HEWLETT-PACKARD JOURNAL

More Memory For Desktop Calculators

Extended memory unit adds as many as 3472 program steps.

By Russell Sparks

ADDITIONAL MEMORY FOR THE HP MODEL 9100A/B DESKTOP CALCULATORS is provided by a new, self-contained, rack-mounted unit. By adding 17 times the programmable memory to the Model 9100A, and 8 times the programmable memory to the Model 9100B, this Model 9101A Extended Memory, Fig. 1, greatly extends the range of their problem solving abilities. The calculators are still the controlling and calculating elements of the system; the extended memory is operated by Format (FMT) commands from the calculators. It has a 20,832 bit memory capable of storing up to 3472 program steps, or 248 fourteen-digit registers for data storage.

When used with other peripherals, the Model 9101A becomes the main data and program storage unit. With appropriate interface cards in the Model 2570A Coupler,[1] the calculators are able to feed data to, or accept inputs from teleprinters and other equipment.

To simplify programming, register addressing and program selection are in decimal numbers rather than binary, octal or hexadecimal systems. The programmer need not be concerned with internal addressing. He needs only to remember corresponding program numbers and their functions. Address information important to the programmer is displayed when programs are transferred. Diagnostic codes are shown whenever there is a program error related to the 9101A operation.

Basic Features

The HP 9101A extended memory is divided into 248 fourteen-digit registers (ten digits displayed, two digits overrange, and two digits exponent). The registers of the 9101A are numbered decimally from 0 through 247, and are always addressed from the 9100A/B X-register.

Programs stored in the 9101A are transferred from the 9100A/B, starting at location 00 in the 9100A and +00 in the 9100B; and stopping when an END statement is encountered in the program. To identify the program, a two-digit number (decimal number 00 through 99) is placed in the 9100A/B X-register prior to the transfer. To assist the programmer in making maximum use of the 9101A storage, any new program added in storage is stored starting with the first available register, incrementing registers sequentially (14 program steps per register as in the 9100A/B) until the END statement is reached.
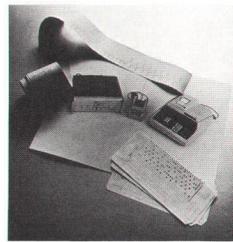
Cover: *Marked cards, lower right, are used to enter programs and data in the Model 9100 Calculator System through the Model 9160A Card Reader, center. Hard copy output, upper left, is produced by the Model 9120A Printer, mounted on the calculator. The HP Model 9101A Extended Memory, left, greatly increases the computing power of the system.*

2

Fig. 1. *This Hewlett-Packard Model 9101A Extended Memory attaches directly to the Model 9100A/B Calculators. It adds 248 registers of memory greatly increasing the computing power of the calculators.*

This gives maximum program packing in the core memory within a register each time.

Each program is started in a new register. The availability of registers for storage is defined by a protection flag on each register. This protection flag is automatically set when a program is stored and may be moved manually to any desired register by another format command. (A program stripped of its protection is no longer recognized as a program.) Thus, sequential storing of all programs into the 9101A results in the most efficient use of registers, with the maximum area open for data storage. The user need not be concerned with locating each program.

Location of programs is, however, often of interest for future editing, and, in the case of the last program stored, it indicates the boundary where data can be stored. The 9101A returns the last register address used in a given

program to the X-register of the 9100A/B whenever a program is stored into or recalled from the 9101A Extended Memory.

Subprograms which are subject to repeated calls from another program are often useful. Since all programs are executed in the 9100A/B Calculators, this requires that the calling program be destroyed in the calculator in order to execute the subprogram and then restored into the 9100A/B after the subprogram has finished its execution. This requires that all programs, including main calling programs, be assigned a number and stored in the 9101A. A return vector is kept in the 9101A which retains the information as to what the calling program number was and remembers where to return in that program to resume executing it. Subprograms can call other subprograms if desired (up to 14 subprograms in one
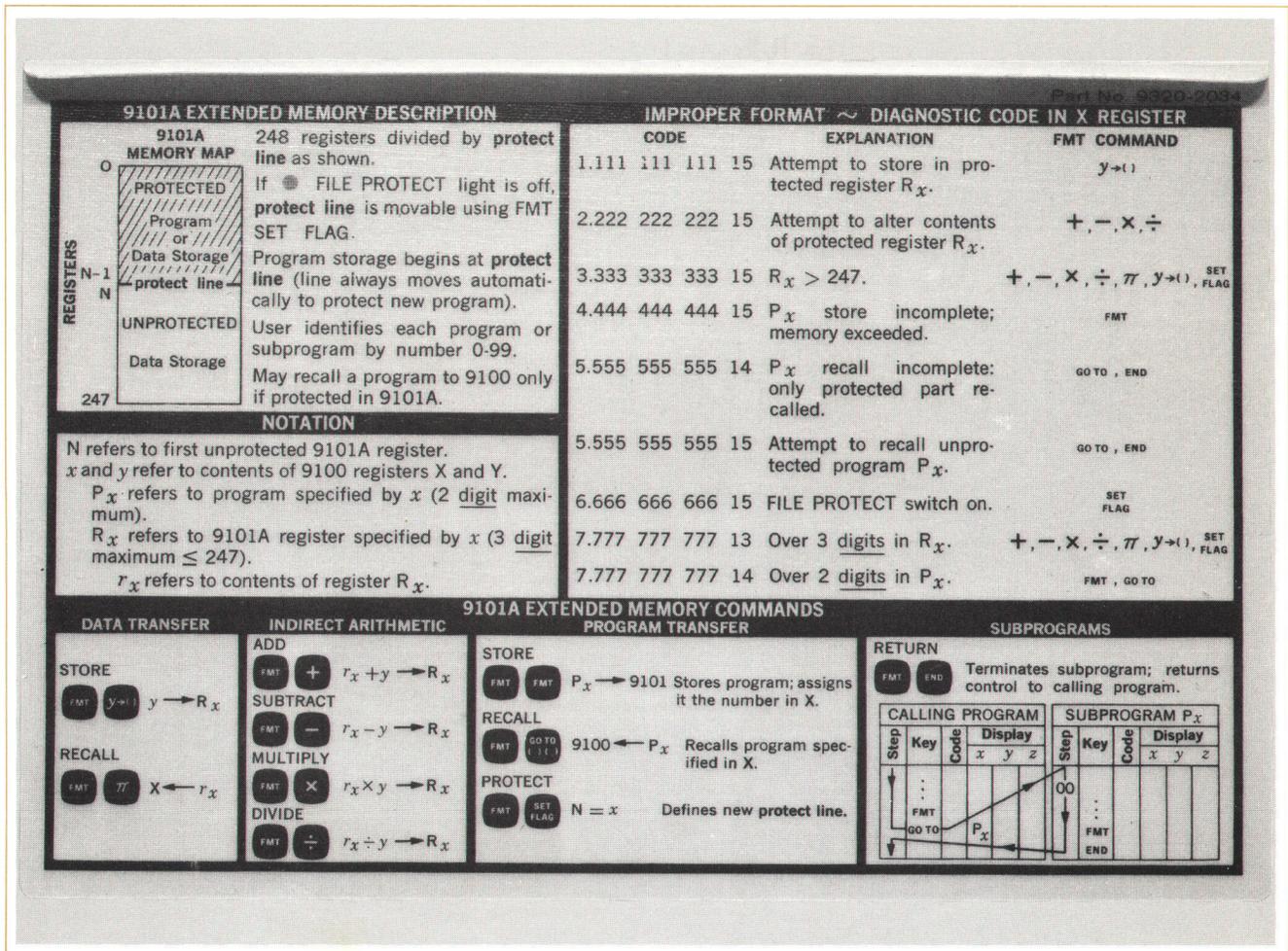
9101A EXTENDED MEMORY DESCRIPTION

9101A MEMORY MAP

PROTECTED
Program or Data Storage
protect line
UNPROTECTED
Data Storage

REGISTERS: 0, N−1, N, 247

248 registers divided by **protect line** as shown.

