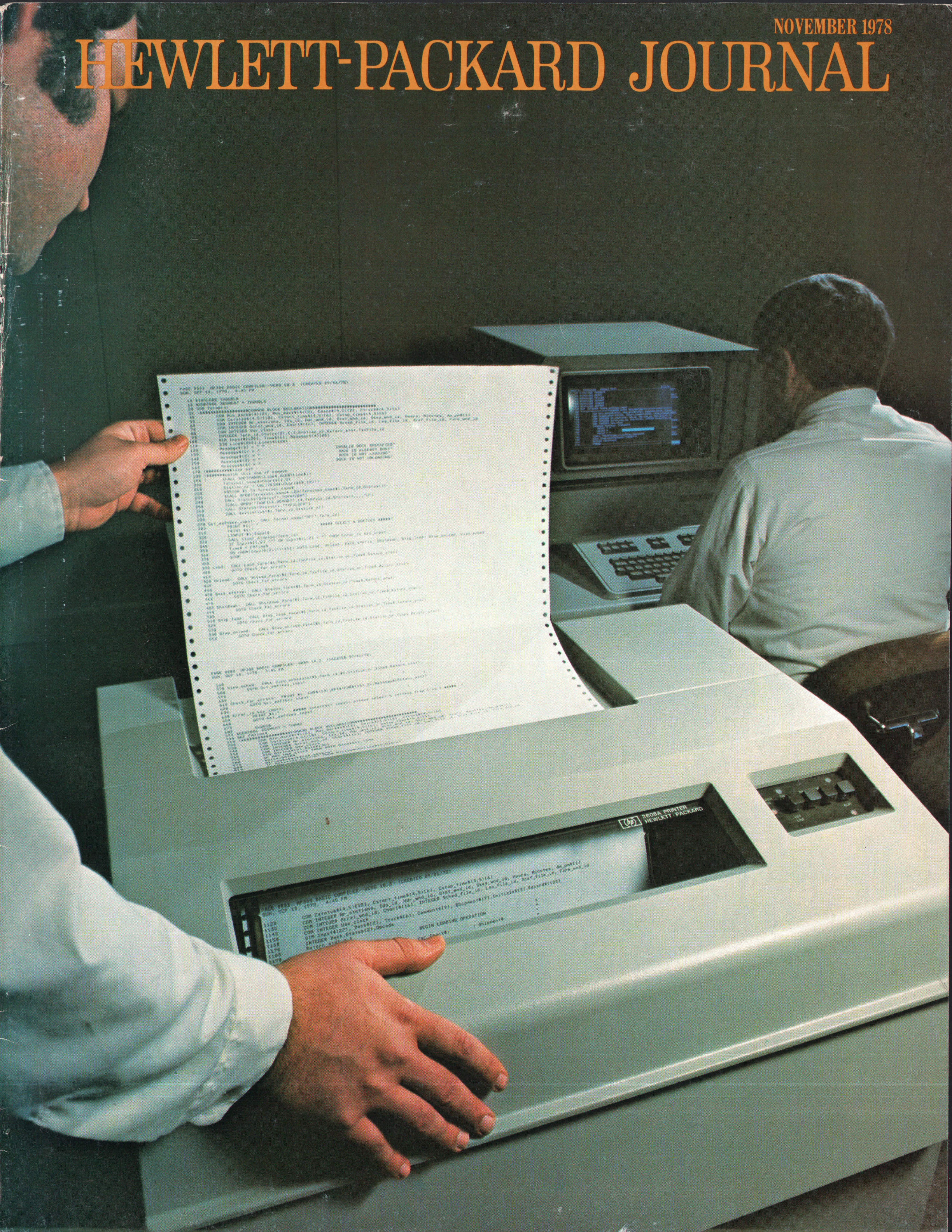
NOVEMBER 1978

HEWLETT-PACKARD JOURNAL



Printer and Printing Terminal Gain Versatility and Mechanical Simplicity with Microprocessor Control

A 180-character-per-second dot-matrix printing mechanism achieves high throughput by skipping over blanks and printing in either direction under microprocessor control. Versatility is enhanced with variable horizontal pitch, a full 128-USASCII-character set, and flexible interfacing.

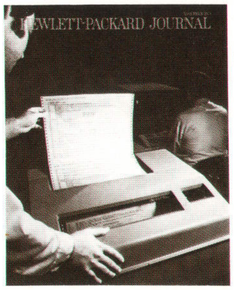
by Todd M. Woodcock

IN MOST COMPUTER SYSTEMS, the final product resulting from data collection, storage computation, and manipulation is a printed page that transmits information to a person. At present, devices that produce these pages fall into the general category of *printers*. With the possible exception of tape and disc drives, these are the only devices in a computer system whose role is primarily mechanical.

Since computer systems range from compact desktop units to elaborate multiprocessing mainframe units, it is not surprising that the available printers range from those that basically resemble typewriters to high-speed machines that produce complete pages at rates faster than one per second. Nevertheless, although each computer system requires a somewhat different set of printer capabilities, there are several identifiable printer groupings within which the performance requirements are essentially alike.

One such grouping would be in the low- to medium-speed range, defined as about 200 lines per minute. Printers capable of this speed typically are used as small system printers where they are generally dedicated to only a few users and are not required to supply the massive throughput typical of the larger systems. These printers are generally used for program listings and for short jobs such as might be needed by small businesses, scientific investigations, and individual programmers during program development. In these applications, the printing rate is most often measured in lines per minute, since traditional line printers with their fast, parallel interfacing are most easily adapted to this measure of performance.

When combined with a keyboard, a printer becomes a terminal and is generally used as a system console or a remote terminal. The performance of terminals is most often measured in terms of their transmission rates, with those capable of operating at data rates of 1200 baud and above being classified as



Cover: *The end result of most computer processes is a printed page, requiring the use of a mechanical device that often restricts the computer's throughput rate and print flexibility. Described in this issue are print mechanisms that have a mechanical simplicity conducive to higher print speeds. Microprocessor-control contributes to the simplicity while enhancing versatility.*

In this Issue:

high-performance terminals. The printing mechanism for a high-performance terminal, however, would be in the 200-lines-per-minute range.

A basic electrical difference between terminals and printers is the interfacing. Printers have traditionally used parallel interfacing where the machine receives data usually on seven or eight parallel lines with several other lines provided for paper control and handshaking. A terminal traditionally interfaces through a serial protocol in which data is transferred as a pulse train on a single line. In general, serial data transfer occurs at standardized rates of 10, 30, 60, 120, or 240 characters per second.

A New Printing Mechanism

At the inception of a new printer project at Hewlett-Packard, we felt that a high-performance printing mechanism could be adapted to fulfill the needs of users of both high-performance terminals and medium-speed printers, a significant portion of the total market. The design criteria for the new mechanism were based on:

- Performance
- Flexibility
- Long life
- Cost of ownership.

In a sense, a figure of merit for such a machine could be expressed as:

$$\frac{\text{Performance} \times \text{Flexibility} \times \text{Life}}{\text{Cost of Ownership}}$$

The performance requirements were fairly well specified by the applications in which the machines would be used. As keyboard terminals, they should operate at a throughput of 120 characters per second. In the serial environment, however, the effective throughput is often characterized by mechanical overhead rather than the actual printing rate. For example, a machine that requires one second to advance the paper to the next line can never exceed a rate of 60 lines per minute no matter how fast the printing rate.

Overhead is also involved in any computational procedures within the machine. The machine must react to any control codes that alter the response of the machine to the characters in the data stream in sufficient time to cause the proper action at the proper time. For example, a printer must respond to any command sequence within one character period or risk losing the next character in the data sequence. Therefore, to ensure data integrity a printer must be designed with a considerable print-speed margin of safety. In appreciation of the overhead requirements of a high-performance terminal, we determined that the minimum print rate required to allow continuous 120 character/second operation under average conditions would be about 165 characters/second. To provide a comfortable safety margin, we aimed for a printing speed of 180 characters/second.

The result of this design effort is shown in Figs. 1 and 2. The Model 2631A Printer (Fig. 1) is a versatile dot-matrix printer capable of printing at a speed of 180 characters/second. The Model 2635A Keyboard Terminal uses the same printing mechanism and can work with serial data streams at rates up to 9600 baud, depending on the chosen interface protocol. Two others have evolved from this first pair: the Model 2631G Printer that prints raster-style graphics as well as alphanumerics, and the Model 2639A Terminal that has a versatile, asynchronous, serial interface, allowing its use with a variety of computer systems.

Fig. 1. *Model 2631A Printer is designed for either table-top or pedestal mounting. Microprocessor control of the dot-matrix printing operation contributes to the printer's versatility while simplifying the printing mechanism. A high throughput rate is obtained with a print speed of 180 characters per second plus the ability to print in either direction and to slew past strings of blank spaces. Model 2631G, which can also print raster-style graphics, is identical in appearance.*

Dot-Matrix Printing Provides Flexibility

The method used to print a character determines to a great degree the overall flexibility of a printing mechanism. Type slugs that print one fully formed

Fig. 2. *Model 2635A Terminal (and the identical-looking Model 2639A) uses the same printing mechanism as the Model 2631A Printer. A wide selection of interface options enable its use with a variety of computer systems. Symbols are paired on the keys in the traditional typewriter arrangement, making it easy for the user to convert from typewriter to terminal operation, and a 10-key numeric pad speeds the entry of numerical data.*

character per impact are not readily adapted to automatic changes of the character set because the entire wheel, ball, or cylinder that contains the type slugs must be changed. The most flexible method of printing is the dot-matrix method in which the individual dots in a matrix are selectively energized to form the desired characters (Fig. 3).

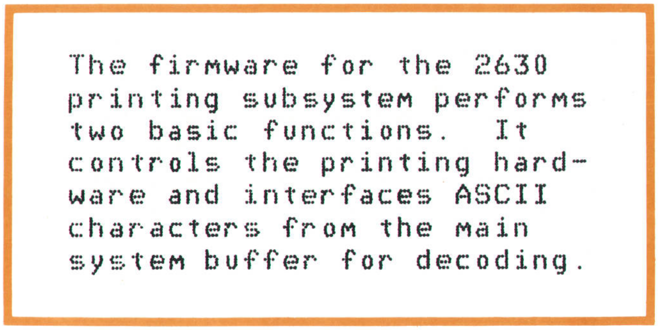
Dot-matrix printing can be done with mechanical simplicity by thermally pulsing dots on a printhead in contact with thermally-sensitive paper. However, the applications we envisaged for the new printers would often require multiple copies, so we chose to use an electromechanical method of impacting dot-shaped hammers on an inked ribbon and paper. This technique also has the advantage of not requiring special paper.

