5
Key Sequence: 150 STO 1 250 STO 2 650 STO 0 RCL 2 RCL 1 − RCL 0 ÷ 2
+ cos⁻¹ STO 3 RCL 2 π × RCL 2 RCL 1 − RCL 3 TAN RCL 3
− × +

| Print Switch: NORM | Print Switch: ALL |
|---|---|
| CLEAR | CLEAR |
| 150.00000 S 1 | 150.00000 S 1 |
| 250.00000 S 2 | 250.00000 S 2 |
| 650.00000 S 0 | 650.00000 S 0 |
| R 2 | R 2 |
| R 1 | 250.00000 *** |
| − | R 1 |
| R 0 | 150.00000 *** |
| ÷ | − |
| 2.00000 ÷ | 100.00000 *** |
| COS⁻ | R 0 |
| S 3 | 650.00000 *** |
| R 2 | ÷ |
| π | 0.15385 *** |
| ^ | 2.00000 ÷ |
| R 2 | 0.07692 *** |
| R 1 | COS⁻ |
| − | 1.49380 *** |
| R 3 | S 3 |
| TAN | R 2 |
| R 3 | 250.00000 *** |
| − | π |
| × | 3.14159 *** |
| + | ^ |
| 1932.16658 *** | 785.39816 *** |
|  | R 2 |
| User presses PRINT x to print | 250.00000 *** |
| answer. | R 1 |
|  | 150.00000 *** |
|  | − |
|  | 100.00000 *** |
|  | R 3 |
|  | 1.49380 *** |
|  | TAN |
|  | 12.96148 *** |
|  | R 3 |
|  | 1.49380 *** |
|  | − |
|  | 11.46768 *** |
|  | × |
|  | 1146.76842 *** |
|  | + |
|  | 1932.16658 *** |

**Fig. 2.** *Sample problem illustrates HP-91 printout modes. In ALL mode, the problem can be recreated simply by pressing every key printed, reading left to right and top to bottom, and ignoring all answers, which are identified by ***

command the machine to print an answer by pressing the PRINT x key. To distinguish a printed answer the mnemonic field is flagged with *** to catch the eye. In addition to PRINT x there are three functions that output only and have no effect on the calculator's memory or display; they merely list the contents of certain registers with appropriate mnemonics. LIST STACK lists the operational stack with stack designators, X,Y,Z, and T. LIST REG lists the contents of the 16 data storage registers with register-number designators. LIST Σ lists the contents of the statistics registers with the designators N, ΣX, ΣY, ΣX², ΣY², ΣXY, since these are the values in these registers if their contents have been accumulated by means of the Σ+ key. The calculator's response to the PRINT x or the LIST functions is the same in all print modes. In MAN (manual) mode the printer is disabled unless one of these functions is called for.

A reverse Polish calculator is ideally suited to provide a recorded history of a problem. The mode that causes the HP-91 to do this is NORM (normal). In normal mode every keystroke, whether digit entry or function, is faithfully recorded on the tape. The design objective was to make it possible to recreate a problem merely by consulting a tape generated in normal mode. An example is shown in Fig. 2. Digits are printed as entered, regardless of display format, except that trailing zeros are added to fixed-point numbers, if possible, to fill out the field and line up the decimal points on the tape. Answers printed by the PRINT x key or LIST functions are in the user's chosen display format. Answers can be printed at any time by pressing PRINT x. When recreating the problem all the user needs do is press every key (digit

characters long (ten mantissa digits, two exponent digits, one decimal point, and two signs), there is enough extra room on a line for a single space and a four-character function mnemonic. The user may

---

## Printing on the HP-97

The HP-97 uses the same printer mechanism and electronics as the HP-91. The tape format for the two machines is also very similar. All the printing operations of the HP-91 are found on the HP-97 as programmable instructions with the exception of LIST Σ which is easily implemented by writing a small program.

Differences between the two machines are, for the most part, the result of the fact that the HP-97 is programmable while the HP-91 is not. For example, the ALL mode of printing on the HP-91 corresponds to the TRACE mode on the HP-97. When running a program, TRACE mode records the step number and mnemonic of each instruction as it is executed as well as printing intermediate answers. Such a listing can be used to trace the execution of a program to isolate errors. The HP-97 can also print program listings with optional printing of keycodes. Printing a program is, of course, not programmable. Effective use of the HP-97 printer can simplify the task of writing, debugging and documenting a program and make output results permanent for later use.
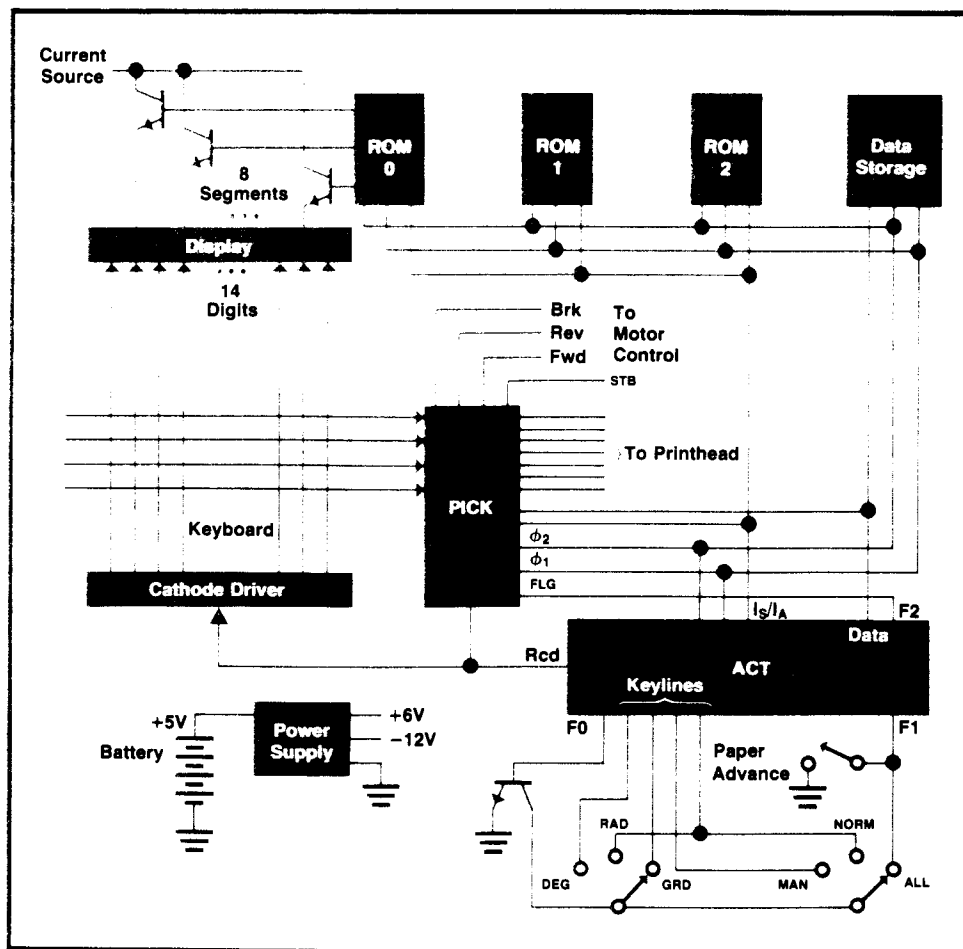
*-William E. Egbert*