If ⬤ FILE PROTECT light is off, **protect line** is movable using FMT SET FLAG.

Program storage begins at **protect line** (line always moves automatically to protect new program).

User identifies each program or subprogram by number 0-99.

May recall a program to 9100 only if protected in 9101A.

NOTATION

N refers to first unprotected 9101A register.
x and y refer to contents of 9100 registers X and Y.
  P_x refers to program specified by x (2 digit maximum).
  R_x refers to 9101A register specified by x (3 digit maximum $\leq$ 247).
  r_x refers to contents of register R_x.

IMPROPER FORMAT ~ DIAGNOSTIC CODE IN X REGISTER

CODE	EXPLANATION	FMT COMMAND
1.111 111 111 15	Attempt to store in protected register R_x.	$y \rightarrow ()$
2.222 222 222 15	Attempt to alter contents of protected register R_x.	$+, -, \times, \div$
3.333 333 333 15	$R_x > 247$.	$+, -, \times, \div, \pi, y \rightarrow ()$, SET FLAG
4.444 444 444 15	P_x store incomplete; memory exceeded.	FMT
5.555 555 555 14	P_x recall incomplete: only protected part recalled.	GO TO, END
5.555 555 555 15	Attempt to recall unprotected program P_x.	GO TO, END
6.666 666 666 15	FILE PROTECT switch on.	SET FLAG
7.777 777 777 13	Over 3 digits in R_x.	$+, -, \times, \div, \pi, y \rightarrow ()$, SET FLAG
7.777 777 777 14	Over 2 digits in P_x.	FMT, GO TO

9101A EXTENDED MEMORY COMMANDS

DATA TRANSFER

STORE: FMT $y \rightarrow ()$ $y \rightarrow R_x$

RECALL: FMT π $X \leftarrow r_x$

INDIRECT ARITHMETIC

ADD: FMT $+$ $r_x + y \rightarrow R_x$

SUBTRACT: FMT $-$ $r_x - y \rightarrow R_x$

MULTIPLY: FMT $\times$ $r_x \times y \rightarrow R_x$

DIVIDE: FMT $\div$ $r_x \div y \rightarrow R_x$

PROGRAM TRANSFER

STORE: FMT FMT $P_x \rightarrow$ 9101   Stores program; assigns it the number in X.

RECALL: FMT GO TO   9100 $\leftarrow P_x$ Recalls program specified in X.

PROTECT: FMT SET FLAG $N = x$ Defines new **protect line**.

SUBPROGRAMS

RETURN: FMT END   Terminates subprogram; returns control to calling program.

CALLING PROGRAM			SUBPROGRAM P_x		
Step	Key	Code / Display x y z	Step	Key	Code / Display x y z
	FMT GO TO	P_x	00		
				FMT END	

Fig. 2. *Instruction card on the Model 9101A front panel is a handy reference for the operator, eliminating the need to keep an operating manual nearby.*

sequence, not including the original calling program). Unless a subprogram is used several times, it is generally more efficient to write one long program and break it into parts which do not require returning to previous parts.

The indirect arithmetic routines allow the programmer to operate on the contents of a 9101A register, to multiply by, divide by, add to, or subtract from those contents a number which was placed in the 9100A/B Y-register. The address of the 9101A register desired is placed in the 9100A/B X-register. The original information is retained in the 9100A/B X- and Y-registers after the operation.

Operation of the 9101A memory is from the 9100A/B Calculators by use of the (FMT) key followed by a second key that represents a command code, Fig. 2. This second key will not be recognized as an operator, but will output a code only; e.g., FMT π gives out octal code 56 only and does not put Pi in the X-register of the calculator.

Command Codes

FMT Y → ( ): Stores 9100A/B Y-register data into 9101A register addressed by the 9100A/B X-register.

FMT π: Recalls into the 9100A/B X-register the data stored in the 9101A register addressed by the number which was originally placed in the 9100A/B X-register.

FMT (+, −, × or ÷): Takes the data stored in the 9101A register addressed by the 9100 X-register and operates on it as dictated by the FMT code (+, −, ×, or ÷), using the data in the 9100A/B Y-register as the operator.

FMT FMT: Transfers the program in the 9100A/B into the 9101A starting from Step 00 in the 9100A/B (+00 in the 9100B) through the END statement in the program. The number in the X-register prior to the transfer is the number which is assigned to that

Error Indication in the Model 9101A Extended Memory

When an error occurs in a program which results in unacceptable information offered to the Model 9101A, two visible indications appear. An IMPROPER FORMAT lamp on the front panel of the Model 9101 turns on, and a diagnostic code number appears in the X-register on the calculator display. Program execution stops when this occurs.

The diagnostic codes are given to help the programmer determine his error. One of these nine codes appears when the IMPROPER FORMAT lamp lights:

CODE	ERROR
1.111 111 111 15	Attempt to store in a protected register.
2.222 222 222 15	Attempt to accumulate in a protected register.
3.333 333 333 15	A nonexistent register address given, that is, larger than 247.
4.444 444 444 15	Insufficient storage remaining for a program.
5.555 555 555 14	Recall of a program to the Model 9100A/B is incomplete. Only protected part recalled.
5.555 555 555 15	Attempt to recall an unprotected program.
6.666 666 666 15	Attempt to move a protected program to a new address while the FILE PROTECT switch is on.
7.777 777 777 13	More than three digits given to address a register in the Model 9101A. (Registers are numbered 0 through 247.)
7.777 777 777 14	More than two digits given to address a program in the Model 9101A. (Programs are numbered 0 through 99.)

Sample Program

Divide each element of a matrix by n, where $0 < n < 9$, and a_n is an element stored in the Model 9101A memory locations 100 through 109 (address $= 100 + n$).

START

INITIALIZE
n = 0, ADDRESS = 100

SUBROUTINE
n!

$$\frac{a^n}{n!} \to a^n$$

INCREMENT
n

LAST ADDRESS? — Yes → END

No

ADDRESS = 100 + n

program. After the transfer is complete, the address of the last register in the 9101A used to store the program is returned to the 9100A/B X-register.

FMT GO TO: Recalls from the 9101A the program identified by the number placed in the 9100A/B X-register. The program is loaded from 00 in the 9100A/B (+00 in the 9100B) up through the END statement. After the program has been transferred, the address of the last register used in the 9101A to store the program is returned to the 9100A/B X-register.

FMT END: Executes the return vector at the end of a sub-program to allow return of the calling program, and to return to the proper place in this program. Provision is made here to allow for the case where the calling 'program' was the operator calling it from the 9100A/B keyboard. In this case, the FMT END is recognized as an END and no further execution is required.