Although the mechanics of dot-matrix printing can be simplified to the repetitive use of a single dot, the controller that directs this operation is considerably more complex than that required for a full-font printing mechanism. Not only must the character be properly placed on the page, but each dot must be properly sequenced and positioned so as to produce a correctly formed character. Full-font printing needs an information rate of only one identifiable code per character whereas a dot-matrix character composed, say, of twenty dots needs to have twenty times as much information processed, and hence needs twenty times the control speed. The trade-off is the ability to print any character that may be constructed from a matrix of dots.

The manner in which a character is printed is crucial to the utility of the machine. To be useful as a keyboard terminal, each character the machine prints must be visible to the person operating the keyboard as soon as it is printed. We chose, therefore, to use a movable ballistic matrix printhead (Fig. 4), a type of device that has already proven its reliability in widespread use.

The ballistic matrix printhead used in the new machines has nine tungsten wires arranged in a vertical column. Each wire is associated with a coil that drives an armature forward when energized. The armature in turn propels the tungsten wire but when the armature closes against the coil's pole pieces, it stops while the wire continues its forward motion with enough energy to form a dot by impacting the ribbon and paper against the platen. The wire then rebounds and comes to rest at its initial position. As the printhead moves along the paper, the coils are selectively energized "on the fly" to print the dots that form each character.

The character cell, a 10 × 9 dot matrix, is diagrammed in Fig. 5. An uppercase character is formed in the upper seven rows of the matrix, leaving the bottom two rows for descenders and underlining. Columns 0, 8, and 9 are left blank to form the intercharacter spacing. Even though all the dot positions in columns 1 through 7 are available to form a character, every other dot is omitted in the horizontal lines so that no hammer is required to print dots successively in adjacent columns. Since there is an upper limit to hammer



The firmware for the 2630 printing subsystem performs two basic functions. It controls the printing hardware and interfaces ASCII characters from the main system buffer for decoding.

Fig. 3. *Example of the new mechanism's printing, shown here actual size. The mechanism forms characters by selectively activating the dots in a 7 × 9 matrix. The use of nine dots in each column enables the printing of lower-case descenders and of simultaneous underlining. Impact printing permits the printing of multiple-part forms.*

repetition rate, this dot skipping allows an increase in printhead scan speed with resulting higher throughput. The slight overlap of adjacent columns partially masks the fact that the lines were formed with non-adjacent dots.

As mentioned previously, dot-matrix printing enables the printing of any character that can be formed by dots. The new printer and terminal contain a full 128-USASCII*-character set that includes upper- and lower-case letters, numbers, symbols, and control codes. A special 128-character set can be included at the same time and selected under program control. Either character set can be changed by changing the appropriate ROM.

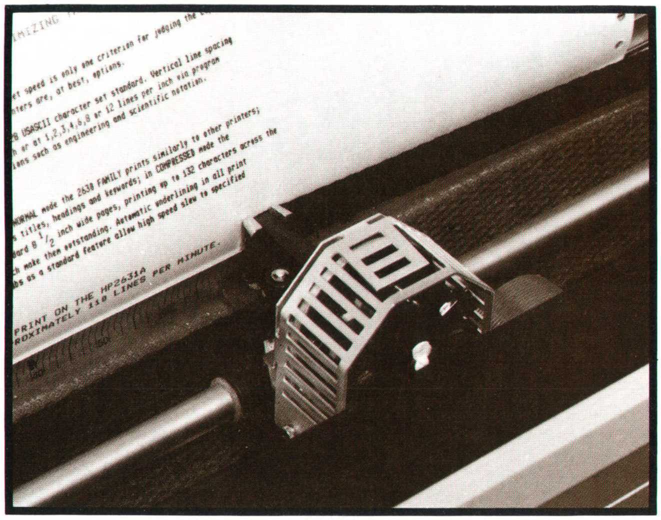


Fig. 4. *Moving printhead allows the line being printed on the 2635A Terminal to be seen. Whenever 0.6 second elapses since the last character was entered, the printhead moves to the right so the most recent character is visible. The printhead returns immediately to its previous position when it prints the next character. Every other dot is printed in the horizontal lines to permit a faster print rate.*

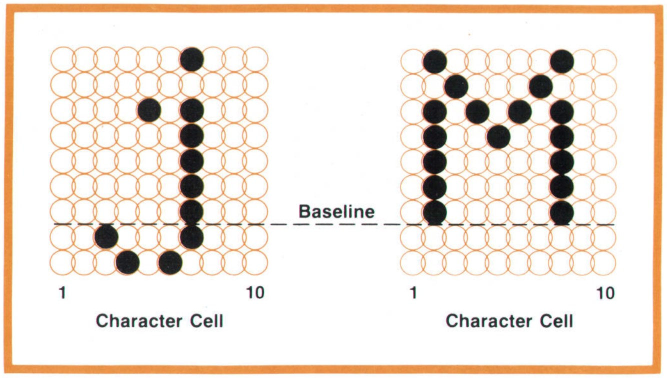


Fig. 5. *The character cell consists of a 10×9 overlapping dot-matrix. The seven upper rows are used for upper case characters while the lower two are used for descenders and underlines.*

Cartridge Ribbon

The ribbon and ribbon advancing mechanism were also viewed as areas critical to the general utility of the printing mechanism. Consequently, the ribbon is placed in a cartridge that spans the width of the print line, enabling easy replacement with a minimum of contact or mess (Fig. 6). In the Models 2631A and 2631G Printers, the ribbon slants such that the print head impacts the ribbon along the top edge when at the left side of the paper, and along the bottom edge at the right side. Wear is thus distributed over the entire ribbon. In the Models 2635A and 2639A Terminals, the printhead always contacts the ribbon along the top edge so the line being printed can be seen (a roller guides the ribbon away from the paper on the left side of the printhead). A Moebius turn where the continuously-circulating ribbon enters the cartridge inverts the ribbon on each pass so wear is distributed over a greater area.

The cartridge was designed to hold 55 metres of ribbon so at least ten million characters of acceptable print per cartridge can be obtained in any of these machines. In subsequent testing on the 2631A Printer, cartridges provided as many as fifty million characters, albeit lightly printed. Considering that an average college textbook contains about two million characters, a single cartridge ribbon could print between five and twenty-five books, demonstrating the economy of this type of printing.

The ribbon drive mechanism is harnessed to the motion of the printhead servo motor so that whenever the printhead moves from right to left, the ribbon is advanced to the right. Thus, under all conditions of printhead motion, there is fresh ribbon on which to print. This simple system is less costly and more conservative of energy than a system that requires a separate, constantly running motor.

---

*USASCII: United States of America Standard Code for Information Interchange.

Fig. 6. *Full-width ribbon cartridge is easily replaced.*

Designed for Long Life

The mechanical components of the printer were designed so that lubrication would not be required, contributing to the low cost of ownership. For example, the printhead carriage uses dry-contact fluorocarbon sleeve bearings and the tractor drive shaft rides in a self-lubricating acetal plastic bearing. Similarly, the printhead leadscrew nut is made of a self-lubricating plastic. The other rotating parts of the machine incorporate sealed ball bearings. The result of this design approach is a mechanically simple, easily maintained machine.

The new machines have sucessfully completed HP's standard class B environmental testing which includes tests under conditions of high and low temperatures, high and low humidity, vibration, and shock. In addition, 35 printers were cycled two hours on and one-half hour off continuously under computer control, 24 hours a day, seven days a week, for 22 weeks. A number of potential problems were uncovered and corrected during this test.

As a result of this test, it can be stated with 90% confidence level that the mean number of characters between failures (MCBF) for these machines exceeds 316 million. Printing with a 40% average printed page density, 316 million characters would fill 65,000 pages of computer paper. On the average, the printing mechanism will operate 2400 hours in normal use before a failure can be expected.

Organization

From the systems standpoint, both the keyboard terminals and the printers are viewed as having four basic components: the printing mechanism, the device electronics, the controller, and the package. The device electronics, such as the power supply, servo system, and control switches, were already partially defined once the basic printing mechanism was identified.

The controller, however, although invisible to the user, is the determining factor in both performance and flexibility. To provide the multitude of features required by the diverse applications of the printing mechanism, the controller must be powerful, versatile, and at the same time compact. This need was met by the Hewlett-Packard CMOS silicon-on-sapphire MC^2 controller,[1] which was being developed at the time the printer project was getting under way. A number of support devices such as read-only memories (ROMs) and read-write memories (RAMs) were also being developed so all of the necessary controller functions would be available with this family.

The control program resides in 4K 16-bit words in ROM distributed between the processor and interface printed-circuit boards. The program common to both the keyboard terminals and the printers (exclusive of interfacing) resides in 2K words on the processor board. An additional 1K words is required by each machine. With the loading of this particular block, the processor board gains an identity specific to either the keyboard terminal or the printer.

The interface ROM located on the interface board contains 1K words. Thus, the interface can be changed by changing just one board, since each interface board carries with it sufficient program memory to control its particular functions. In addition, various interfaces, although appearing to be different, often require identical signals, the arrangement of the signals and the connector being the only differences. To take advantage of this situation, an adapter was designed to fit between the connector (at the back of the machine) and the interface board. This allows a single interface board to be used for several different interface configurations.

The control firmware closes all of the control loops within the machine. Because of the great decision-making power of the MC^2 processor, a number of optimization routines were made possible. For example, the print head normally moves at 18 inches/s, the maximum allowed by the hammer action while printing in the normal mode (10 characters/inch). Normally it would step through spaces at the same rate. An optimization routine, however, looks at the number of contiguous spaces in a line, and if more than 10 occur, it accelerates the printhead so it skips over the spaces at high speed.