**Fig. 3.** *HP-91 uses the microprocessor developed for the HP-21 series of pocket calculators.- Two new circuits in the HP-91 are the 14-digit cathode driver and the printer interface control and keyboard buffer (PICK).*

entry or functions) printed in the tape reading left to right and top to bottom and ignoring all answers (numbers tagged ***).

In ALL mode the answer for every function is printed on the next line after the function key is pressed and its mnemonic is printed. One way to think of ALL mode is that the machine is doing an automatic PRINT x after any function that could change the value of the displayed X register (other than ENTER ↑, storage operations, or clearing operations). In ALL mode, functions that provide more than one number in an answer (P→R, R→P, ↳R, ×, S, RCL Σ+) print both numbers with appropriate mnemonics to distinguish them.

### Hardware Organization

Fig.3 shows the organization of the system hardware. Regular HP Journal readers may recognize this system as being very similar to the HP-21/22/25 calculator series[1] and in fact this is the case. The main system processor (ACT), the ROMs, and the data storage circuit are essentially the same as those used in the hand-held calculators. Only two new integrated circuits had to be developed for the HP-91, the 14-digit cathode driver and the printer interface control and keyboard buffer circuit (PICK).

Because the combination of complex scientific functions and shuttle-head thermal printing occasionally keeps the system processor busy for extended periods, it was decided to buffer the keyboard so that keystrokes are recognized regardless of the state of the system. This is particularly important because the keyboard mechanism is specially designed for this series of calculators; it has a positive but light feel that is ideally suited for rapid key entry.

Since the new PICK circuit was required for printer control anyway, we decided to include a seven-keystroke-deep buffer on this device. Every keyswitch has two contacts, one connected to one of the four key lines that input to the PICK, and the other connected to one of the 14 digit drive lines from the cathode driver. Contact is made when the key is down. The reset cathode driver signal (RCD), which synchronizes the digit scan to the display information in the ACT, is also fed to the PICK, so when one of the key lines is pulled down, an appropriate eight-bit code generated by a counter is loaded into a first-in-first-out shift register. The contents of the key buffer are shipped out on the DATA line when the processor is ready for another key.

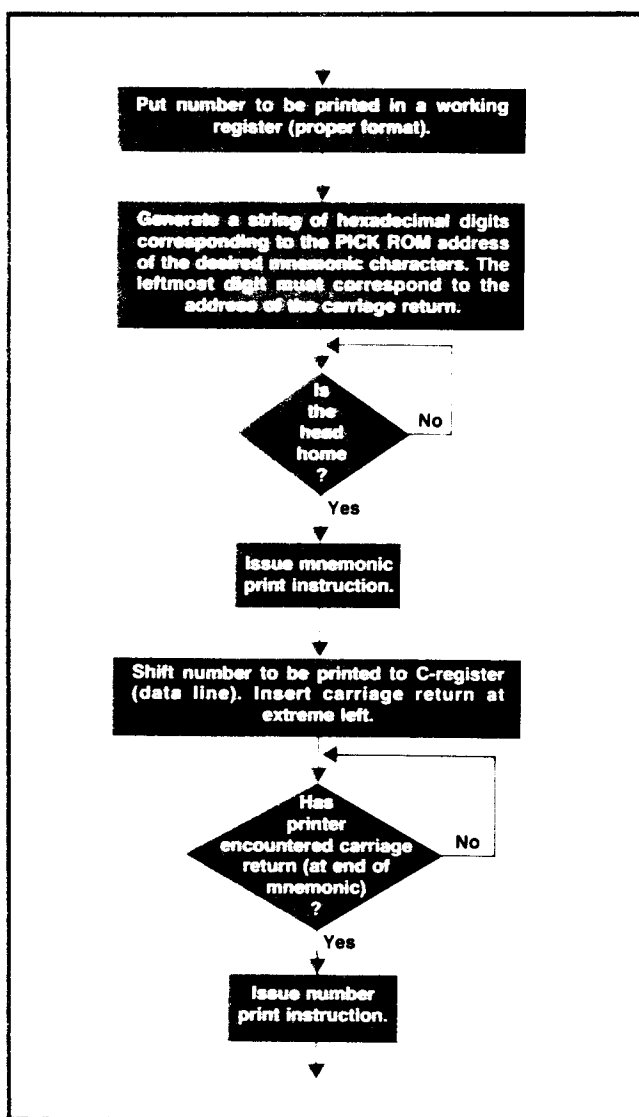Since the main system key register is not used to in-

**Fig. 4.** *Flow chart for printing a typical line containing a four-character mnemonic on the right and a 15-character numeral field on the left.*

terrogate the keyboard, we connected the ACT key lines to the calculator's three-position mode switches. Toggling the software-settable output flag on the ACT makes the ACT's key buffer respond to a dummy "key down" signal and as a result the contents of the buffer indicate the position of the two mode switches at any time.

The other function of the PICK circuit is to control the printer. The printer can print any of 64 5×7-dot-matrix characters, at least one of which must be a special end-of-line or carriage-return character. The characters are specified as mask-programmable ROM on the integrated circuit so that future machines can use other character sets and the characters may be shifted about to minimize the printing portion of the microprogram. For information exchange with the ACT, the PICK circuit resembles a high-order data

storage device and the information to be printed is transmitted serially along the DATA line.