FMT SET FLAG: Allows the operator to set the boundary of protection where desired. The boundary is set such that the register address placed in the 9100A/B is the first unprotected register (all addresses numerically less than this are protected).

Electronics

The basic operation of the 9101A is very similar to that of the 9100A/B where two ROM's (Read Only Memories) are used in control.[2] The first is a diode ROM used as a basic sequencer. The second controls the sequences involved in operations dictated by the selected FMT codes. This is a bipolar transistor ROM in twelve integrated circuit packages.

A buffer for isolating peripherals, the new HP Model 9102A Buffer, is necessary for multiple connection of calculator peripherals to the 9100A/B. A fairly large system can be assembled using the 9101A. Some systems of this kind become susceptible to noise induced in the interconnecting lines within the system. Several precautions were taken in the 9101A to prevent noise from giving false signals. Among the most important of these are careful ground circuit connections and time sampling of the FMT input signal.

Since the 9101A interfaces with the 9100A/B using —15 volt true logic, many of the circuits used are very similar to the 9100A/B. Integrated circuits have, however, been incorporated into the core memory addressing and into the ROM.

Speed is important in a system where programming requires many iterations in arriving at the solution to a problem. The 9101A has speeds compatible with the operations in the 9100A/B. Thus, any single transfer operation (data or program) has speeds comparable to the functions of the 9100A/B (ranging from a few milliseconds to about 100 milliseconds).

One of the unique techniques used to obtain information from the 9100A/B at high speeds is that of picking the digits from the 9100A/B X, Y, and Z registers using the ROLL key. In addition to high-speed data transfer, this operation also transfers the two guard digits (giving twelve significant digits instead of ten) from these registers.

Programming

Whenever large programs are encountered, an area of core memory is often assigned as data storage. In the 9100A/B/9101A system, large programs must be broken into parts, either functionally (sub-programs) or sequentially as segments of a large program. This requires that some area of memory be set aside that will not be altered by the transferring of program segments. If small amounts of data are common to these segments, then parts of the 9100A/B core memory can be assigned to this task and program segments written short enough that no one segment overlaps into the data area. This is possible since the programs are always loaded into the 9100A/B from the 9101A starting at location 00, and stopping at the END statement. If larger common storage is required then a block of addresses in the 9101A is set aside which can be addressed uniquely from any program.

The FMT commands can be given from a program stored in the 9100A/B and executed much in the same manner as any key code of the 9100A/B; when the function has been executed, the program will continue on to the next step. If any of these FMT commands are given from the 9100A/B keyboard, then execution is halted when the function is complete.

One of the problems with large programs is that of debugging the program after it has been written. The 9101A supplies the user with an error indication and a code to aid him in finding any programming problems related to 9101A operations.

File protection has been provided to prevent accidental destruction of protected programs or data. Additionally, there is a 'FILE PROTECT' switch on the front panel that gives further protection by disallowing the use of FMT SET FLAG, thus inhibiting the function which otherwise could remove protection. It is, however, still possible to store new programs, which are automatically protected, when this switch is on.

SPECIFICATIONS
HP Model 9101A
Extended Memory

DESCRIPTION

The HP 9101A is an extended memory unit offering expanded storage capacity for the HP 9100 family of calculators. When attached to any 9100A/B output connector, the 9101A provides an additional 248 registers giving greater programming and data handling flexibility. These registers are of the same configuration as the calculator registers and useable for storing data or instructions. The execution times for functions in the extended memory are comparable to the times to perform the same function in the calculator alone. A basic system configuration consisting of a 9100A/B calculator and a 9101A extended memory unit requires no interfacing unit. Additional peripherals such as the 9120A printer or 9125A plotter may be added to the basic system using a 9102A buffer unit.

COMMANDS

The 9101A is addressed by the FORMAT (FMT) instruction from the calculator followed by a second instruction to define the desired operation (FMT) (   ). These commands provide for indirect addressing.

Data Storage: FMT y ⇥ ( ) and FMT π are used to store and recall data respectively.

Program Storage: FMT FMT and FMT GO TO are used to store and recall programs or subprograms. FMT END returns control from a subprogram to the calling program.

Indirect Arithmetic: FMT +, −, ×, ÷ enable operation on and modification of the contents of any of the 248 registers.

Subprograms: FMT GO TO to call a subprogram and FMT END to terminate execution of a subprogram, returning control to the calling program.

A numeric diagnostic code appears in the 9100A/B X register when an illegal machine operation has been specified, thus assisting the operator in rapid program debugging.

MEMORY

Registers: There are 248 registers that may be used to store 248 constants or up to 100 programs (totaling 3,472 program steps), or a combination of constants and programs. Total of 20,832 bits of core memory.

Protection: FMT SET FLAG protects the contents of consecutive registers specified by the user. Initial program protection is always automatic; the protected region expands to include a newly stored program. Register contents remain intact when turning power on or off.

GENERAL

Temperature: Operating range, 0–45°C.
Weight: Net 30 lbs (13.6 kg), shipping 40.5 lbs (18.4 kg).
Power: 115 or 230 V ±10% (slide switch), 48 to 60 Hz, 95 V/ amps max.
Dimension: 7 9/32 in high by 16¾ in wide by 19 5/16 in deep. (184.9 mm × 425.5 mm × 490.5 mm).

PURCHASE PLAN

Purchase:	HP 9101A, $3690.00.
Rent:	$195 per month.
Lease:	$ 95 per month.

HP Model 9102A
Buffer Box

DESCRIPTION

The HP 9102A buffer box is a necessary unit when using a 9100A Calculator, 9101A extended memory and one or all of the peripherals. The buffer isolates the peripherals in a system and strengthens the output signals.

The buffer also allows for a longer cable to be attached to the 9101A. This optional 5 ft cable (Part No. 11162A—$85) allows the 9101A to be placed farther than the 18 ft standard cable will allow.

PURCHASE PLANS

Purchase:	HP 9102A, $280.00.
Rent:	$ 15.00 per month.
Lease:	$ 7.25 per month.

Service Contract Available

MANUFACTURING DIVISION: LOVELAND DIVISION
P.O. BOX 301
815 FOURTEENTH STREET S.W.
LOVELAND, COLORADO 80537

References

(1) Gibson F. Anderson, 'A Programmable, Modular, Bi-directional Data Coupler,' **Hewlett-Packard Journal,** September 1970.
(2) Charles W. Near and Robert E. Watson, 'Read-only memory adds to calculator's repertoire,' Electronics. February 3, 1969, page 70.

Russ Sparks

Russ received his BSEE degree from the University of Michigan in 1963 and continued at Michigan for his MSEE which he received in 1965. He joined the Hewlett-Packard Loveland Division in 1965 and has been engaged in various projects relating to computational equipment. He was project leader on the HP Model 9101A Extended Memory.

Russ is interested in flying and likes backpacking and fishing. When weather limits these activities, Russ does woodworking.

Optical Card Reader
for Fast Calculator Programming

By Gene Zeller

CARDS MARKED WITH PENCIL, PUNCHED HOLES, OR BOTH, can be used to enter programs rapidly into the HP Model 9100 Calculators. The HP Model 9160A Mark Sense Card Reader, Fig. 1, detects coded marks representing key codes. It transmits this coded information to the calculator. Many users can simultaneously write machine-readable programs without tying up the keyboard. Individual programs are entered into the calculator in seconds. Users in the field can write on the cards then, with the card reader, enter the data into the calculator during a program execution.