A further improvement uses the microprocessor to store incoming data in a buffer, and if an entire line is buffered before the printing of the previous line is completed, the new line is tested to determine whether it should be printed in the forward or reverse direction. This decision is based upon a number of criteria, including the presence or absence of control codes (such as form feed or line feed), the column positions of the first and last printable characters in the line, and the position of the last character printed in the previous line. In each case, the quickest means of printing the line is used, whether forward or reverse. Similarly, spaces at the beginning and end of a line are processed to determine where printing should begin so that the printhead can be moved directly to that position. With these and other optimizing routines, the controller improved the effective printing rate significantly.

Control by Escape Sequences

The character-serial environment of the printer and terminal allow escape sequences (control codes) to be imbedded in the data stream. The traditional dependence on separate control lines and their associated driver software are thereby avoided.

The use of escape sequences to control operation provides the user with a considerable degree of flexibility in formatting. For example, underlining may be implemented simultaneously with printing, and the three print modes—NORMAL, COMPRESSED, EXPANDED—may be enabled at any point in a line of print. In the EXPANDED mode, the machines print five widened characters per inch for use in headings and other areas where special attention is desired. Variable line spacing and the intermixed use of NORMAL and COMPRESSED characters enable information to be formatted for emphasis in a way that is not possible with constant character and line spacing. The COMPRESSED mode, which prints at 16.7 characters/inch, is also useful for printing the standard 14-inch wide, 132-column computer printout on 8½×11-inch paper for notebook use or easier filing. The microprocessor, incidentally, monitors the incoming data stored in the input buffer and if the lines contain escape sequences that affect the following characters, it prevents the mechanism from attempting to print from right to left.

Escape-sequence control also extends to other machine functions. For example, a printer used for logging data in a remote location may be sent an escape sequence via a telephone line to initiate an internal self-test routine. The printer may then be asked by another escape sequence to transmit its status and the results of the self test back to the host computer. In this way, a single operator can keep track of a number of remote printers.

Details of the print mechanism design are described in the article that follows.

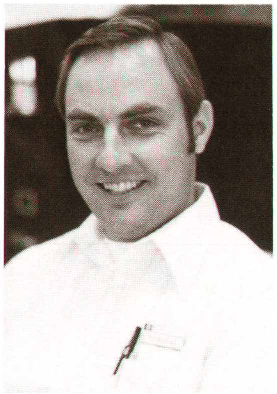
Acknowledgments

The design of the new printers and terminals required contributions from many people. In addition to those mentioned in the following article, major contributions were made by Gary Campbell, who developed the print-line buffer management and the microdiagnostics, Mark Anderson, who designed the processor and memory boards, Randy Mazzei, who did the industrial design, and Bob Carlson and Lyle Loeser who were involved in the all-important tool engineering. Doug Ballard, Larry Beck, Mike Harrigan, and Pat Donnelly were involved in the early design stages before transferring to other projects. Clyde Gregg contributed the computer diagnostics, Lamar Goats developed the thorough-going reliability verification testing, and Gary Gapp did the all-encompassing product performance testing. Overall guidance and support was provided by Jim Barnes, R and D manager.

Reference

1. B.E. Forbes, "Silicon-on-Sapphire Technology Produces High-Speed Single-Chip Processor," Hewlett-Packard Journal, April 1977.

Todd M. Woodcock



Todd Woodcock joined HP in 1975 as a designer of thermal print mechanisms and then became project manager for the 2631A and 2635A printers. Before coming to HP, Todd worked as a fruit harvesting equipment designer, an optical vision analyzer designer and a nuclear plant engineer. Todd is a 1964 BA graduate in German and speech and a 1974 BSME graduate from the University of California at Berkeley. After graduation in 1964, Todd served as a lieutenant in the U.S. Navy Reserve for three years. Born in Evanston, Illinois, he now lives in Boise, Idaho, with his wife and three daughters, ages 17, 3, and ½ years. Todd's favorite pastimes are fishing, hunting, camping and fly tying. A real cooking enthusiast, Todd won first place for his rye bread in the 1976 Western Idaho State Fair.

Managing Dot-Matrix Printing with a Microprocessor

by John J. Ignoffo, Michael J. Sproviero, Phillip R. Luque and Kenneth B. Wade

THROUGH USE OF MICROPROCESSOR control, a reliable, high-performance printing mechanism built with minimum labor and material costs was achieved for the Models 2631A and 2631G Printers and the 2635A and 2639A Printing Terminals.

The mechanism (Fig. 1) has few moving parts. The printhead carriage, moved by a 25-mm-per-revolution leadscrew, slides on two steel guide rails. The position of the printhead is accurately controlled by the machine's microprocessor, which issues commands to a hardware velocity servo loop to move the head either right or left at a controlled velocity during printing. The use of a microprocessor resulted in a control system that is much simpler than a hardware position-control system.

During printing operations, the printhead carriage speed ranges from 1 inch/second, when printing with a data input of 110 baud, to 18 inches/second with a printing rate of 180 characters/second, the maximum speed allowed by the print hammers. At any printing speed, the velocity must be controlled within 5% so smooth carriage motion, and hence evenly formed characters, result. During nonprinting movement, the carriage accelerates to 35 inches/second.

Paper motion is also controlled by the microprocessor. Paper advance is provided by a set of tractors driven by a step motor. A custom-made motor that has 180 steps per revolution was chosen so spacings of both six and eight lines per inch are possible with the tractors directly coupled to the motor. This direct drive eliminated a number of mechanical parts. A clutch decouples the drive shaft from the motor for manual fine adjustment of preprinted forms.

Character Generation

Characters are transferred from the internal buffer memory to the paper by the following sequence of events. As the printhead moves, either left or right, an interrupt is generated by the head-position logic at the end of each print cell (between column 9 of one cell and column 0 of the adjacent cell). When the print position read by the microprocessor is the desired one for the next character, the interrupt causes the microprocessor to transfer the seven-bit ASCII-coded

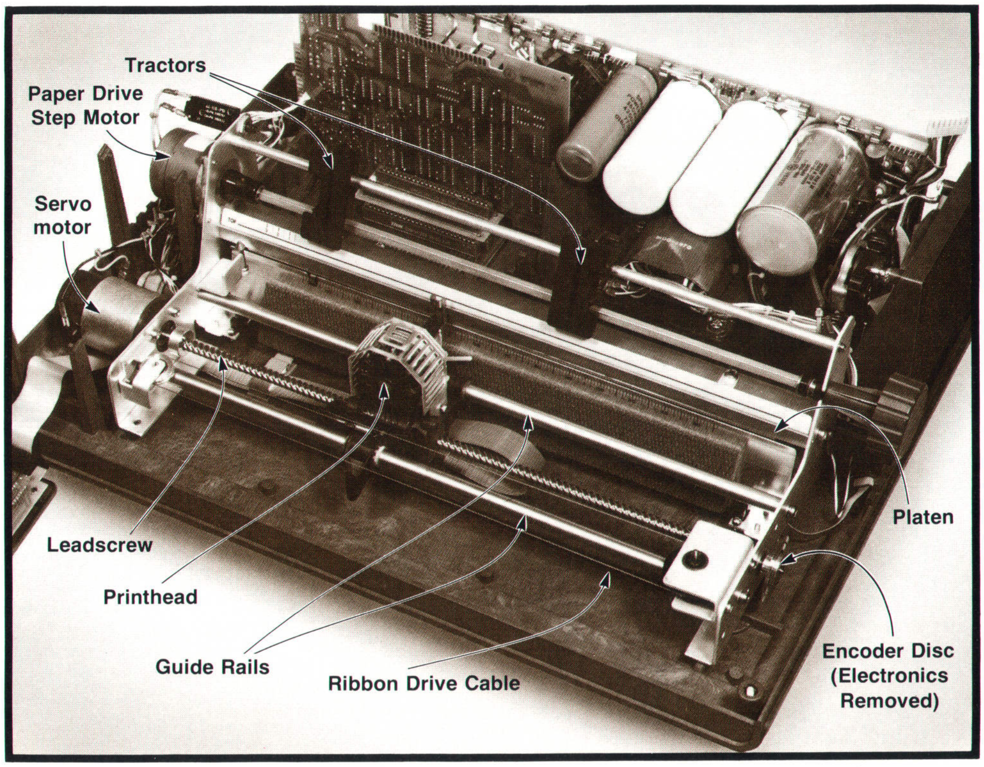


Fig. 1. *Printing mechanism has few moving parts. The same mechanism is used in both the printers and the printing terminals.*

Mechanical Design of a Durable Dot-Matrix Printer

The reliability of any mechanical device is usually inversely proportional to the number of moving parts. The Models 2631A and 2631G Printers and 2635A and 2639A Printing Terminals were designed to be as mechanically simple as possible. The use of a microprocessor reduced the printhead positioning mechanism to a servomotor-driven leadscrew and mating nut. The use of a custom-made step motor in the paper-advance mechanism enabled direct coupling of the step motor to the forms-tractor drive shaft. The few remaining moving parts were designed with sealed roller or dry-contact bearings so periodic lubrication is not required. Preventive maintenance consists simply of cleaning the fan filter and the printhead.

A photo of the mechanism is shown in Fig. 1 on page 8. The print structure assembly is the basis of the printer mechanism. During manufacture, two guide rails, the steel platen, and two end plates are accurately positioned in a fixture, drilled, and pinned together. No further adjustments are needed, or even possible. This assembly provides parallelism within 0.05 mm between the carriage line of travel and the platen, assuring good print quality for the life of the machine.

The front surface of the platen is ground to a 19-mm radius. As the paper wraps around this surface it conforms closely to the radius, thereby presenting a solid surface on which to print.

The carriage, molded of fiberglass-reinforced polycarbonate, includes the two slider bearings and other features for holding the printhead connector and for attaching the head-drive cable, the ribbon cartridge drive cable, and the leadscrew nut. A bracket attached to the top surface holds the printhead. The bracket includes an eccentric-link mechanism for adjusting the head-to-platen gap to obtain optimum print quality with multiple-part forms of varying thicknesses. The self-aligning printhead has an average life of 130 million characters but if replacement is ever necessary, it is easily done by the user.

The 75-mm long main slider bearing is insert-molded into the carriage. The other bearing, whose purpose is to resist the torque applied by the leadscrew, floats in the carriage to accommodate variations in guide-rail spacing. Both bearings have a glass-reinforced structural shell and a fabric liner containing Teflon. These do not require hardened rails or lubrication, and are unaffected by paper and ribbon debris.