Two of the printer-related instructions that actuate the PICK are translated as "Is the printhead in the home position?" and "Has the carriage-return character been encountered in this print line?" If either answer is true the PICK will pull up one of the external flag lines on the ACT. When it is determined that the printhead is in the proper position for printing, one of five possible print instructions, which we call PRINT 0, PRINT 1, PRINT 2, PRINT 3, or PRINT 6, is issued. A six-bit address is required to select one of 64 characters, but since the ACT processes data in either BCD or hexadecimal four-bit digits, it was thought advantageous for the printer to respond to four-bit address inputs. This is particularly useful when printing only numerals that have been built up into four-bit digits for display purposes. The PRINT 0 instruction causes the PICK to add 00 to each four-bit digit on the data line to complete the full six-bit address and print the appropriate character. PRINT 1 adds 01, PRINT 2 adds 10, and PRINT 3 adds 11. The PRINT 6 instruction takes the DATA line output and processes it six bits at a time to print any of the 64 characters.

The data word is only 14 digits (56 bits) long, but the printer can print up to 20 characters in a line. Additional print instructions may be issued after the head has printed the last character and has encountered the first carriage return, but before the printer control electronics has had time to reverse the lead screw driving the head and begin it on its trip home. For example, a typical line of print that contains both a four-character mnemonic on the right edge and a 15-character numeral field on the left can be printed using the flow of logic as shown in Fig. 4.

**Thermal Printer**

The HP-91 printer is a moving head, thermal printer that prints alphanumeric characters using a 5×7 dot matrix. It prints from right to left (right justified) and advances paper on the return stroke. Lines may contain from 8 to 20 characters; the short line capability gives more throughput for most applications.

The HP-91 printer prints 20-character lines at less than one line per second and eight-character lines at 1.5 lines per second. As the printhead moves from right to left it prints only the number of characters required before returning to the home position. An advantage of the relatively slow line speed is a reduction in the instantaneous power required; this is important in a battery-powered calculator.

Thermal printing was chosen because the mechanism is simpler and lighter than an impact printer. In fact, an impact printer capable of printing alphanumeric characters, and small enough and light enough to fit inside the HP-91, did not exist. Electro-

static and electrosensitive printing do provide simplicity and light weight, but we felt that thermal paper looked better than metallized paper.

The choice of a moving head instead of a fixed head was based on cost. The technologies available within HP for making printheads are thick-film and thin-film. A cost comparison showed that, using either technology, the moving head concept would be less expensive than the fixed head concept. The additional mechanism to move the head had to be kept simple for high reliability and low cost. A simple mechanism design combined with extensive use of injection molded plastic parts helped meet the cost goal.

The choice between thick-film and thin-film technology was based on performance. Above 15 characters per second the thermal mass of the thick-film resistor becomes critical, and the thin-film element becomes attractive. The HP-91 printer operates at 28 characters per second, which means that the head resistors must heat up to several hundred degrees centigrade and cool down below paper threshold temperature in less than 5.1 milliseconds.

Each printed dot is the result of a color reaction in the paper, which is coated with two types of pulverized thermoreaction compounds and a binder. When heat is applied, a chemical reaction takes place that changes the color of the paper at the point of heat.

### Thin-Film Thermal Printhead

The printhead consists of a column of seven 10-ohm thin-film resistors on a ceramic base 0.64 cm wide by 2.0 cm long. When energized by electrical pulses the resistors heat to 350°C. As the head travels across the width of the heat-sensitive paper the pulsed resistors leave a trail of dots that form the seven-dot-high-by-five-dot-wide printed characters.

The printhead had to be mass-produced at high volume. This was another reason for choosing thin-film



$V_B = 4.4V$

$0$

$T_1$

$T_p$

$T_1 = 85\ \mu s$
$T_p = 100\ \mu s$

$V_{RMS} = V_B \sqrt{\dfrac{T_1}{T_p}}$

$= (4.4)\ (.922)$
$= 4.05V$

$V_B = 5.6V$

$0$

$T_1$

$T_p$

$T_1 = 52\ \mu s$
$T_p = 100\ \mu s$

$V_{RMS} = 5.6 \sqrt{\dfrac{52}{100}}$

$= 4.05V$

**Fig. 5.** *Print intensity control compensates for changes in battery voltage by varying the duty cycle of a fixed-frequency signal.*

technology over the thick-film approach.

The printhead substrate is 99.5% aluminum oxide ceramic. There are four layers on the substrate: an outer protective "wear-layer" of aluminum oxide, an aluminum conducting film for interconnections, a tantalum-aluminum resistive film forming the print-dot, and finally, a glaze that acts as a thermal barrier between the resistor and the ceramic body of the printhead. The three thin-film layers—the wear-layer, the aluminum, and the tantalum-aluminum—are all deposited by sputtering. Argon ions formed in a low-pressure glow-discharge are accelerated toward a cathode composed of the material to be deposited, e.g. aluminum. This ion-bombardment causes material to be ejected or sputtered from the cathode and deposited on the substrate. This technique can readily be controlled to deposit films with reproducible composition, resistivity, and thickness. The resistor and conductor films are patterned by photolithographic techniques similar to those used to fabricate integrated circuits.

The glaze underlying the print-dot must be thick enough to permit the resistor to retain heat and develop the paper but thin enough to allow rapid resistor cool-down to avoid elongation or "smearing" of the printed dot. This thickness proved to be about 60 micrometers. Since each of the seven print dots undergoes, on the average, two 300°C temperature cycles per character printed, the wear-layer, resistor film, and glaze must have compatible thermal-expansion coefficients. In addition, the wear-layer must be chemical and abrasion resistant to protect the relatively thin (0.3-micrometer) resistor film. The interconnections, one common and one for each of the seven resistors, were required to add less than 0.75 ohm; this dictated a 4-micrometer-thick aluminum film.

The resistive elements are on a mesa about 10 $\mu$m high. Tests have shown that this provides better contact with the paper, creating a better dot image at lower power and lower head force.

### Printer Drive Circuits

One of the challenges in developing the HP-91 was to find efficient and inexpensive ways to control the motor speed and print intensity. A motor speed control circuit is necessary to keep the length of the printed line constant even though changes occur in moving friction and in the motor supply voltage (the battery). The speed control technique used is a combination of analog and digital.