The model 9160A detects pencil marks and/or punched holes in the program-data card by means of photo transistors that look downward from the optical head onto the program-data card. Clock or strobe marks located near the edge of the program-data card, provide the timing and control sequence, Fig. 2.

The card reader detects marks on the card by monitoring the amount of light reflected to the individual photo transistors. The unmarked program-data card has a very rough surface that scatters the light originating from the light source. Marking the card with a soft lead pencil smooths the rough surface, causing a pseudo-mirrored surface. The optical head contains both the light source and the photo transistors. Its geometry is such that the
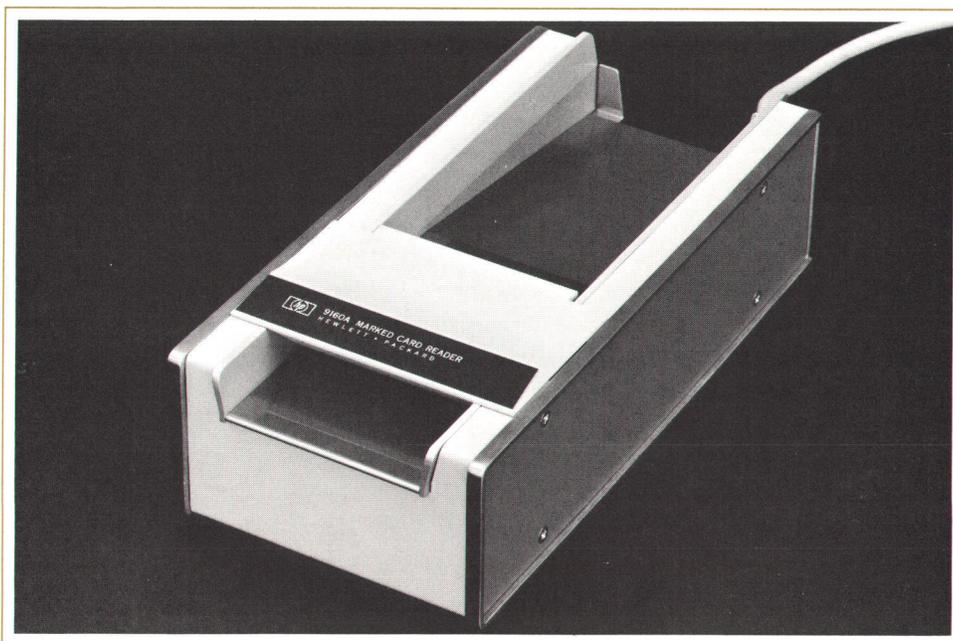
Fig. 1. *Pencil marks in boxes on cards are read by the Hewlett-Packard Model 9160A Marked Card Reader, and the information fed to the Model 9100A/B Calculators.*
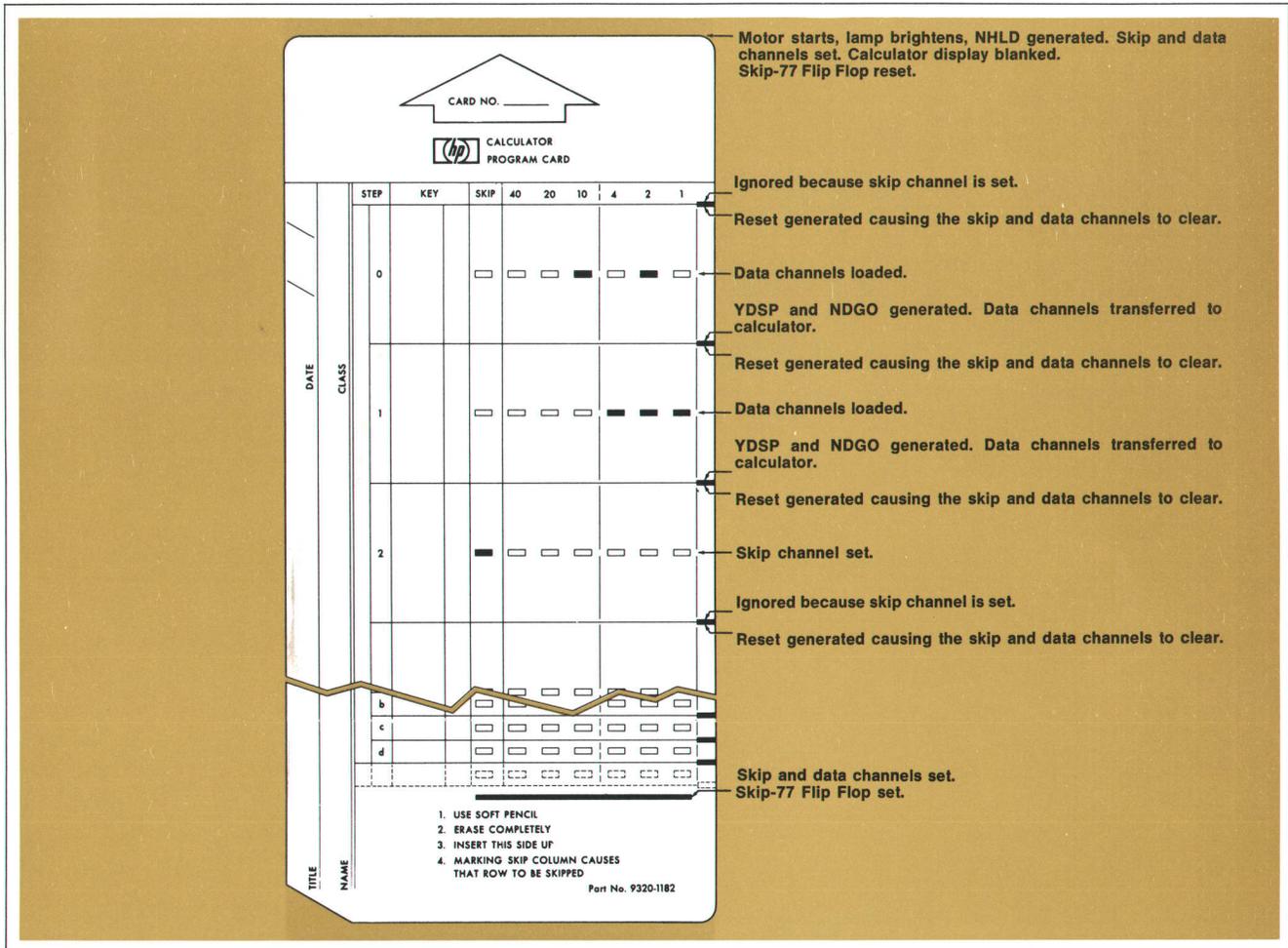
Fig. 2. *Interpretation of marks on a Model 9160A program card.*

reflected light, in the presence of a mark on the program-data card, is reflected away from the photo transistors. The *absence* of light is detected as a mark on the program-data card.

The Model 9160A Card Reader, Fig. 3, consists of six data channels, a skip channel and a strobe channel. The data channels read the data in the data field of the program-data card, and stores this information. The skip channel, when marked, causes skipping of the data channel information adjacent to the marked skip channel.

Functions of the strobe channel are to generate signals that 1) start the motor that draws the program-data card through the reader, 2) brighten the lamp to the required intensity, 3) load information stored in the data channel flip-flop into the calculator, and 4) reset data channel and skip channel flip-flops.