The ribbon-drive cable anchors at one end to the carriage, passes around a ribbon-drive pulley at the right side of the printer mechanism, back across the printer to an idler pulley at the left side, and around the pulley back to the carriage. A spring keeps it taut. Unidirectional clutches in the ribbon-drive pulley rotate the ribbon drive shaft during right-to-left motion of the carriage but disengage during left-to-right motion.

The Package

Two structural-foam packages were designed with identical features for holding the printing mechanism and the basic electronic components. Many design details are molded in to reduce the number of parts, the assembly time, and consequently the cost. An aluminum plate on the bottom provides electrical shielding, mounting points for rubber feet, and a smooth guideway for paper fed in from the rear (paper may be fed in from the bottom when either machine is mounted on the optional pedestal).

The only difference between the packages is the extension on the terminal package for holding the keyboard. A single access cover was designed such that merely by changing inserts in the mold, parts are provided for both the terminal and the printer. Controls that are not needed by the casual user—for example, the baud-rate switch—are concealed beneath this cover.

–Robert Cort

---

character to a latch (Fig. 2). These seven bits become part of the 10-bit character-generator ROM address. The remaining three bits are dot-column position bits derived from a four-bit counter that counts pulses from an optical encoder attached to the leadscrew. Encoding logic maps columns 8 and 9 into column 0, all of which are blank, to give the intercharacter spacing, thereby obtaining the three dot-column position bits, used in the character address, from the four-bit counter output.

Transfer of the seven-bit ASCII character to the latch sets a flip-flop called PRINT GO. This enables the power electronics to be responsive to the succeeding dot-position pulses from the head-position logic. As each slot in the encoder disc is encountered, the resulting dot-position pulse triggers the hammer-enable and overdrive one-shot pulse generators. The output of the character ROM then determines which hammers will be fired. Two ROMs are provided, each storing 128 characters, thus allowing two character sets to be available at any time.

The overdrive pulse places a large voltage across the print head coils during the initial part of the hammer pulse to shorten the coil current rise time.

Printing at each column position continues until the next cell interrupt occurs. This resets the PRINT GO flip-flop which inhibits printing until the microprocessor transfers another character to the character latch.

Printhead Positioning

A diagram of the head drive servo control circuit is shown in Fig. 3. An electrical current proportional to velocity is obtained by gating a current sink with constant-width pulses triggered by the optical encoder disk attached to the leadscrew. The average current through this sink is thus proportional to carriage velocity. This current is summed with a source current proportional to the commanded velocity, generated by a microprocessor-controlled current source. The error voltage obtained at the current-summing node is buffered and applied to a slew-rate limiting amplifier which, by limiting the rate at which the servo motor accelerates, limits the power

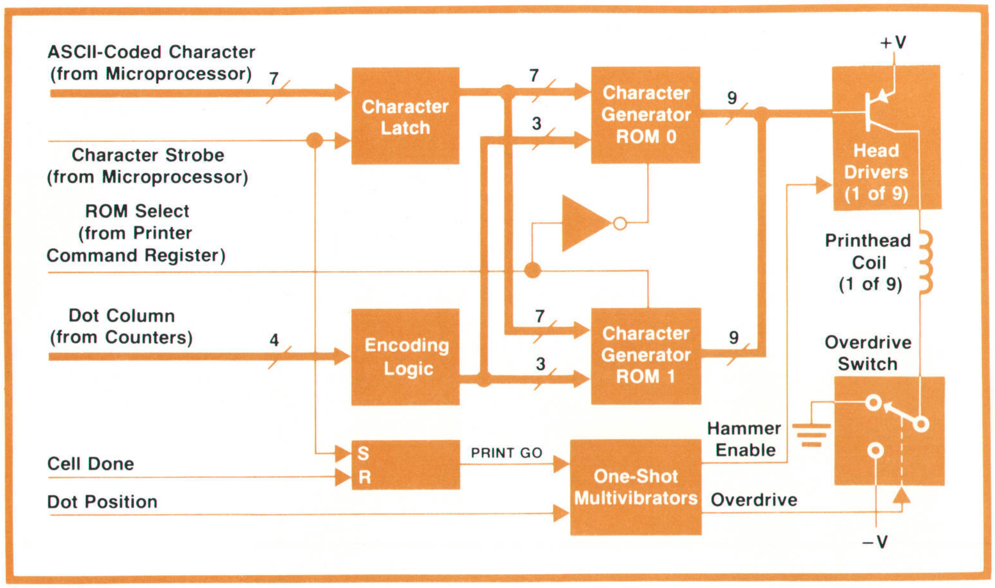


Fig. 2. *Dot generation control circuits determine which printhead hammers will be fired at each dot-column position in a character.*

dissipated in the servo motor. Without limiting, the servo motor could overheat if it were repeatedly accelerated to a high speed, as when skipping blank spaces.

The slew-rate limiting amplifier is an emitter-follower that has a capacitive load. When the error voltage rises rapidly, the transistor turns off and the output rise time is determined by the rate at which the capacitor charges through the emitter resistor. Small voltage changes and rapidly decreasing voltages,

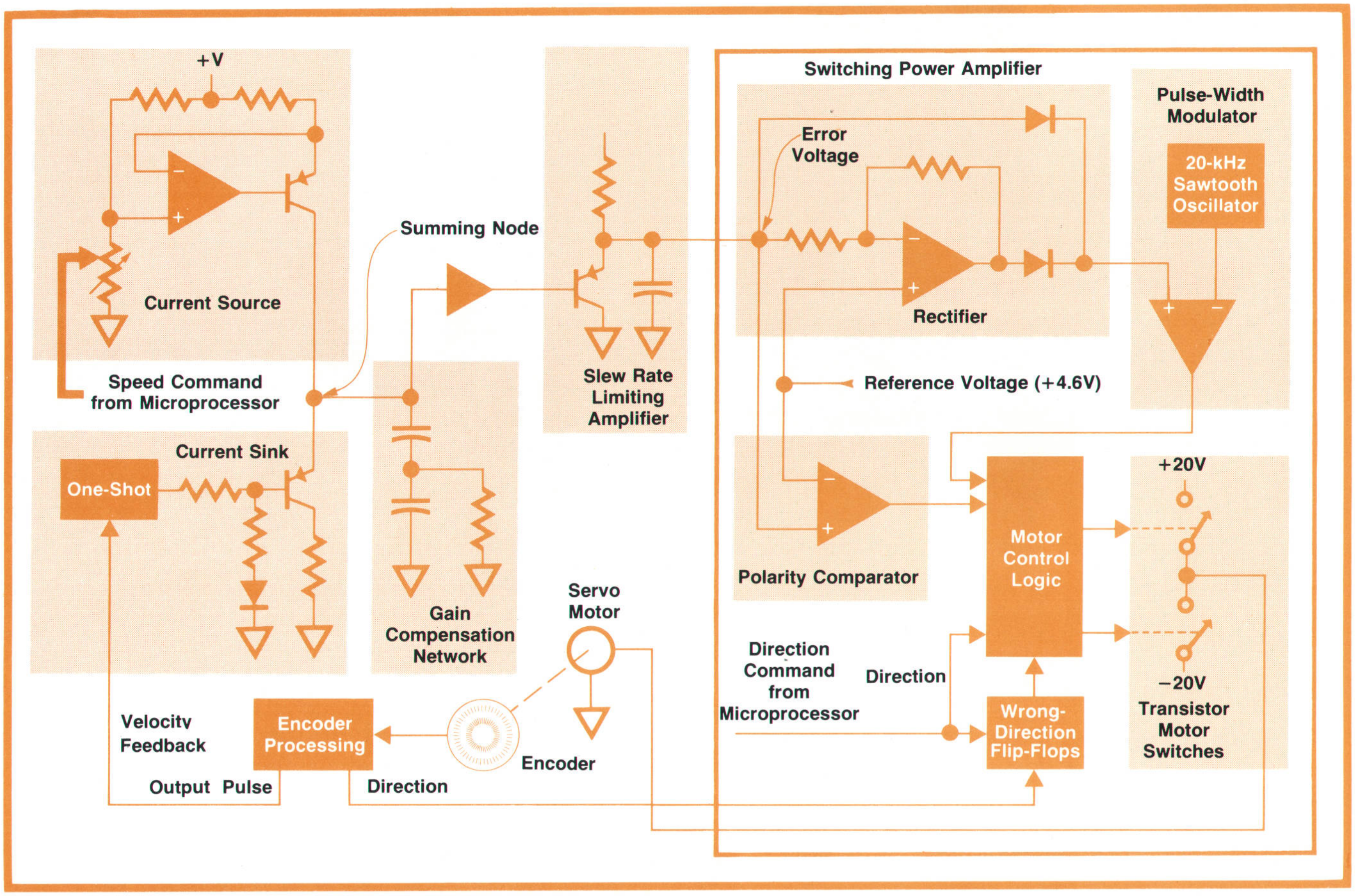


Fig. 3. *Simplified block diagram of the printhead servo control circuits. The servomotor is driven by a switching power amplifier to reduce energy dissipation.*

An Impact Graphics Printer

A second version of the Model 2631A Printer was designed as a hard-copy output for the Models 2647A and 2648A Graphics Terminals.[1] On ordinary paper, including multipart forms, this printer (Model 2631G) reproduces any raster display presented on the graphics terminals, printing a complete 720×360 dot display in 57 seconds or less.

Although capable of printing graphics material, Model 2631G retains all the alphanumeric printing capability of the Model 2631A. An escape sequence in the input data stream converts printer operation from alphanumerics to graphics and another escape sequence converts back to alphanumeric operation.

In addition to graphics capabilities, Model 2631G has some other new capabilities. One is the ability to retain several character sets simultaneously and use them interchangeably. One of the new character sets is a high-density set for use where superior print quality is required. With this character set, the machine prints overlapping dots in the horizontal lines, rather than skipping every other dot position as the 2631A does. Print speed, however, is limited to a maximum of 90 characters per second in this mode.