The motor drive circuit is a seven-transistor circuit that drives the motor forward, reverses it, or brakes its rotation. The brake capability stops the reverse motion quickly when the printhead carriage reaches home position. This permits narrow margins and
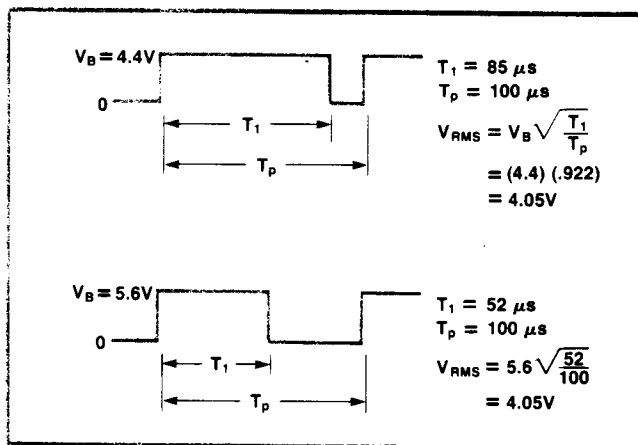
13

maximizes the printed line length.

The motor speed control circuit samples the generated voltage of the drive motor by turning the drive power off briefly and treating the motor as a generator. The measured voltage, which is proportional to the speed of the motor, is compared to a constant reference. If the measured voltage is greater than the reference voltage, the motor is turning too fast and power to the motor is removed until the generated voltage falls below the reference voltage. If the measured voltage is less than the reference voltage, the motor is turning too slowly, so power to the motor is restored immediately after the sample time. The sample rate is fast (several hundred samples per second) and the motor mechanical response time is much longer than the sample period, so the motor smooths out its pulsed input power and turns at a steady speed. The speed may be changed by changing the value of the reference voltage.

To maintain a uniform print contrast, the thin-film head resistors must develop approximately the same temperature each time they are energized. The same temperature must be attainable even with changes in battery voltage. A constant resistor temperature profile requires an applied voltage with a constant rms value. The print intensity control, or power regulator, generates the required voltage by varying the duty cycle of a fixed frequency signal in a nonlinear manner in response to changes in the battery voltage (see Fig. 5).

Generation of the signal is by analog computation. The duty cycle for constant rms value $V_{RMS}$ with battery voltage $V_B$ can be computed as follows.

$$V_{RMS}^2 = \frac{1}{T_p} \int_0^{T_1} V_B^2 \, dt \qquad (1)$$

or

$$V_{RMS}^2 = \frac{T_1}{T_p} V_B^2 \qquad (2)$$

So, duty cycle equals

$$\frac{T_1}{T_p} = \left( \frac{V_{RMS}}{V_B} \right)^2 = \frac{K_o}{V_B^2} \qquad (3)$$

or since $T_p$ is constant,

$$T_1 = \frac{K}{V_B^2} \quad \text{or} \quad T_1 = F(V_B^{-2}) \qquad (4)$$
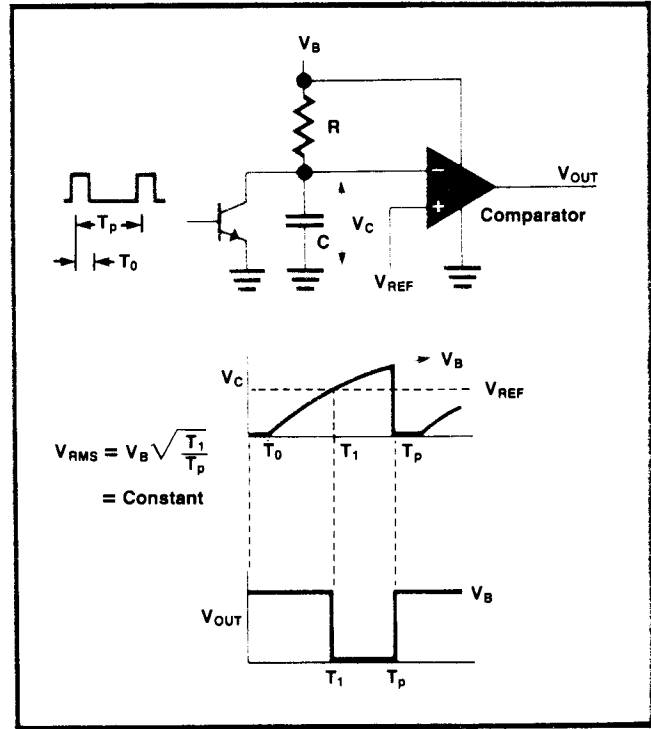


**Fig. 6.** *Simplified diagram of the circuit that monitors the battery voltage and varies the duty cycle of the print signal to stabilize its energy content.*

This function is generated by a circuit that monitors $V_B$ and supplies a time $T_1$ in response to it. A simplified diagram is shown in Fig. 6. By careful selection of $T_0$, R, C, and $V_{REF}$, the function of Equation 4 is evaluated to within one percent.

The frequency of this variable duty cycle signal is about 10 kHz. The signal modulates the applied resistor voltage, as shown in Fig. 7, but because of the high rate of modulation, the head resistor temperature profile does not follow the instantaneous signal but responds to the rms value or heating value of the signal.

Since the power control is basically a pulse modulation scheme, it is very efficient and well suited to
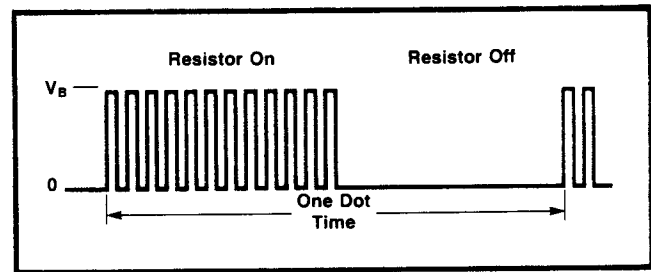


**Fig. 7.** *The variable duty-cycle signal has a frequency of about 10 kHz. This signal modulates the voltage applied to the resistors on the thermal printhead. The resistor temperature responds to the rms or heating value of the composite signal.*

14

battery operation. The actual head drivers are seven NPN high-gain transistors. These receive timing and modulation information from the PICK (printer interface chip). Fig. 8 shows a block diagram of the motor and head control. The PICK chip supplies the timing to control the printer functions.