All channels contain a light sensor which depends upon the incident flux originating at the light source. Since the

absolute value of incident flux can change considerably over the life of the instrument, logarithmic output is desirable. Such an output is obtained by using a phototransistor as a current source through two diodes. The voltage across the diodes varies logarithmically with the current through them.

The strobe channel receives its inputs from the clock marks located near the edge of the program-data card. When the strobe channel light sensor 'sees' the area below on the card going light to dark, a positive step is generated and amplified. This signal, after squaring, is differentiated and fed into the load signal generator whose outputs are NDGO and YDSP. These outputs are sent to the calculator if the inhibit signal is not present. NDGO and YDSP are load commands for the calculator.

Finally, as the black strobe mark passes beyond the field of vision of the strobe channel photo transistor, a negative going step is generated and amplified. The signal,
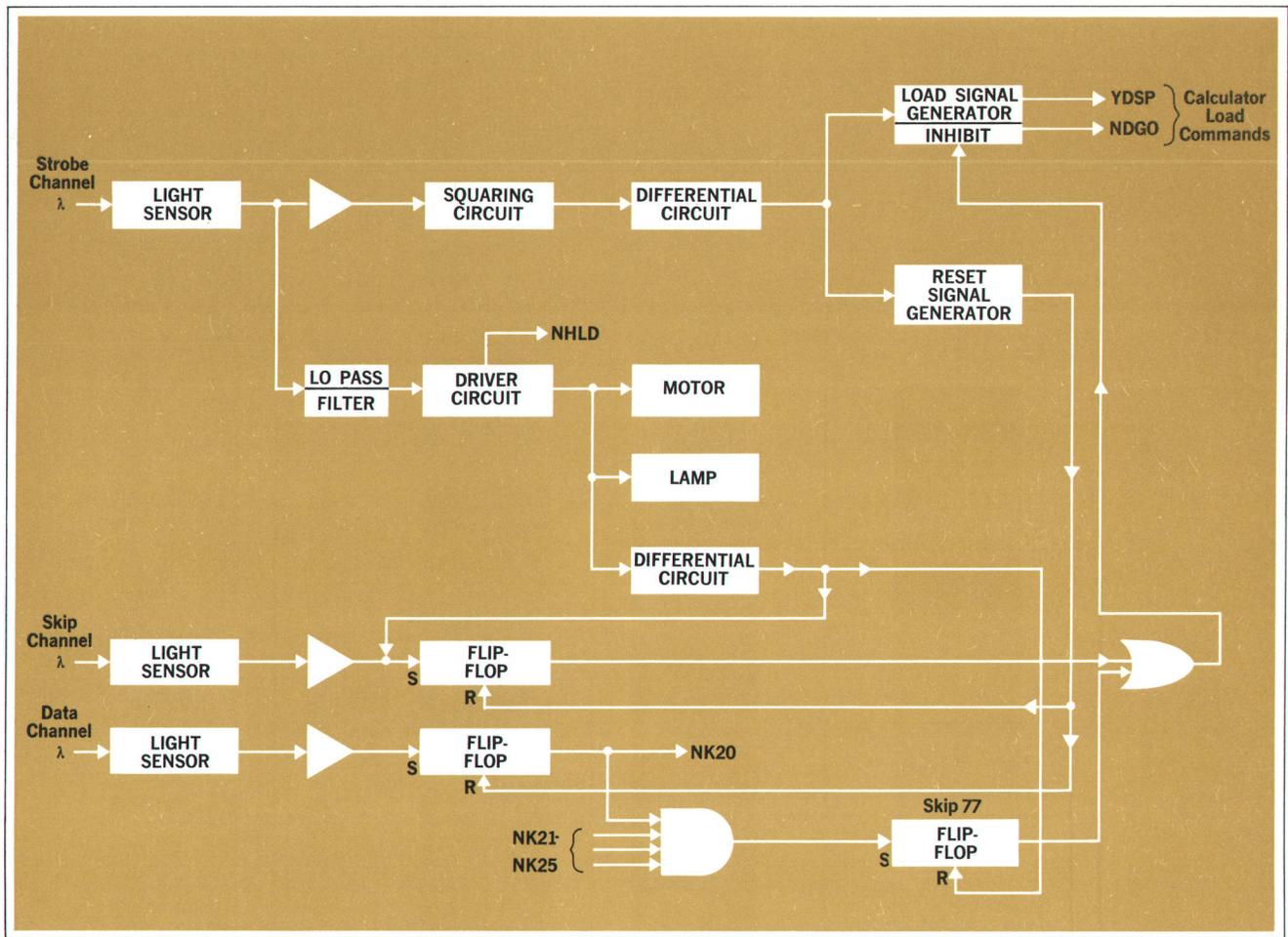
9

Fig. 3. *Block diagram of the Card Reader. There are a total of six data channels to generate signals NK 20 through NK 25.*

after squaring, is differentiated and fed into the reset signal generator. This reset signal will clear all data channel flip-flops and skip channel flip-flop if they are previously set.

In the data channel, the light sensor detects the absence of light due to the pseudo-mirrored surface created by the mark on the program-data card. A pulse is generated, amplified, and sets a flip-flop. Data is then loaded into the calculator by the strobe channel load commands.

The skip channel operates much as the data channels do with one exception. The skip channel has two ways of being set. In addition to the setting of the skip channel flip-flop by the light sensor, this flip-flop is set by way of the strobe channel every time a card is inserted into the Model 9160A Card Reader. As a result, the first clock mark is ignored but all of the data channel flip-flops are cleared.

In addition, there is a SKIP-77 flip-flop that is set whenever the skip channel and all data channels are set. The SKIP-77 flip-flop can only be cleared by card entry. Consequently, marking the card with SKIP-77 inhibits the remaining strobe mark load signals. In effect, the remaining data field is completely ignored. This important feature eliminates the necessity of marking the SKIP channel the full length of the card following the data field when using the Model 9160A for fast data entry into the calculator when in RUN mode.