More flexible forms handling is another new capability. The user may now specify a page length and the text length (top and bottom margins). The 8-channel vertical forms control then adjusts to the specified text length—e.g., if the specified text length is six inches and the command "slew to the next half page" is given, the print line will be moved to three inches below the top of the text. In addition, with page and text lengths specified, an AUTOMATIC PERFORATION SKIP mode causes the printer to skip automatically to the top line of the next page when it reaches the bottom margin on the present page.

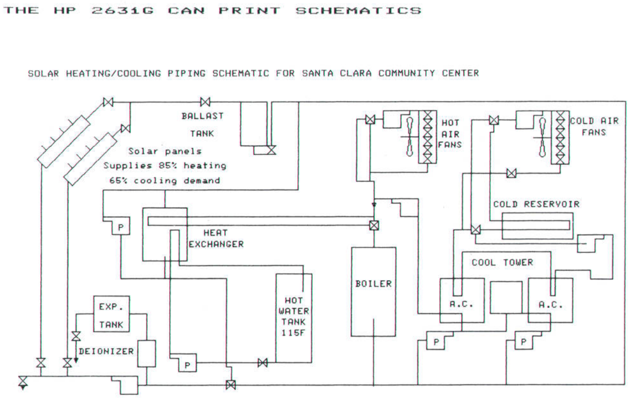
Graphics Operation

The new printer accepts graphics information in raster form line by line in 8-bit bytes, with the 0's and 1's in each byte representing the blanks and dots of a contiguous string of horizontal dot positions in a raster line. The printer accepts and stores eight lines of data in a buffer, and then prints the eight lines simultaneously while accepting and storing the next eight lines in a second buffer.

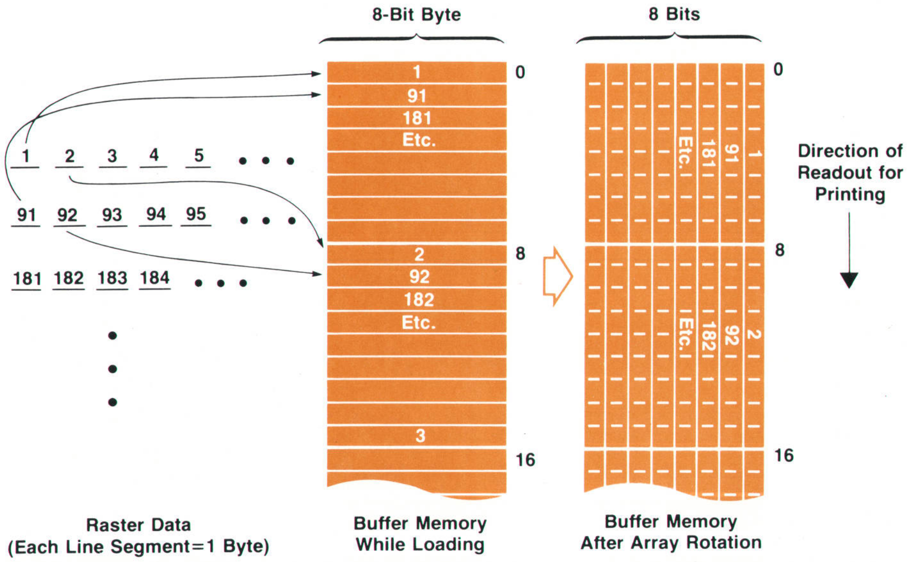
One of the problems that had to be solved was how to organize the data so the eight lines could be presented to the printing hardware in a usable form. This is done by regrouping the data as it is loaded into the buffer as shown below.

After the buffer is loaded, the data is withdrawn in arrays of eight 8-bit bytes. Each array is rotated 90° and then returned to memory. The word stored at each address in memory now has the proper dot pattern for printing the dots at the corresponding dot positions in all eight raster lines simultaneously.

This method of printing the dots required a change in the dot-generation control circuits (see Fig. 2, page 10). The character ROMs shown in Fig. 2 have been replaced by a RAM and the microprocessor loads the dot data from the graphics buffer into the RAM so the dot-printing circuits can access the data for printing as described on pages 8 and 9.



Another change was one made to the leadscrew encoder disc. The second track now has 144 slots so 72 dots per inch can be printed in the graphics mode, which matches the 72 dot-per-inch vertical resolution of the printer. Consequently, when printing alphanumerics the COMPRESSED print mode has a pitch of 14.4 characters per inch. By allowing the dot position counter to count every other dot, this can be expanded to give a



pitch of 7.2 characters per inch. The 2631G thus has four print widths (5, 7.2, 10, and 14.4 characters per inch).

Alphanumeric Operation

Dot patterns for printing alphanumerics are also supplied to the printing RAM by the microprocessor. The microprocessor accesses the character ROMs to find the dot patterns for the specified characters and then supplies them to the printing RAM.

Direct access to the character ROMs on the part of the microprocessor now gives a greater degree of flexibility in handling multiple character sets. Each character set now has an identifying symbol, and whenever one of these symbols is imbedded in the data stream, the microprocessor searches through all the character ROMs to find the location of the desired characters. This allows character sets to share the same characters. For example, European languages and English use the same Roman alphabet and differ only in the use of accents and a few special characters. Only the accents and special characters are stored in the European character ROM since the microprocessor accesses this ROM only when needed. Otherwise, it uses the English ROM. Thus, there is room to place special characters for several languages in a single ROM, giving the printer multilanguage capability without the need to change ROMs.

Reference

1. P.D. Dickinson, "Versatile Low-Cost Graphics Terminal Is Designed for Ease of Use", Hewlett-Packard Journal, Jan. 1978.

however, pass through the amplifier with little change.

The output of the slew-rate limiting amplifier drives a switching power amplifier that in turn drives the servomotor. If a linear amplifier were used, it would need to be capable of supplying at least 20V to drive the servomotor at high speed during carriage return. While printing at slow speed, however, only 5V may be needed but the motor may draw 2 amperes. A linear amplifier would thus need to dissipate (20 − 5) V × 2A = 30W. The switching transistors, on the other hand, dissipate only 2W each in this situation.

The error voltage node is set up so that if it is above a reference voltage (+4.6V), the voltage applied to the servo motor tends to accelerate the motor in the commanded direction. If the error voltage is below the reference, a deceleration voltage is applied.

The magnitude of the motor drive is obtained by rectifying the difference between the error voltage and the reference, thereby converting a difference voltage of either polarity to a unipolar voltage. This voltage is applied to a comparator which functions as a pulse-width modulator. The other input to the comparator is a 20-kHz sawtooth waveform, so the output of the comparator is a rectangular wave whose width is proportional to the error voltage. This waveform is applied to the motor control logic that controls the

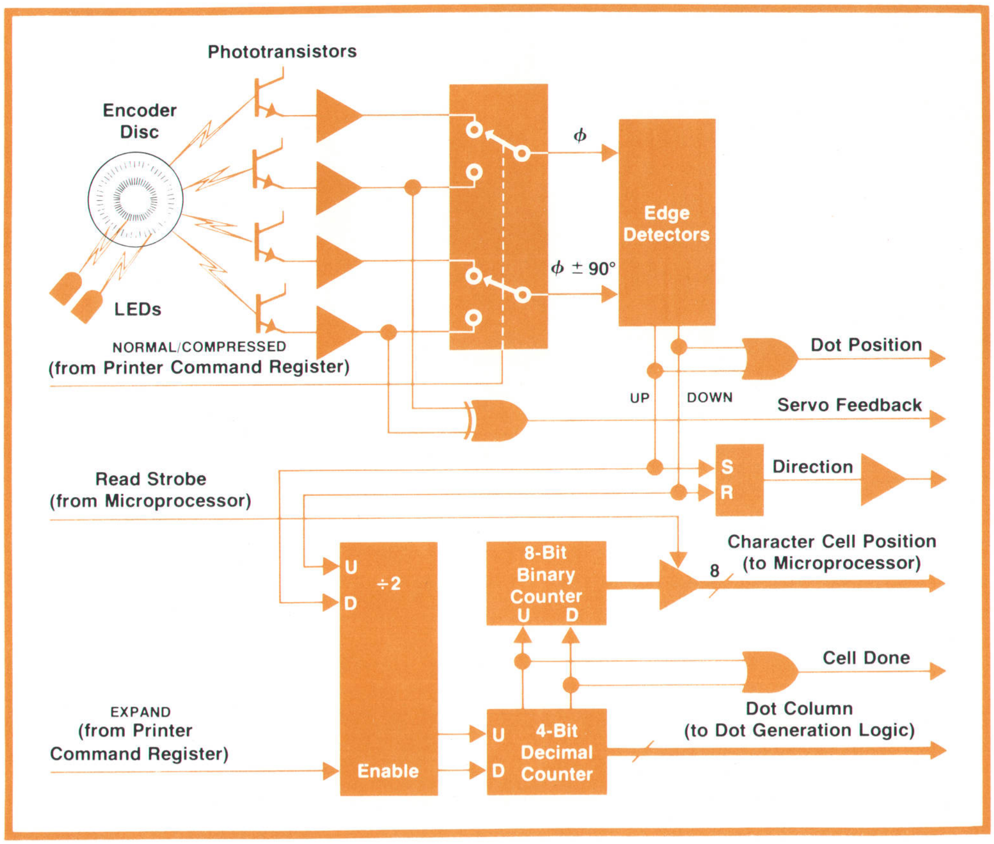


Fig. 4. *Circuits that determine the printhead position.*

power applied to the servomotor.

The polarity of the motor drive is likewise obtained by comparing the error signal to the reference. The resulting polarity signal is applied to the motor control logic along with the commanded direction signals. If the actual head direction differs from the commanded direction, one of two flip-flops will be set. The outputs of these flip-flops are connected to the motor control logic such that if the head is moving in the wrong direction, full power (100% pulse width) is applied to the servomotor to bring it to a stop. Once the motor changes direction, the flip-flop is reset and the servo loop returns to the linear mode.

Position Control

The circuits that determine the carriage position are shown in Fig. 4. The optical encoder disk attached to the leadscrew has two tracks that determine the printing pitch (horizontal spacing) of the characters. The standard track contains 100 slots per revolution to produce a pitch of 10 characters/inch (NORMAL mode). The second track has 167 slots per revolution (except in the Model 2631G Graphics Printer) and is used for COMPRESSED printing at 16.7 characters/inch. This second track is also used for velocity feedback in the carriage servo-drive system.