## Printer Mechanism

The printer mechanism consists of three major sub-assemblies: the paper feed assembly, the printhead/platen assembly, and the printhead drive assembly.

The paper feed assembly uses a cam/clutch arrangement with polyurethane drive rollers for feeding paper. The paper is pinched against the drive roller by two spring-loaded pinch rollers that require no adjustments.

Printing is done "on the fly." To simplify the mechanism, the printhead is always in contact with the paper. The printhead is mounted in a carrier that is threaded onto a lead screw and guided by two stainless steel rods set in a plastic mainframe. The lead screw is driven by an ironless d.c. motor through a gear reduction. Gears are molded out of polycarbonate, with glass and teflon to hold tolerances to AGMA-8 quality and for noise reduction. The ironless dc motor is used for its size, high efficiency, and low inertia.

The printhead moves back and forth against a platen. The platen, mounted in a platen holder assembly, is fully floating and is spring loaded against the printhead to assure proper contact.
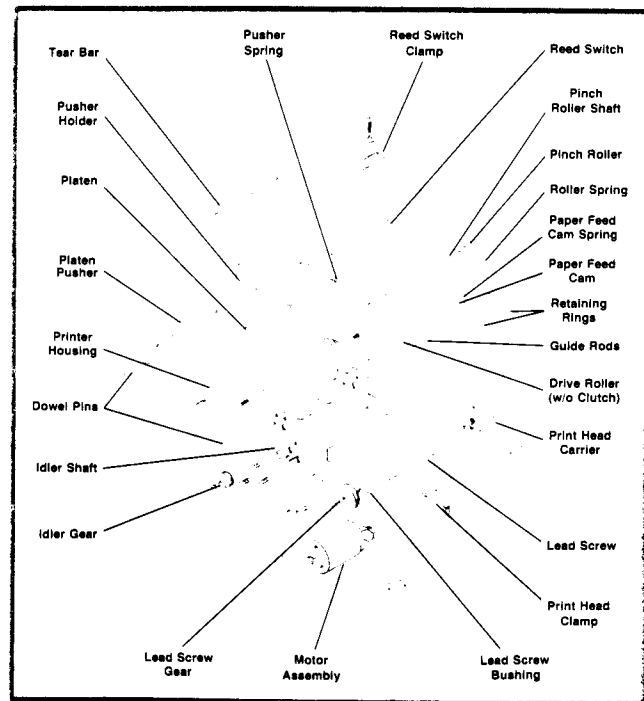
The extensive use of plastics made this mechanism



**Fig. 9.** *Printer mechanism.*

a real challenge for both the mechanical designers and the injection-molding tool designers. To save assembly time, the entire printer frame is molded in one piece with some tolerances held to ±25 $\mu$m. The head carrier must withstand high temperature and wear, and must be held to tight parallelism tolerances with a threaded hole for the drive screw. The rubber drive rollers are made with a double-shot process of injection-molded urethane over an ABS hub. The platen is made of an injection-molded, high-temperature, ceramic-like plastic called PPS. Much development time was spent on the platen to assure high quality. If the printer should run out of paper, the printhead will print directly on the platen. Since some parts of the printhead can reach 350°C and more, this presents a severe challenge for plastic. By solving the problem with injection-molded plastic, a low-cost but high-quality platen was achieved. The platen is designed to withstand direct continuous printing without paper for over 72 hours.

The printhead resistors are driven with the battery voltage of approximately five volts. To reach printing temperature in three milliseconds requires about 1.1 watts, which means that over 500 milliamperes must flow into each printing resistor. If four or five dots are printed at the same time, the current level can approach 2.5 amperes in the common lead of the head and the flexible cable. This created some difficult problems in the circuit design and in finding a flexible cable to pass that much current. The flexible cable conductors must be wide and thick to avoid
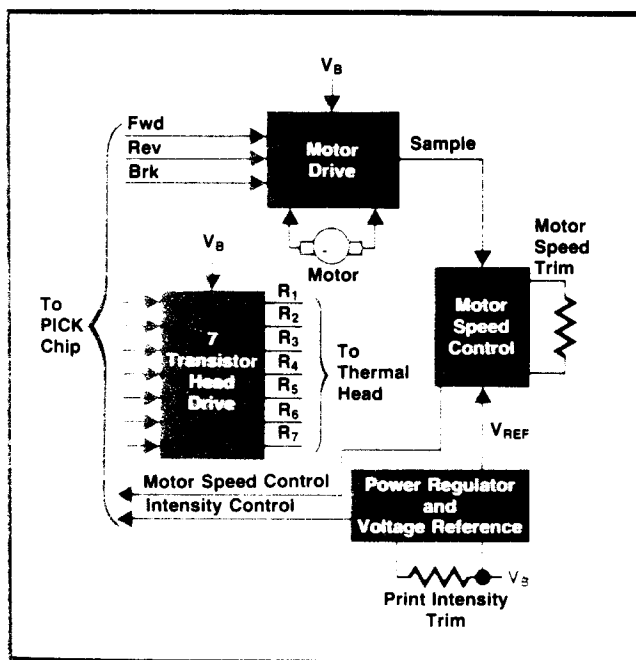


**Fig. 8.** *Motor and head control circuits. Timing information comes from the PICK chip.*

15

high resistive losses, but the thicker the copper, the more difficult it is to make it flex for millions of cycles without stress fatiguing in the tight bend radius required by the small package. The cable materials and adhesives all affect the fatigue life of the copper. After many iterations and extensive testing, a Kapton-copper-Kapton sandwich laminated with special adhesive was selected.

Because the cable connectors are copper and the printhead interconnects are thin-film aluminum, conventional solder bonding was not possible. Ultrasonic-wire or stitch bonding provided the answer.