When there is no card in the Model 9160A, the lamp used as the light source glows dimly, waiting to detect a card. When a card is in place, the lamp's intensity increases to that required for reading. With this automatic dimming technique, the life of the thin-line filament of the incandescent lamp (that spans the entire card width) is increased. 🔖
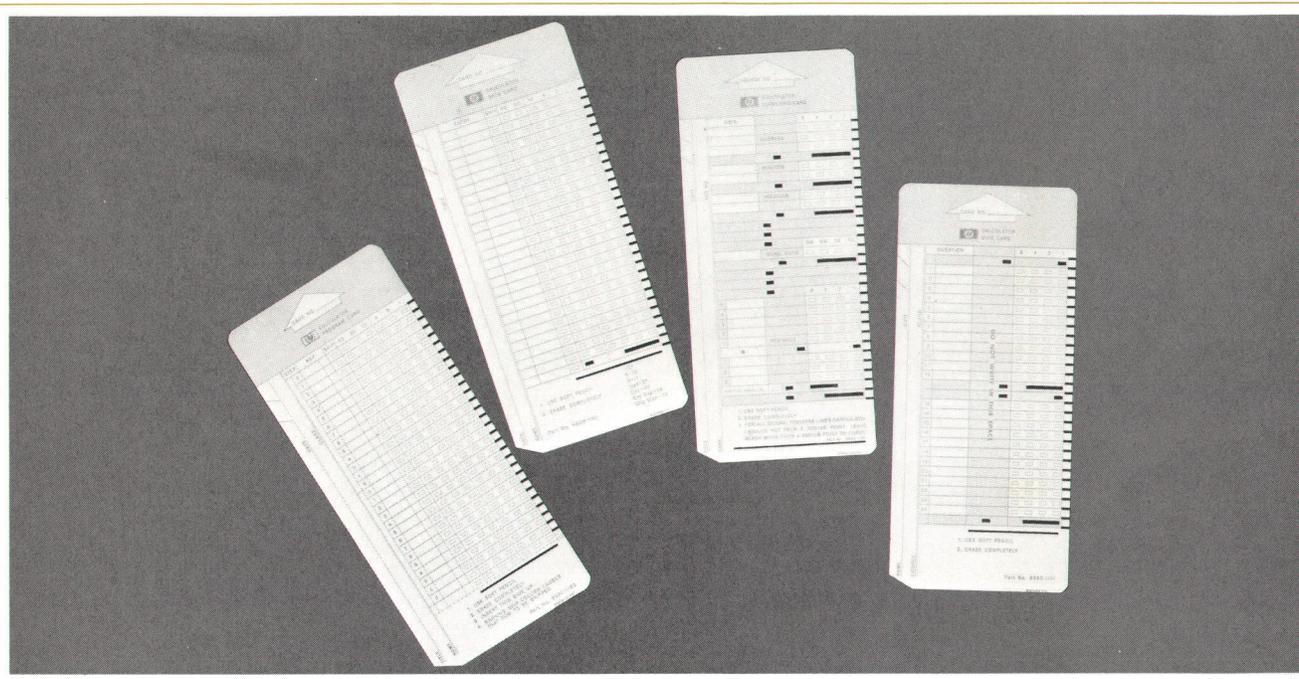
10

Fig. 4. *Four cards are used with the Card Reader. At left, a program card, next a data card, a surveying data card, and at right a quiz card. The quiz card can be used to mark answers on up to 24 multiple choice questions.*

SPECIFICATIONS
HP Model 9160A
Marked Card Reader

DESCRIPTION

Line Width:
Minimum 0.020 inch pencil mark required for reliable sensing.

Reading Rate:
20 milliseconds per character. Inserting card starts motor which pulls card through reader.

Codes:
Column weights of marked columns are added to total the calculator key code.

Operating Modes:
a) Program mode—program instructions on the card are loaded into successive calculator core memory locations.
b) Run mode—any key operation may be performed by marking the appropriate octal key code.

Skip Channel:
Marking in the skip channel causes that character to be ignored.

GENERAL
Weight: Net 4.5 lbs., shipping 5.5 lbs.
Power: Takes power from -hp- 9100 Calculator.
(Idle 2.5 watts, running 3.5 watts.)
Dimensions: 3½" high by 5⅓" wide by 11¼" deep.
Temperature: Operating range, 0°–55°C.

PURCHASE PLANS
Purchase: HP 9160A, $490.00.
Rent: $ 35.00 per month.
Lease: $ 12.74 per month.

MANUFACTURING DIVISION: LOVELAND DIVISION
P.O. BOX 301
815 FOURTEENTH STREET S.W.
LOVELAND, COLORADO 80537

CARD FORMAT
The 9160A uses standard size tab cards printed in the format shown here. Each card will hold 28 program instructions. Cards may be cascaded for longer programs.

Marking boxes are used to indicate areas of the card that are examined by the mark reader for presence of a pencil mark. A soft-lead pencil is recommended; the mark provided by a hard-lead pencil may not be read consistently by the mark reader.

CARDS
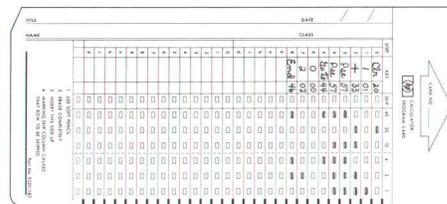**Furnished:** Package of 100 Calculator Program Cards
Available: 9320–1182 Calculator Program Format
9320–1191 Quiz Answer Format
9320–1192 Calculator Data Format

**Price: Pkg. of 2,000 $15.00**
Box of 10,000 $60.00

SPECIAL PURPOSE CARD INK SPECIFICATIONS
The mark reader reads marks on the tab card as contrasts in light reflection which are detected by photo sensors as the card passes under a reading head. If a mark is not present, a maximum amount of light will reflect; if a mark is present very little light will reflect. This requires that a non-reflective ink be used for the clock marks and reflective ink be used for all other printing on the card.

Clock marks and preprinted codes, if any, must be printed in non-reflective black ink with a reflectance value of less than 30 millivolts as measured on a Kidder MR-8 Tester. (Non-reflective ink should not be used elsewhere on the card.) The entire area of the clock mark must have solid ink coverage without voids. Edges must be sharp, clean, and free of raggedness. Marking boxes must be printed in reflective ink with a reflectance value of more than 50 millivolts as measured on a Kidder MR-8 Tester. Lithat Red ink, L 3 series, can be used.

Typical Program

The equation for f(x) is solved by marking the cards as shown here.
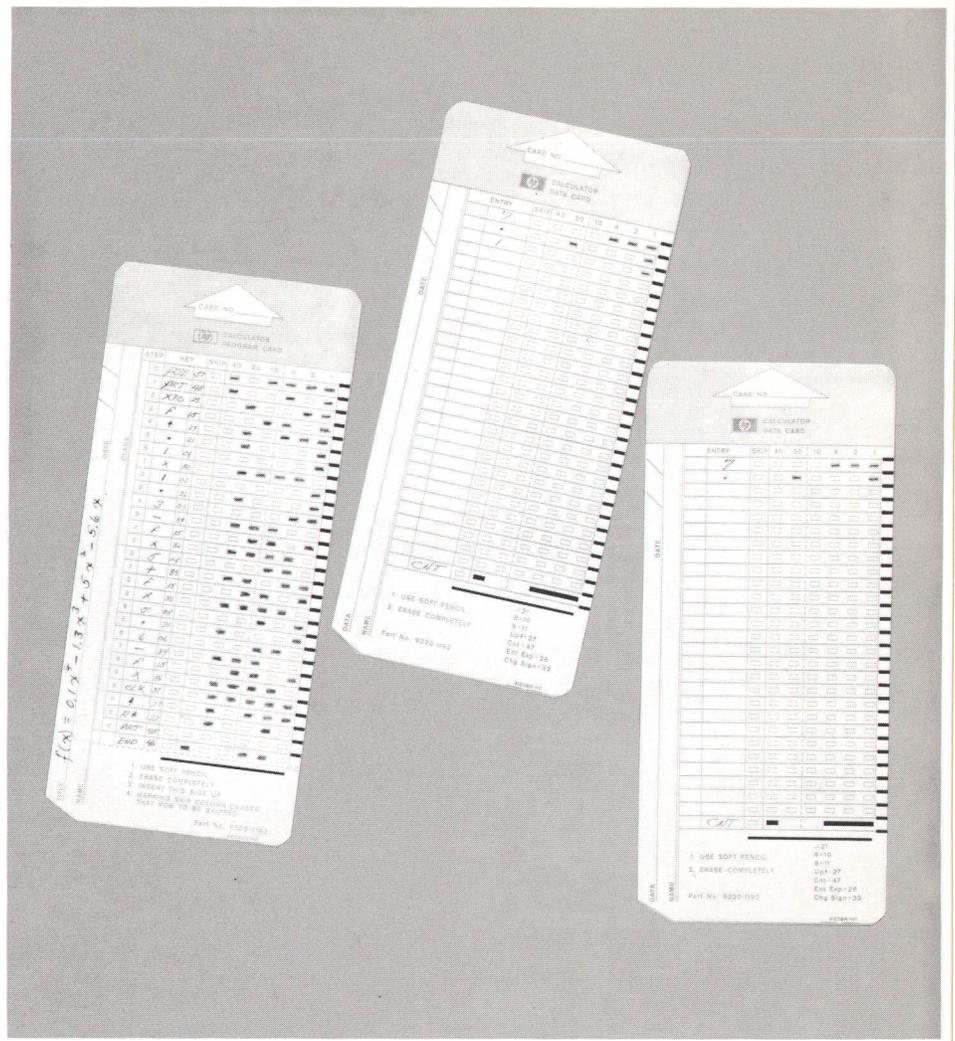
Acknowledgments