Each encoder track is sensed by two LED-phototransistor pairs that are spaced to produce two signals 90° out of phase with respect to each other. Edge detectors monitor the phototransistor outputs to determine the direction of printhead motion and generate an appropriate count pulse (an UP pulse for motion to the right and a DOWN pulse for motion to the left). These pulses set or reset the DIRECTION flip-flop. They are also ORed together to create the DOT POSITION pulse.

The UP/DOWN count pulses are also fed to an up/down counter chain that keeps track of the printhead position as the head moves across the page. The counter has a four-bit decimal-count stage that indicates the dot column within a character cell followed by a binary stage that contains the number of the cell. The counters keep track of the number of character spaces (logical print positioning) rather than the actual position of the head since the print mode (pitch) may change several times in a line. The microprocessor can read the cell number so it has access to this information when commanding the printhead carriage to move to any position along a line of print.

As explained previously, the dot-column data from the counter is used by the character generation logic as part of an address in the character generator ROM.

Several printing features are handled by the hardware. One is auto underline. When the microprocessor is commanded to invoke auto underlining, it sets a bit in the printer command register that replaces the data for the printhead's bottom hammer with the least-significant bit of the dot-column counter. Thus, dots are printed on the bottom dot position of the odd-numbered columns, forming an underline.

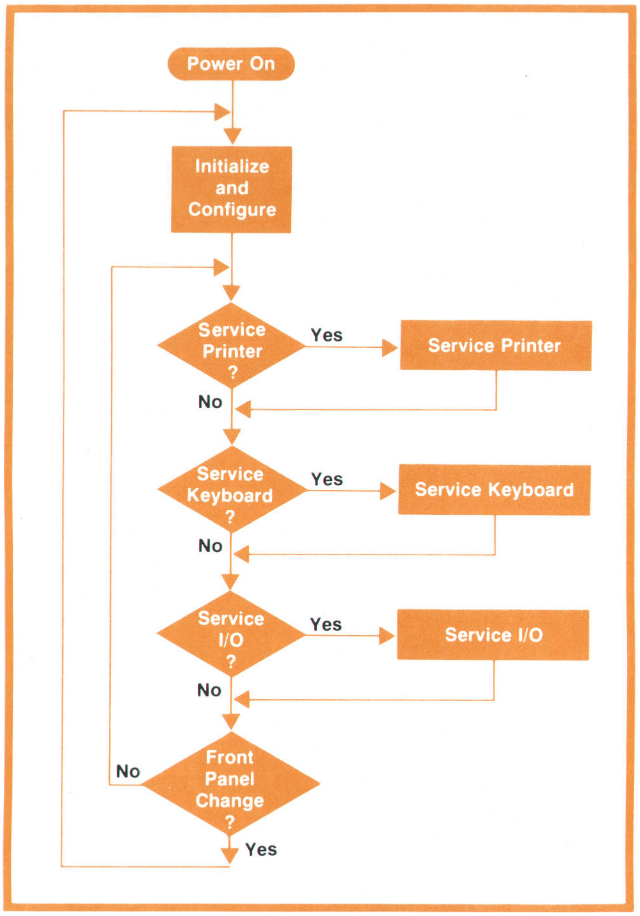


Fig. 5. *Basic idle loop executed by the microprocessor.*

The print mode (pitch) is determined by two bits in the printer command register. One of these bits determines which encoder track will supply inputs to the edge detectors (Fig. 4) and thus determines the choice of NORMAL or COMPRESSED modes. The other bit, used in the EXPAND mode in conjunction with the NORMAL track, enables ÷2 logic preceding the dot-column counter. This allows every other UP or DOWN pulse only to be counted by the dot-column counter, causing each column to be printed twice in succession.

System Firmware

The programs resident in firmware within the new printers and printing terminals are divided into five parts. These include the main system firmware, the printer driver firmware, the I/O driver firmware, the self-test firmware, and the keyboard/control-panel firmware. The driver portions are mainly concerned

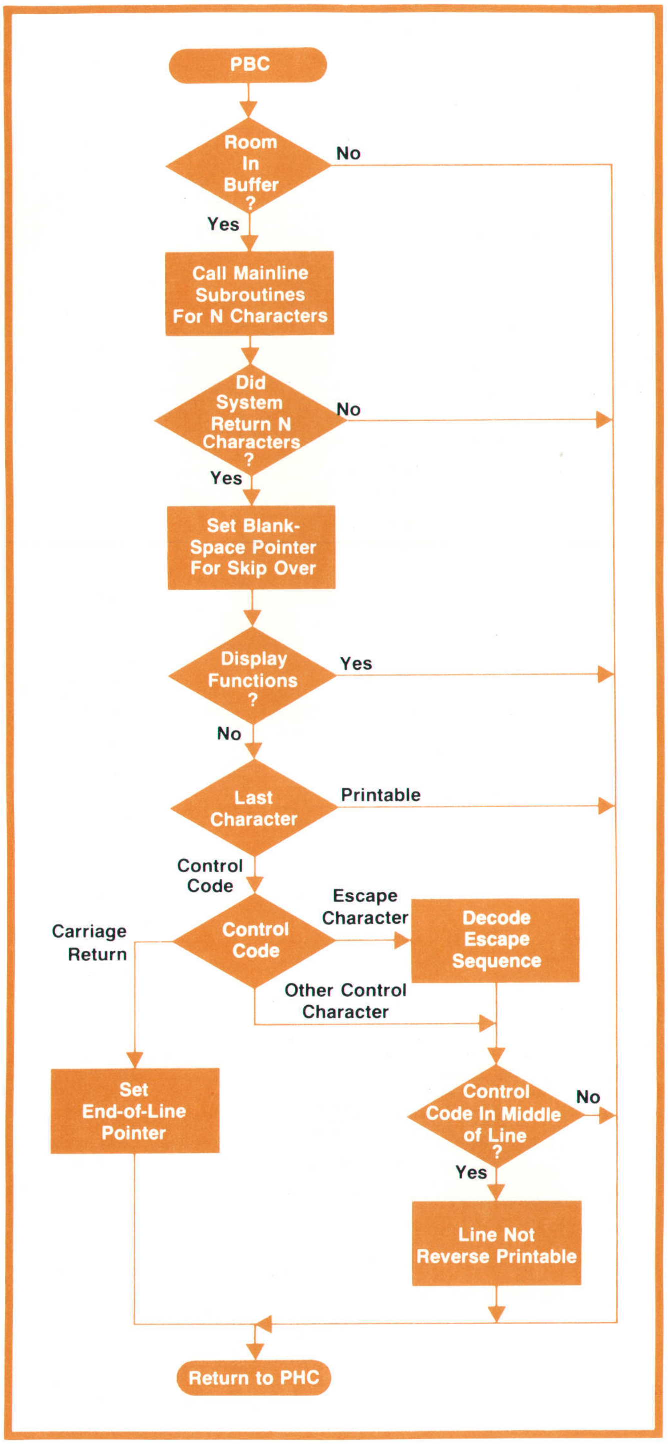


Fig. 6. *Printer buffer control (PBC) flowchart.*

with controlling hardware while the system firmware provides data-path handling functions and a common set of general utilities for the drivers. The total amount of memory is 4K 16-bit words of ROM and 512 16-bit words of RAM.

The system firmware normally executes a basic idle loop, as shown in Fig. 5, but is driven out of the loop by interrupts. When an interrupt occurs, the processor saves the present state of the machine and transfers control to the interrupting routine. Upon conclusion of the interrupt routine, the processor restores the previous machine state and resumes execution from the point of interrupt. If the interrupt does not require fast response, the processor merely sets a flag for the appropriate driver and continues with the main program. The flagged driver is subsequently called when the processor reaches that point in the main idle loop where the flag (called a service request) is tested.

The processor responds to three sources of interrupts. These include the interface (I/O) hardware, the printer hardware, and the 2-ms real-time clock. The printer and I/O functions are purely interrupt driven while the keyboard is scanned at 10-ms intervals using the real-time clock. The 10-ms scan time is fast enough to detect all keystrokes but slow enough to ignore switch bounce. The processor keeps track of the keys down on the previous scan to provide 2-key rollover.

Since the processor handles the keyboard scan and converts the ASCII codes for the keys by a table-lookup algorithm, the keyboard hardware is extremely simple and inexpensive. Yet, keyboard scanning and ASCII conversion require only about 2% of the processor's time.

System firmware functions are detailed in the following list.

- Provide a power-up initialization routine.
- Handle all interrupts; also provide a firmware mechanism using the real-time clock to allow any driver, such as move head to the right for viewing, to suspend itself for a specified time period.
- Call proper drivers as needed.
- Provide a queuing structure so drivers can use queues with system calls, implemented with source and sink drivers. For example, in local mode the keyboard driver is a source and the printer is a sink for the keyboard queue.
- Scan the remote/local and full/half-duplex switches and maintain internal data paths according to the selected mode.

Printing Firmware

The firmware that controls printing performs two general functions: monitoring and controlling the printing hardware, and interfacing information with the microprocessor operating system. The internal language consists of ASCII-coded eight-bit characters.

The print driver firmware has two major subroutines. One, the printer buffer control, receives characters and places them in a printer line buffer. The other, the printer hardware control, contains all

Interfacing to a 180-Character/ Second Printer/Terminal

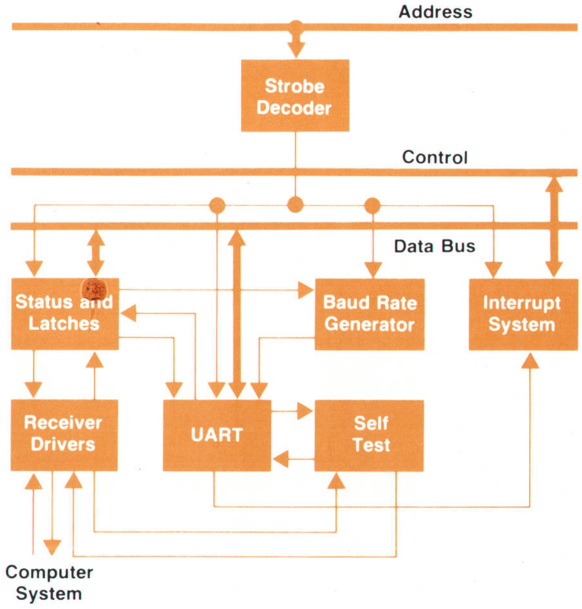
The Models 2631A/G Printers and 2635A/39A Printing Terminals were designed so that the interface I/O can be changed to match different I/O protocols simply by changing one printed-circuit board. To permit this, the machine's firmware is divided so that 1K of the 4K words of operating firmware is on the I/O pc board. Each I/O board thus carries its own firmware, making it easier to adapt the machine to different computer systems.