# The New Accuracy: Making $2^3 = 8$ *

## by Dennis W. Harms

One of the engineering efforts that made the HP-35 possible was the development of the algorithms for the mathematical functions. This work was done by David S. Cochran.[1] These algorithms have served the HP hand-held calculator family well, but have now been replaced by more accurate ones on the HP-91. As one of the more obvious examples of the improvement, $2^3 = 8.000000003$ on the HP-35 but $2^3 = 8000000000$ on the HP-91. Other examples of the improvement are shown in the table below:

| OPERATION | HP-35 | HP-91 |
|---|---|---|
| $10^{52}$ | 9 99999 9964 × $10^{51}$ | 1 00000 0000 × $10^{52}$ |
| $25^5$ | 97656 25 056 | 97656 25 000 |
| $9^5$ | 59049 00020 | 59049 00000 |
| $15^5$ | 759375 0034 | 759375 0000 |
| $(-2)^3$ | error | -8 00000 0000 |
| ln (1.00001) | 9 999900000 × $10^{-6}$ | 9 99995 0000 × $10^{-6}$ |
| ln (.99995) | -5 00012 0000 × $10^{-5}$ | -5 00012 5004 × $10^{-5}$ |
| tan 89.999999° | 58363500 16 | 57295779 51 |
| tan 90.000001° | -58823529 41 | -57295779 51 |
| sin .000 0000001° | 1 7 × $10^{-9}$ | 1 745329252 × $10^{-9}$ |
| sin 359.9999999° | -2 184413879 | -1 745329252 × $10^{-9}$ |
| tan 720° | 4 00000 0000 × $10^{-9}$ | 0 00000 0000 |
| tan (720 × $10^{90}$)° | 2 02975 2090 | 0 00000 0000 |

The overriding consideration in the HP-35 was saving ROM (read-only memory). Only three ROMs of microcode were used to implement all the functions of the calculator. Better accuracy, comparable to that of the HP-91, would have consumed two additional ROMs and would have increased both the size and cost of the HP-35. Since the slide rule was the standard of the day, the

*This discussion is essentially the same as a paper presented by the author at Electro 76 in Boston. Massachusetts. U.S.A., May 11-14, 1976.

improvement in accuracy bought by these additional ROMs did not seem worth the expense.

Two things have happened to make the additional ROM worth the cost. First, ROM has become cheaper and smaller by a factor of four. Second, calculators are now being applied more widely than originally anticipated, and users now have a higher level of sophistication and expect the same from their calculators. Programmable calculators, especially, require greater accuracy. The user of a nonprogrammable calculator monitors his calculation step by step, so unreasonable intermediate results caused by algorithm inadequacies are recognized when they appear. However, when a calculator is programmable, the user is often aware of only the input and the final answer.

### Avoidable and Unavoidable Errors

Displayed errors can be divided into two types, avoidable and unavoidable. The first is caused by inadequacies in the algorithms. Examples of this type of error are shown in the table above and are typified by $2^3 = 8.000000003$. This type of error has been almost completely eliminated from the HP-91.

The second type of error is the unavoidable error caused by using finite computers to approximate nonfinite processes. For instance, the decimal representation of 1/7 is the repeating nonfinite decimal:

$$1/7 = 0.142857\ 142857\ 142857 \dots$$

which cannot be represented exactly with ten figures. The closest approximations are:

0.142857 1429 (closest) and
0.142857 1428 (next closest).

When we write
$$1/7 = 0.142857\ 1429$$
we are lying only by a little bit. Unfortunately,
$$1/(0.142857\ 1429) = 6.99999\ 99979\ 00000\ 00062\ 99...$$
which rounds to 6.99999 9998 instead of 7. Therefore, the calculator cannot satisfy the identity
$$1/(1/7) = 7.$$
This is not a mistake. It is an unavoidable consequence of rounding.

There is no ten-figure approximation to 1/7 whose reciprocal rounds to 7. To see this, let us look at the next closest approximation to 1/7.
$$1/(0.142857\ 1428) = 7.00000\ 00028\ 00000\ 00112 ...$$
which rounds to 7.00000 0003 instead of 7. This is farther from 7 than the reciprocal of the other approximation to 1/7.

Perhaps the best way to avoid the confusing statement
$$\text{"}1/(1/7) = 6.99999\ 9998\text{"}$$
is to write
$$[1/(1/7)_{10}]_{10} = 6.99999\ 9998$$
where the subscript 10 indicates a rounding to 10 places.

## How Improvements Were Achieved

Accuracy in the HP-91 has been improved principally by rewriting the algorithms' subroutine structure. An example of this is with the $y^x$ routine. All HP calculators calculate $y^x$ by the formula
$$y^x = e^{x\ \ln y}$$
This is evaluated by subroutine calls to ln, multiply, and exponential. On the HP-35 these subroutines were identical to the keyboard functions of the same name. The function $y^x$ did the same thing as the key sequence
$$x \rightleftharpoons y, \ln, x, e^x$$
It is not surprising then that $2^3 = 8.000000003$, for this answer has been subject to three intermediate roundings. The improvement was made by rewriting the basic subroutines to allow for carrying intermediate results to extra digits. Enough extra digits are carried and carefully looked after to assure the desired accuracy. Special subroutines of this type are also used in the trigonometric functions, in rectangular-to-polar conversions, and in polar-to-rectangular conversions.

The second method of improving accuracy is to trap critical arguments and calculate the functions at these arguments in a special way. These critical arguments include numbers near 1 when calculating ln or log, numbers near 0 when calculating $\sin^{-1}$, $\cos^{-1}$, or $\tan^{-1}$, and numbers near zero or multiples of $\pi/2$ when calculating sin, cos, or tan.

Another improvement was made by prescaling the trigonometric functions in the user's units. The HP-35 converted degrees to radians and then subtracted multiples of $2\pi$ to convert the input argument to the unit circle. This made tan 720° = $4 \times 10^{-9}$. On the HP-91, if the calculator is in degrees or grads, multiples of 360 or 400 respectively are subtracted until the result is within the unit circle. This makes tan 720° = 0 and improves symmetry properties of the trigonometric functions.

## Possible Improvements

There is a point where the new algorithms appear at first glance to be less accurate. If we calculate tan $\pi$ in radians, the answer is $4.10 \times 10^{-10}$ on the HP-91 whereas the HP-55 yields 0. This is misleading, for $\pi$ on the keyboard should really be labeled $\pi_{10}$, showing that it is ten digits of $\pi$. Thus
$$\pi_{10} = 3.14159\ 2654.$$
To ten digits, tan $\pi_{10} = 4.12067\ 6150 \times 10^{-10}$. One can see that the HP-91 gives the better answer to this problem.