Don Miller and Dick Osgood of HP Labs, Palo Alto, did the electrical design and mechanical design respectively of the first 9160A prototypes. Harry Schneider and Jerry Simon designed the molds and tooling for production. Al Vigil coordinated arrival of parts and assumed production engineering responsibility for the 9160A. Special thanks is due Lou Dohse whose guidance and experience contributed greatly to the project.

Gene Zeller

Gene is a graduate of the University of Wisconsin, 1969, with a BSEE. He joined the Hewlett-Packard Loveland Division in 1969 and had engineering responsibility for both the Model 9160A Card Reader and the Model 9102A Buffer Box. He is now working on memory system design.

Gene enjoys camping, backpacking and climbing. He is also a star softball player.

Hard Copy Output For The
System 9100 Computing Calculators

By Chuck McAfee

In using the HP computing calculators, situations develop where automatic hard copy is especially desirable. Some of these include programs from which large tables of numbers result, such as those from a loan amortization schedule. Very often a record of data, as well as calculations performed on that data, is required.

Quiet, rapid reduction of input and computed data to permanent printed copy is accomplished with the HP Model 9120A Printer, Fig. 1. The Printer is designed for use with the HP Model 9100A and 9100B Computing Calculators[1] and prints any information stored in the 'X' (keyboard), 'Y' (accumulator), or 'Z' (temporary) display registers of the calculator. The data are printed exactly as they appear on the CRT display. Any combination of the three registers can be printed.

Print Command

Display registers to be printed are selected by push-button switches on the front of the printer. The printer then responds to keyboard (manual) or programmed PRINT/SPACE command.

Paper Advance

The same instruction (PRINT/SPACE) is used to advance paper between prints. If the machine is being operated manually, paper may be advanced by repeated depressions of the PRINT/SPACE key or by simply holding the key down. Paper will be advanced during program execution by inserting successive PRINT/SPACE instructions in the program. In either manual or program operation, the result of successive PRINT/SPACE commands is that the selected registers are printed on the first command, after which paper is

advanced one line for each command. In addition, paper can be advanced when no register select button is depressed, or between PRINT/SPACE commands during a program, simply by pressing the PAPER button on the top of the printer. This button is also used to feed a new roll of paper into the printer mechanism.

Memory List

In addition to the displayed information, a copy of calculator memory contents can be obtained. This is accomplished by momentarily pressing the LIST button on the printer. The resulting printout will begin at the memory location determined by the present state of the calculator program counter and will continue until an END instruction is encountered in memory or until the STOP key is pressed. An example of a memory list, which consists of the memory location and an octal code[2] representing the keystroke stored in that location, is shown in Fig. 2.

Pause

The PAUSE key can be used to halt execution of a program at a PRINT/SPACE instruction. When the PAUSE key is held down the calculator will stop when a PRINT/SPACE instruction is executed, and will resume calculation when CONTINUE is pressed. This feature might be useful, for instance, if the operator decides to enter new data into the program immediately after printout of the results of calculations on previous data.

Keyboard Inhibit

The printer inhibits the calculator keyboard, except for the STOP and PAUSE keys, during the time it is executing a PRINT/PAUSE command.

13

Fig. 1. *Contents of all three calculator registers is printed at three lines per second with almost no noise. The new Hewlett-Packard Model 9120A Calculator Printer uses an electric writing technique.*
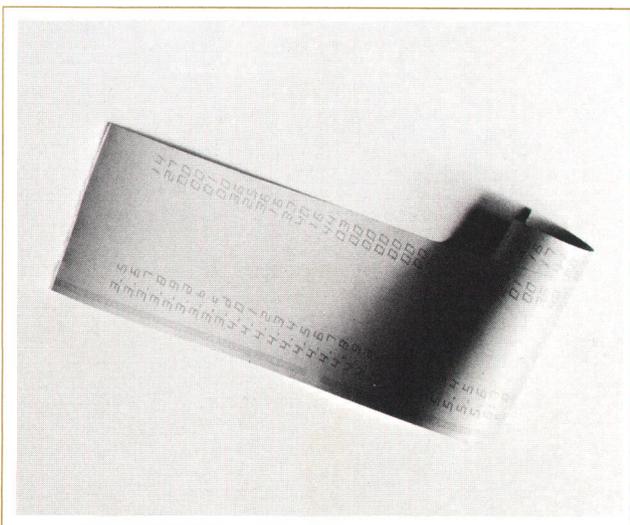

Fig. 2. *By pressing the LIST button on the printer, a complete printout of the contents of the calculator memory is made.*

Paper

The writing technique uses electro-sensitive paper shown in cross section, Fig. 3. The base paper is coated with a layer of black polyvinyl alcohol (PVA). A thin layer of aluminum foil is evaporated over the PVA, and the aluminum is then coated with a layer of conductive zinc oxide dispersed in an organic binder.[3]

Writing Process

A short burst of current passes from the stylus through the zinc oxide coating into the aluminum layer. The aluminum film is vaporized in an area beneath the stylus, exposing the black PVA layer to produce a mark and open-circuiting the current path. Thus writing is quiet and very rapid; no further processing is necessary to develop the mark, and the mark is permanent.

Control Logic

Fig. 4 is a block diagram showing printer control logic

and lines of communication between calculator and printer. Logic-control stimulation and displayed-character information are supplied to the printer from the calculator in digital form. This information is then processed by printer circuitry and fed to various circuits that 1) cause the print styli to scan the paper, 2) decode and print the characters, 3) cause paper to be advanced, and 4) indicate to the calculator that the operation has been completed and the printer is awaiting the next PRINT/ SPACE command.

Character Generation

The current pulses which mark the paper are supplied through seven in-line tungsten styli (Fig. 5) which sweep across the paper while the digital character information from the calculator is being decoded. The current pulses are transmitted from the decoder-driver circuitry to the tungsten styli through a flexible printed circuit cable. The styli are offset from vertical at a slight angle to improve readability.

Each character is formed by printing the necessary dots in a 5 x 7 dot matrix, Fig. 6. The numeral 1 is shifted to the center of the matrix to enhance appearance of the printout.[4]

Motor

Force to sweep the print styli across the paper and to advance paper is provided by a synchronous motor, fabricated with the rotor external to the field windings. This 'inside-out' construction makes it possible to achieve the necessary torque and inertia in a relatively small size. The scan rate depends on the motor speed, and therefore is reduced from the normal rate of 180 lines per minute at 60 Hz power line frequency to 150 lines per minute at 50 Hz.