The self-test routines for the entire machine are also included in the I/O ROM so the self test can include tests for the particular I/O in use.

I/O cards have been designed for both serial and parallel data inputs. In either case, each character is transmitted as an eight-bit byte. Since the characters used by the machines can all be encoded in seven bits, the eighth bit may be used as a parity bit—options allow the parity protocol to be either odd or even—or it may be used to designate which of the machine's two character sets is to be used for that character.

To permit this I/O flexibility, there are seven entry points in the first seven addresses of the I/O firmware. These are:

1. I/O mainline. This is called by the machine's operating routine whenever service is required. It performs many functions based upon the flags set by other routines. It may also set its own service requests to reenter at a later time. After a particular service is performed, the I/O mainline clears the service request and checks for any more requests.
2. I/O power on. This routine, called once during reset, initializes any words or hardware prior to the machine becoming operational. It also brings up the hardware with the designated parity and baud rate.
3. Interrupt. It is in this routine, called by the I/O card itself, that characters are received and transmitted. In the case of the serial I/O, considerable processing is done by this routine but with the parallel I/O, because of its faster byte-by-byte transfer, the flag for the I/O mainline service request is set and the interrupt is handled later.
4. Scanner. This point is entered every few milliseconds as the operating system has time. The serial I/O uses this point to monitor outside control lines while the parallel I/O returns immediately to the operating system.



5. Self-test I. Directs the I/O firmware to a routine that tests the I/O hardware. The results of the test are returned by way of an internal register to the calling program.
6. Self-test II. This causes the I/O to power up to resume normal communications with the outside world following the self test.
7. I/O status. This routine is called when the machine has received an external request for machine status.

Serial I/O

Serial data is accepted asynchronously, that is, the data is not synced with the machine's internal clock. When no data is being transmitted between the machine and the computer, the communication line is held in a marking (logic high) state. When a character is to be transmitted, the line is put into the space (logic low) state for one bit period and then the eight bits are transmitted, least significant bit first. After the eight bits, the line goes back to the mark state and remains there until the next character is transmitted.

Although the new machines are capable of printing at a speed of 180 characters per second, they may be used with computers that transmit at rates up to 960 characters per second. To cope with high data rates, the approach here is to use the ENQ/ACK handshake (enquiry/acknowledge), as used with the HP 2640 series CRT Terminals. With the ENQ/ACK system, the computer sends 80 characters followed by the ASCII ENQ character (05_{16}). The characters are stored in the machine's I/O buffer and when the ENQ character is received, the machine checks the buffer to see if there is room for 80 more characters. If there is room, it transmits the ACK character (06_{16}) and the computer then sends another 80 characters followed by the ENQ character. If there is not room, the machine delays sending the ACK character until room for 80 more characters becomes available.

The ENQ/ACK system makes more efficient use of computer time than do systems that intentionally slow the character transmission rate by inserting non-printing characters to slow the effective rate.

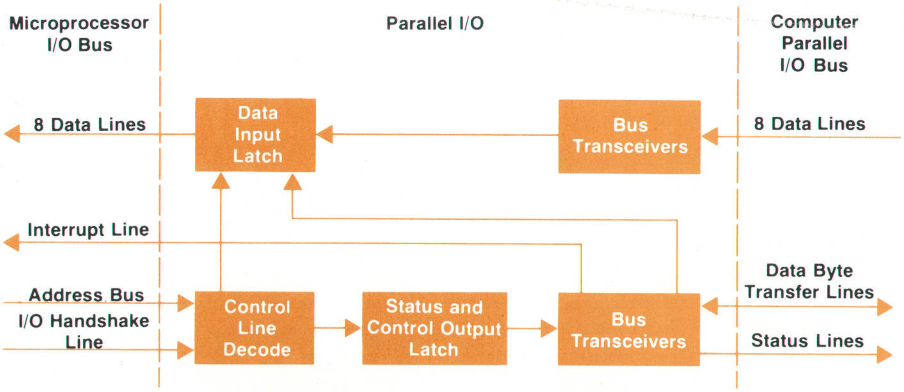
A block diagram of the I/O hardware is shown at left. Operation is controlled by the strobe decoder that decodes the I/O's read and write pulses for controlling data bus use. The heart of the hardware, however, is the UART (universal asynchronous receiver/transmitter). It receives the serial data, checks the parity if used, deletes the start and stop bits, and presents the character bits in parallel to the main data bus. Another portion of the UART accepts a character to be transmitted, adds start, stop, and parity bits, and transmits them.

The self-test block allows the UART to transmit characters to itself, thereby testing the UART and the circuits that support it. Incoming data is ignored and outgoing data is suspended during the I/O self test.

The interrupt system generates an interrupt for the main processor whenever the UART receives a character or becomes ready to transmit another character. It also identifies itself to the processor as the interrupt generator.

The block labelled Status and Latches accepts and latches control signals from the data bus for controlling the outgoing lines and other functions on the board. The status portion transmits configuration information, set into DIP switches on the I/O board and read by the I/O firmware, to the data bus for controlling protocol and various operating modes.

(continued)



It is in the firmware interrupt routine that characters are received and transmitted. Upon entry to this routine, the UART is immediately checked for a character, since they could arrive as often as one per millisecond. Incoming characters are checked for parity and framing errors and whether or not they are ENQs. If an ENQ is received, a flag is set for the transmitter routine to check for buffer room and send an ACK. After the character is checked, it is put into the system queue and the transmitter routine is entered. In the transmitter routine, special characters like ACK, STX, and ETX, and terminal status are checked first, then the system queue is checked for characters.

Parallel I/O

In the standard parallel interface, the machine's processor handles the input data on a byte-by-byte basis, controlling all handshakes through the control line decode circuit, shown in the block diagram above. Good service response is required on the part of the processor to insure accurate data transfer. The CMOS SOS microprocessor used in the machine is capable of doing this.

Upon receipt of a byte, the processor services the interrupt and sets up a service-request call to the I/O from the main operating system. When the next free time occurs, the data input latch is read and the interface is prepared for another byte of data.

On every service, the processor loads the status latch so the latch always contains the current status, in case the system computer should request status.

HP-IB Interface

The HP-IB interface I/O is basically similar to the standard parallel I/O except that the data input and output latches are replaced by a CMOS SOS integrated circuit.[1] This IC, together with bipolar three-state transceivers, provides a complete logical and electrical interface to the interface bus defined by IEEE-488-1975. It handles the bus protocol and handshakes involved in data transfer and provides buffering for inbound and outbound data transfer through two FIFO queues that can be accessed by the machine's processor. Up to eight bytes of data can be transferred to the I/O before an interrupt requiring service from the processor is generated.

In this interface I/O, the control line decode enables the IC when the machine's processor talks to it, and it also enables a switch that places the HP-IB address, set by a switch on the rear panel, on the processor's data bus. Another switch places the I/O in the listen-always mode so the machine will respond to whatever is on the HP-IB bus.

The HP-IB I/O can operate at high speed up to the limit of the HP interface bus (1 megabyte/second) or it can operate in the print-and-space mode without any loss of printer capability.

–Gerard Carlson
Michael Lee
Roy Foote

Reference

1. J. W. Figueroa, "PHI, the HP-IB Interface Chip," Hewlett-Packard Journal, July 1978

printing firmware overhead as well as control of hardware functions. Both subroutines execute as non-interrupting routines and comprise about 95% of the printing firmware. The remainder involves paper feed and buffer-to-printhead subroutines that operate under interrupt control to service time-critical functions rapidly.

Separation of the driver into two sections evolved naturally because of the timing differences between the processing of characters by the printer buffer control (PBC) and their printing by the printing hardware control (PHC). The PBC first writes characters to the line buffer and then the PHC writes the buffer to the printhead hardware for printing. Both routines maintain separate configuration states that may affect logical decisions.

Printer processing begins when the PBC calls a routine in the microprocessor operating system specifying the number of character spaces available for the next line and the starting location in the line buffer (Fig. 6). The routine then transfers characters stored in the input queue to the line buffer. Transfer stops when the number of characters requested is transferred, or when the entire queue is transferred if less than the requested number, or if a control character is received. The PBC examines the last character to see if it is a control code and if so, determines whether or not it is executable by the printing mechanism. Non-executables—for example, "inverse video"—are eliminated, printables require no further attention, while executable control codes, including escape sequences, cause additional processing. Because con-