The reason that the HP-91 does not give the correct answer is that prescaling is done with a finite-length $\pi$. To give the correct answer to this problem, a $\pi$ of length 20 would have to be used. It can be shown that to produce the correct answer to all trigonometric problems in radian measure, it takes a $\pi$ of length 120. The HP-91 actually uses 13 digits of $\pi$.

Another place where the accuracy can be improved is in the statistical functions. If the following data is put into the HP-55 using the $\Sigma+$ key, the standard deviation is not calculated exactly.
$$6666666123$$
$$6666666246$$
$$6666666369$$
The standard deviation calculated is 0.000000000 instead of the correct value of 123.0000000. This problem could be solved by storing different sums in the statistics registers. However, the sums that are now stored are useful for vector arithmetic.

The user can usually avoid this problem by leaving off the redundant leading sixes. If the data is keyed in as follows, the correct standard deviation is calculated.
$$123$$
$$246$$
$$369$$

## References

1. D.S. Cochran, "Algorithms and Accuracy in the HP-35," Hewlett-Packard Journal, June 1972.
2. R.B. Neff and L. Tillman, "Three New Pocket Calculators: Smaller, Less Costly, More Powerful," Hewlett-Packard Journal, November 1975.
3. P.H. Sterbenk, "Floating Point Computation", Prentice Hall, 1974.

**Dennis W. Harms**
Dennis Harms was born, grew up, and attended college in Waverly, Iowa, graduating from Wartburg College in 1969 with a BS degree in mathematics. He went on to receive his MS and PhD degrees from Iowa State University in 1970 and 1974, teaching undergraduates calculus and finite mathematics while pursuing his own studies. He joined HP in 1973, wrote many applications programs for the HP-65 and HP-55, did microprogramming for the HP-91, and improved the algorithms used in HP hand-held calculators. He's a member of the American Mathematical Society. Dennis relaxes by playing duplicate bridge, basketball, and tennis. He's married to a librarian and lives in Santa Clara, California.

# FEATURES AND SPECIFICATIONS
## HP-91 Scientific Printing Calculator

**PRINTER FEATURES:**

| | |
|---|---|
| PRINT MODE | Selects printing modes. |
| SWITCH | You can print nothing, digit entries and functions, or digit entries, functions and results. |
| PRINT X | Prints the displayed number. |
| LIST STACK | Prints and labels the contents of the stack registers. |
| LIST $\Sigma$ | Prints and labels the contents of the summation registers. |
| LIST REG | Prints and labels the contents of the addressable registers. |
| PAPER ADVANCE | Advances the paper. At the same time clears errors and halts list operations. |

**MATHEMATICAL FUNCTIONS:**

| | |
|---|---|
| ANGULAR MODE SWITCH | Selects degrees, radians or grads mode. |
| Sin, Cos, Tan | Trigonometric functions |
| $\text{Sin}^{-1}$, $\text{Cos}^{-1}$, $\text{Tan}^{-1}$ | and their inverses |
| ln, $e^x$ | Natural log and antilog |
| log, $10^x$ | Common log and antilog |
| $y^x$ | Exponential function |
| $\rightarrow$R, $\rightarrow$P | Rectangular polar coordinate conversion |
| H.MS +, H.MS - | Time (angle) addition and subtraction in hours, minutes and seconds and hours |
| $\rightarrow$H.MS, H.MS$\rightarrow$ | Time (angle) conversion between hours, minutes and seconds and hours |
| $1\div x$, $\sqrt{x}$, $\pi$, $x^2$ | Convenient math functions |
| +, -, ×, ÷ | Arithmetic functions |
| % | Percent |
| $\Delta$% | Percent difference |
| %$\Sigma$ | Percent of total |

**STATISTICAL FUNCTIONS:**

| | |
|---|---|
| $\Sigma+$ | Accumulates x, y, $x^2$, $y^2$, xy, n |
| $\Sigma-$ | Deletes unwanted data |
| $\bar{x}$ | Mean—two variables |
| s | Standard deviation—two variables |
| L.R. | Linear regression coefficients |
| $\hat{y}$ | Linear estimate |
| n! | Factorial |

**DATA MANIPULATION, OPERATIONS AND STORAGE FUNCTIONS FOR 16 REGISTERS:**

| | |
|---|---|
| ENTER* | Separates numerical entries |
| x≒y | Exchanges contents of x and y stack registers |
| R$\downarrow$ | Rolls down contents of stack registers |
| R$\uparrow$ | Rolls up contents of stack registers |
| CHS | Changes the sign of a number |
| EEX | Enters exponent for numbers entered in scientific notation |
| LAST x | Recalls last entry after an operation |
| ■ | Gold shift key, selects functions printed in gold on keyboard |
| STO | Stores displayed value into addressable register. Also used with arithmetic functions in storage register arithmetic. |

| | |
|---|---|
| RCL | Recalls a number to the display from an addressable register |

**DISPLAY CONTROL AND CLEARING OPTIONS:**

| | |
|---|---|
| FIX | Sets fixed decimal notation display |
| SCI | Sets scientific notation display |
| ENG | Sets engineering notation display |
| CLx | Clears display |
| CLEAR | Clears stack, addressable registers, and Last x register |
| CL REG | Clears addressable registers $R_0$-$R_9$ |
| CL$\Sigma$ | Clears summation registers $R_0$-$R_5$ |

**PHYSICAL SPECIFICATIONS:**
CALCULATOR WIDTH: 228.6 mm (9 in)
CALCULATOR DEPTH: 203.2 mm (8 in)
CALCULATOR HEIGHT: 63.5 mm (2.5 in)
CALCULATOR WEIGHT: 1.13 kg (2.5 lb)
RECHARGER WEIGHT: 268 g (9.5 oz)
SHIPPING WEIGHT: 3.16 kg (7 lb)
OPERATING TEMPERATURE RANGE: 0 C to 50 C (32 F to 122 F)
CHARGING TEMPERATURE RANGE: 10 C to 40 C (50 F to 104 F)
STORAGE TEMPERATURE RANGE: -40 C to 55 C (-40 F to 131 F)
PAPER TEMPERATURE RANGE (operating and storage): 10 C to 45 C 50 F to 113 F)

**POWER REQUIREMENTS:**
AC: 115 or 230 V, ±10%, 50 to 60 Hz
BATTERY: 5.0 Vdc nickel cadmium rechargeable battery pack
**PRICE IN U.S.A.:** HP-91: $425.
**MANUFACTURING DIVISION:** CORVALLIS DIVISION
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 U.S.A.