Logic Circuit Board

The bulk of the printer logic circuitry is contained on a printed circuit board of multilayer construction. Each layer of circuitry is duplicated in the board for high mechanical reliability.

Styli Scan Mechanism

The transport mechanism for the print styli consists of a scanning cam, which is driven by the motor, and slider, Fig. 7. A helical traversing track is machined into the cam wall and a track follower fits into the track to pull the slider and styli assembly across the paper as the cam rotates. A magnetic pickup senses the position of the rotating track so scan and character generation can be properly synchronized to position the characters on the paper.
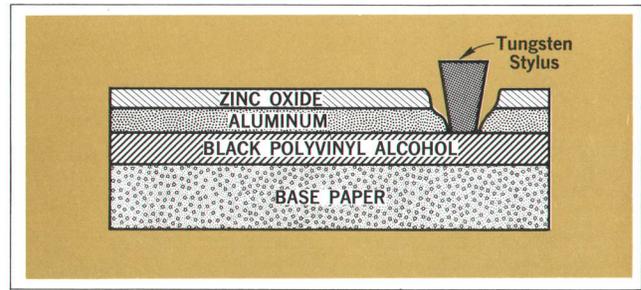

Fig. 3. HP electrosensitive paper is layered as shown here. The tungsten stylus burns through to the black layer.
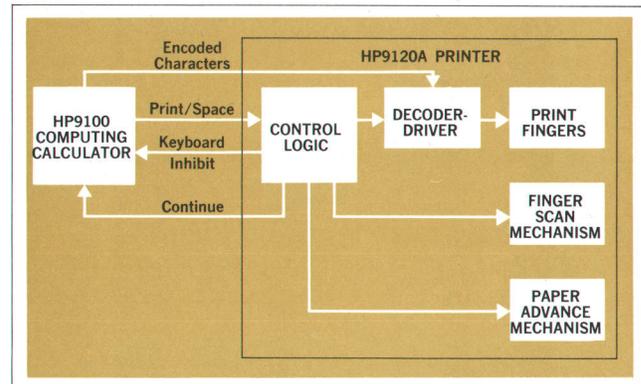

Fig. 4. Control logic of the printer and interconnection with the calculator.
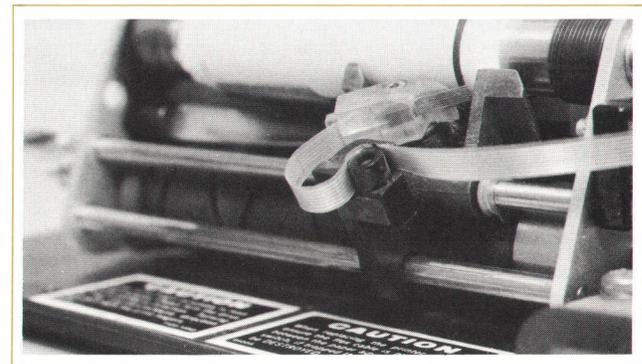

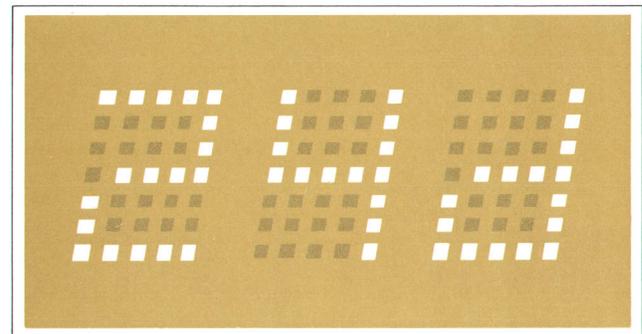**Fig. 5.** Writing head is connected to the instrument via a flexible cable.

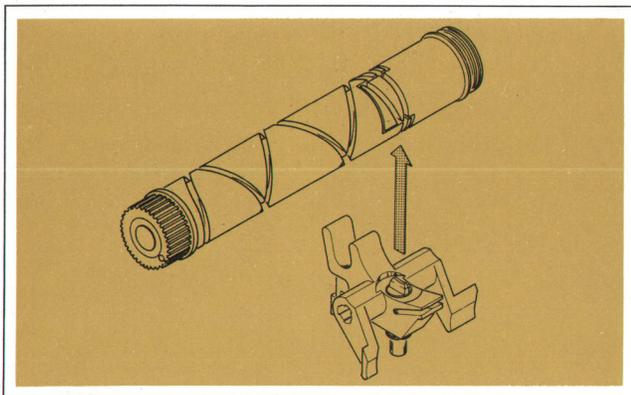**Fig. 6.** How the 5 by 7 dot characters are generated.

Fig. 7. *Printer mechanism is a cam and slider arrangement.*

Acknowledgments

Initial design and breadboarding of the HP Model 9120A Printer was done in Hewlett-Packard Laboratories under the direction of Dick Monnier, with Dave Cochran, Charles Mitchell, Ken Petersen and Clarence Studley contributing. The design team at the Loveland Division included Gary Jacobson, Gary Paulson, and Fred Wellschleger on mechanical design. Don Aupperle and Hudson Grotzinger were responsible for industrial design and packaging. Wally Wahlen and the author handled the electronics. Virgil Jensen, Ron Culver, Henry Hetzel, and Larry Linn also made significant contributions. Wally, Virgil, and Ron continued their support after the Printer went into production.

References

(1) 'The HP Model 9100A Calculator,' **Hewlett-Packard Journal,** September 1968.

(2) Richard E. Monnier, 'A New Electronic Calculator with Computerlike Capabilities,' **Hewlett-Packard Journal,** September 1968.

(3) Dale R. Davis and Charles K. Michener, 'Graphic Recorder Writing Systems,' **Hewlett-Packard Journal,** October 1968.

(4) Thomas E. Osborne, 'Hardware Design of the Model 9100A Calculator,' **Hewlett-Packard Journal,** September 1968.

Chuck McAfee
Chuck is a Colorado native and received a BSEE from the University of Colorado in 1963. He moved West to Stanford University to receive his MSEE in 1965 and the degree of Engineer from Stanford in 1966.

Chuck worked as a part time research assistant at Stanford, then joined Hewlett-Packard in 1966, where he worked on the circuit design for the Model 3370A Gas Chromatograph Integrator. He was project engineer on the Model 9120A Printer and is now involved with memory design.

Chuck is a member of Tau Beta Pi, Eta Kappa Nu and Sigma Xi. In the winter he likes cross-country and downhill skiing; in the summer he enjoys backpacking and camping. He recently started shooting rapids in a kayak.

HEWLETT-PACKARD JOURNAL ⓗ *OCTOBER 1970 Volume 22 · Number 2*

TECHNICAL INFORMATION FROM THE LABORATORIES OF THE HEWLETT-PACKARD COMPANY   PUBLISHED AT 1501 PAGE MILL ROAD, PALO ALTO, CALIFORNIA 94304
Editor: R. H. Snyder   Editorial Board: R. P. Dolan, H. L. Roberts, L. D. Shergalis   Art Director: Arvid A. Danielson   Assistant: Maridel Jordan