# FEATURES AND SPECIFICATIONS
## HP-67 and HP-97 Calculators

### Keyboard Features Summary
**ANGULAR FUNCTIONS:** Sin, Cos, Tan, $\text{Sin}^{-1}$, $\text{Cos}^{-1}$, $\text{Tan}^{-1}$ Hours-Minutes-Seconds Addition; Hours-Minutes-Seconds Conversion to Decimal Hours; Degree Radian Conversion; Polar Rectangular Conversion; Degrees, Radians, Grads Angular Modes.
**LOGARITHMIC FUNCTIONS:** Log, $10^x$, Ln, $e^x$
**STATISTICS:** Summations n, $\Sigma x$, $\Sigma x^2$, $\Sigma y$, $\Sigma y^2$, $\Sigma xy$, Deletion of Unwanted Data; Mean, Standard Deviation
**CONVENIENT MATH FUNCTIONS:** +, -, ×, ÷, $y^x$, $x^2$, $1\div x$, $\sqrt{x}$, N!, %, %CH, $\pi$
**NUMBER ALTERATION FUNCTIONS:** Integer Truncation, Fraction Truncation, Absolute Value, Rounding.
**DISPLAY:** Fixed decimal, scientific, engineering notation. Displays up to 10 significant digits plus 2-digit exponent and appropriate signs.

### HP-97 Printing Features
**PRINT MODE SWITCH:** Selects printing modes. You can print only when you want to, digit entries and functions automatically, or digit entries, functions, intermediate, and final answers automatically.
**PRINT X:** Prints the displayed number.
**PRINT STACK:** Prints and labels the contents of the stack registers.
**PRINT REG:** Prints and labels the contents of the primary data storage registers.

**PRINT PRGM:** Prints contents of program memory.
**SPACE:** Advances paper one space without printing.

### HP-67 and HP-97 Common Programming Features Summary
**CAPACITY:** 224 steps of program memory (all functions are merged and occupy only one step of program memory); 26 data registers, four register automatic-memory-stack, Last x register
**CARD READER FEATURES:** Record Load all data registers, Load selected data registers, Record/Load entire program memory, Merge program subsections. Angular mode, flag settings, and display status are recorded with program recording and reset with program loading. User is prompted for proper operation when loading. Card reader operations can be initiated manually or under program control (except program recording).
**ADDRESSING:** Label addressing. Indirect addressing of labels and data storage. Relative addressing; 10 user-definable keys or 20 user-definable labels. Three levels of subroutines (GSB)
**CONDITIONALS:** x=y, x≠y, x>y, x≤y, x≠0, x=0, x>0, x<0. Four flags. Increment/decrement storage registers and skip on zero
**EDITING:** Single step execution. Single step and back step inspection of a program. Insert/Delete editing. Position the calculator at any step in program memory (GTO. nnn)
**OTHER PROGRAMMING FEATURES:** PAUSE to review intermediate results. key in data, or load magnetic cards.

**HP-97 SPECIFICATIONS:**
WIDTH: 228.6 mm (9 in)
DEPTH: 203.2 mm (8 in)
HEIGHT: 63.5 mm (2.5 in)
WEIGHT: 1.13 kg (2.5 lb)
RECHARGER WEIGHT: 268 g (9.5 oz)
SHIPPING WEIGHT: 3.16 kg (7 lb)
OPERATING TEMPERATURE RANGE: 15°C to 40°C (59°F to 104°F)
CHARGING TEMPERATURE RANGE: 10°C to 40°C (50°F to 104°F)
STORAGE TEMPERATURE RANGE: -40°C to 55°C (-40°F to 131°F)
PAPER TEMPERATURE RANGE: 10°C to 45°C (50°F to 113°F)
AC POWER REQUIREMENT: 90-127V or 200-254V, 50 to 60 Hz
BATTERY PACK REQUIREMENT: 5.0 Vdc nickel cadmium rechargeable battery pack

**HP-67 SPECIFICATIONS:**
LENGTH: 152.4 mm (6 in)
WIDTH: 81 mm (3.2 in)
HEIGHT: 18 to 34 mm (0.7 to 1.4 in)
WEIGHT: 342 g (11 oz)
RECHARGER WEIGHT: 142 g (5 oz)
SHIPPING WEIGHT: 1.4 kg (3 lb)
OPERATING TEMPERATURE RANGE: 10°C to 40°C (50°F to 104°F)
CHARGING TEMPERATURE RANGE: 10°C to 40°C (50°F to 104°F)
STORAGE TEMPERATURE RANGE: -40°C to 55°C (-40°F to 131°F)
AC POWER REQUIREMENT: 86-127V or 172-254V, 50 to 60 Hz
BATTERY POWER REQUIREMENT: 3.75 Vdc nickel cadmium rechargeable battery pack
**PRICES IN U.S.A.:** HP-67, $450. HP-97, $750.
**MANUFACTURING DIVISION:** CORVALLIS DIVISION
1000 N.E. Circle Boulevard
Corvallis, Oregon 97330 U.S.A.

### Bernard E. Musch
Bernie Musch received his BSME degree in 1964 from Lehigh University and his MSME and PhD degrees in 1966 and 1970 from Stanford University. After coming to HP in 1970, he did the mechanical design of the HP-35 and HP-65 Calculators and served as project leader for the HP-55 and HP-91 Calculators. He has several patents, issued and pending, in the calculator field. Having just moved to Corvallis, Oregon last July, Bernie hasn't had time to settle down yet, but before the move he served his community by coaching Little League and soccer teams and by recording books for the blind. Camping, photography, and music are other major interests. A native of Baltimore, Maryland, Bernie's married and has two sons.

### Robert B. Taggart
Bob Taggart received his BSME and MSME degrees from Northwestern University in 1967 and 1968, and the degree of Mechanical Engineer from Stanford University in 1970. With HP since 1970, he worked on biomedical and analytical instrument projects for three years, then served as project leader for the HP-65 card reader and project manager for printing calculators. Ten professional papers and five patents have resulted from his work. Born in Pompton Lakes, New Jersey, Bob is married, has a small son, and now lives in Sunnyvale, California. Tennis and skiing are his favorite leisure activities.