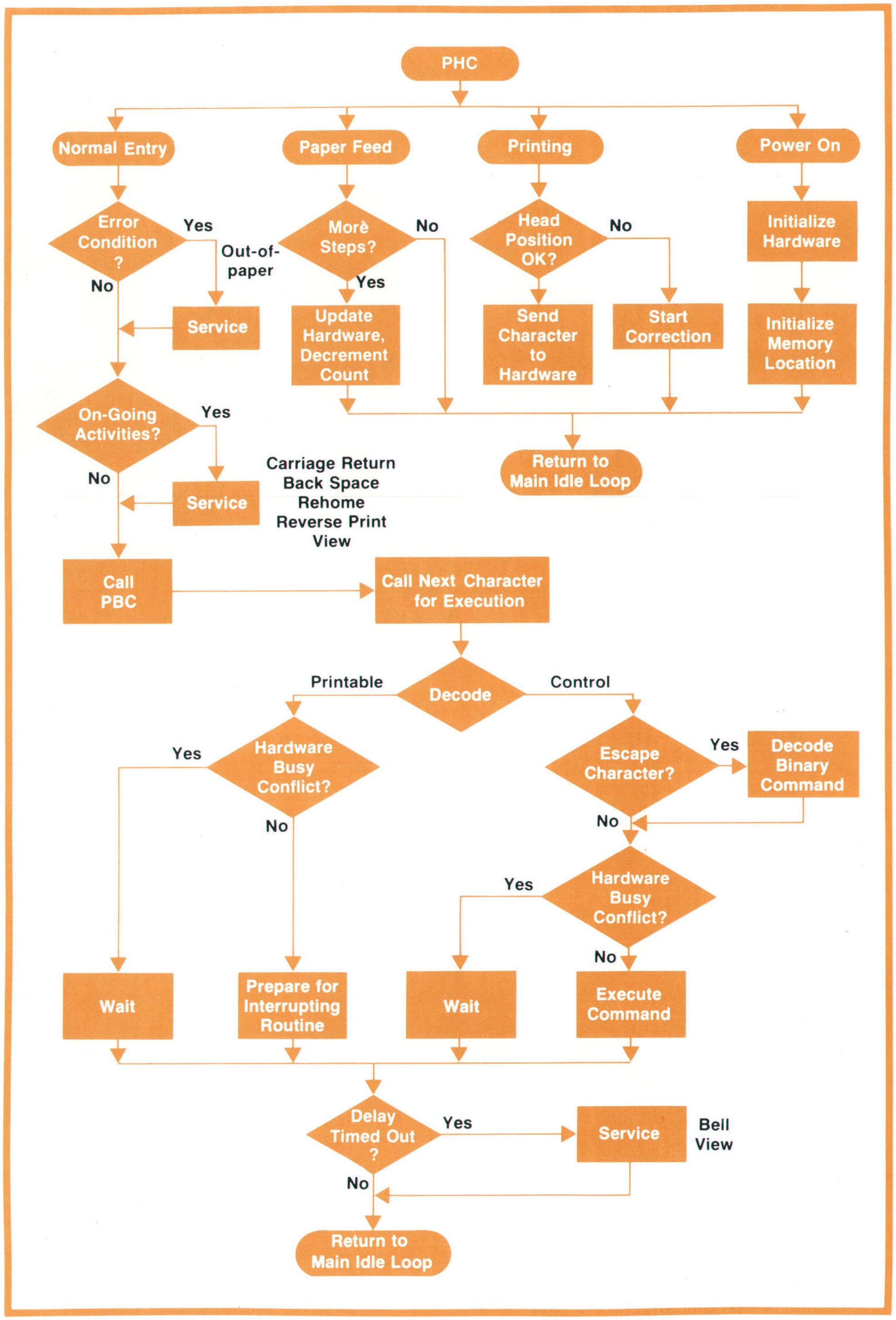


Fig. 7. *Printer hardware control (PHC) flowchart.*

trol codes can affect characters that follow, print lines with imbedded executable control codes cannot be printed right to left. If the machine is in the DISPLAY FUNCTIONS mode, the original data stream remains unaltered so control codes can be printed.

Escape sequences are fully decoded and replaced in the line buffer by an escape character followed by an eight-bit binary code that represents the specific command. The line is terminated when the carriage-return character is encountered. This end-of-line condition, which is marked by a pointer, prevents further loading into the buffer until printing of the present line is completed.

If the PHC is printing left to right, characters are removed from the buffer left to right and the PBC fills in the next line as space becomes available. The PBC examines the incoming data and places a buffer pointer where the first non-blank character appears.

Thus, the printhead can be moved directly to this position when this line is printed, ignoring the blank spaces. The PBC also performs the tab function by inserting blank characters in the buffer.

When the PHC reaches the end-of-line pointer for the present line, a last call is made to the PBC in an attempt to complete the next line. If the next line is complete and is right-to-left printable, trailing blanks are stripped off and the distances from the printhead to the start and end of the next line are determined. If the head is closer to the end of the next line, a reverse print line is started.

Left-to-right print lines proceed through the buffer at print time until the end-of-line pointer is encountered. Right-to-left printing starts at the rightmost printable and proceeds back towards the first printable in the line. When right-to-left printing is completed, the buffer readout pointer is moved to one position to the right of where the rightmost printable was. Control codes at the end of the line are then processed by normal left-to-right processing.

The actual printing and execution of control sequences are handled by the PHC (Fig. 7). First, possible error conditions are checked, then existing tasks are serviced, then the next character is processed, and finally real-time clock counters (out-of-paper alarm, move head for viewing) are checked.

Entry to the main portion of the driver is by way of the system poll loop when the printing service-request flag is set. This flag can be set by a hardware interrupt (printhead crossing a cell boundary), by a printer delay word timing out, or when a character is placed in an otherwise empty system queue. On each pass through the driver, directly after each PBC call, a routine is called to get the next character from the line buffer. The next print location in a printed line is maintained in memory and the actual head position is in the printhead position counter. The head is then commanded to the character position one to the left of the next print position, if printing left to right, or one to the right if printing right to left. From there it is commanded to move in the proper direction at the indicated print speed. When the print velocity command is strobed to hardware, the PHC enables the PRINT GO flip-flop (Fig. 2) to be set by the next character position interrupt. With this interrupt, the printer interrupt routine is entered. This routine checks for the correct print position, then strobes the character to hardware for printing. The next-printing location is then updated and the PRINT GO flip-flop control bit is lowered so it can be cleared when the head reaches the end of the current print cell. The main entry is called again during the next system poll so that contiguous printables can be serviced.

The next-character routine skips over blank characters by simply incrementing or decrementing the next-print location in memory. If more than ten blanks occur in succession, the head is moved rapidly to the next printing location upon completion of the present printing character.

The PHC acts as a position controller for the print head carriage by giving a velocity command to the hardware and then reading the position of the head as feedback for updating the velocity input. Print speed is a function of the present print mode and the number of characters to be printed. The print-speed routine is called each time through the driver. As inputs, it uses the present print mode, the state of the repeat key (in the printing terminals), and the number of characters left to be printed. This enables lower speeds for keyboard entry, a repeat speed for repeat character input, or full speed, when sufficient characters are buffered, with controlled deceleration at the ends of lines.

Like the PBC, the PHC maintains its own DISPLAY-FUNCTION-mode state so that control codes can be printed and not acted upon. The exceptions are the carriage-return character and the command that terminates DISPLAY FUNCTIONS. When a carriage return character is encountered, it's symbol is printed and then a return and a line feed are executed. In the same way, the terminate-DISPLAY-FUNCTIONS symbol is printed and then that mode is terminated.

The four bits maintained in memory that control the printing mode (auto-underline, alternate character set, COMPRESSED and EXPANDED print modes) are updated each time their escape commands are encountered. Then, with each print strobe, a copy of the printing mode state is written to hardware so printing modes are selected on a character-by-character basis. This allows intermixing of character sets and print modes on a given line.

Paper Control

Paper motion can be commanded by a line-feed character, a form-feed character, or a vertical format control (VFC) command. VFC commands are escape sequences that select one of eight possible predefined position stops on a page.

When a paper motion command is received (line feed, form feed, etc.), the microprocessor allows the in-progress printing operations to complete and then initiates paper feed by triggering a state machine that generates the phased signals for advancing the motor to the next step. The paper feed routine calculates the number of steps required to move the paper to the next stop position. This number is stored in a memory location for step counting. Similarly, a line-feed or a form-feed causes the memory to be updated with the number of steps required.

Once the memory is loaded, a 2-ms real-time clock delay is set. When the delay times out, the paper-feed interrupting routine is entered. This routine enables the hardware to step the paper drive, decrement the paper-feed memory count, and check it for zero. If it is not zero, another 2-ms delay is set for the next step.

Each step advances the motor 1/72 inch so twelve steps are required for the standard six lines/inch spacing or nine for eight lines/inch. Firmware counters maintained by the microprocessor keep track of the line count. Other line spacings of one, two, three, four, and twelve lines/inch can be programmed.

SPECIFICATIONS

HP Models 2631A and 2631G Printers, 2635A and 2639A Printing Terminals

PRINTING
TECHNIQUE: 7×9 dot matrix, impact.
SPEED: 180 characters/second; bidirectional.
LINE LENGTH: 136 characters at 10 cpi, 68 at 5 cpi, and 227 at 16.7 cpi.
CHARACTER SET: 128 ASCII.
FORM WIDTH: Edge perforated, 31-mm (1.22 in) perforation-to-perforation, to 400-mm (15.748 in) edge-to-edge.
FORMS HANDLING:
Forms tractors are standard; feed rate: 176.6 mm/s (6.95 in/s).
8-channel fixed vertical form control (for 11-inch long forms).

- Single space
- Double space
- Triple space
- Slew to bottom of current page
- Slew to top of next page
- Slew to next half page
- Slew to next quarter page
- Slew to next tenth page

Horizontal tabbing, up to 227 tabs.
Variable horizontal and vertical pitch.
1-6 copies up to 0.43-mm total thickness.
POWER: 100, 120, 220, 240 Vac, +10%, −12%, 48-66 Hz, 140VA max nonprinting, 265VA max printing.
RELIABILITY:

An extensive reliability testing program was implemented in which 330 million characters were printed on each of 35 machines for a total of 11.5 billion characters. Preliminary results show that the mean number of printed characters between failures (MCBF), excluding print head and ribbons, exceeds 300 million. The printhead itself is good for an average of 130 million characters, and the average ribbon life is 10 million characters.

The 2630 family is recommended for 1 million characters per day, resulting in a mean time between failures (MTBF) of approximately 1 year. Basic monthly maintenance charges are based on this level of usage.

PRODUCT SAFETY: These products are listed by Underwriters Laboratories Inc. in the following categories:
Electronic Data Processing Equipment (EMRT)
Office Appliances and Business Equipment (QAOT)
Teaching and Instruction Equipment (WYFW)
They are also certified by the Canadian Standards Association as Data Processing Equipment.
ENVIRONMENTAL
TEMPERATURE: 0 to 55°C (32 to 131°F) operating.
HUMIDITY: 5% to 95% (non-condensing) excluding media.
VIBRATION: 0.38 mm (0.015 in) p-p, 5-55 Hz, 3 axes.
SHOCK: 30g, 11 ms, ½-sine shock pulse. Also type tested to qualify for normal shipping and handling in original shipping container.

HP Model 2631A Printer Only

SIZE: 215 mm H × 640 mm W × 469 mm D (8½ × 25¼ × 18½ in).
WEIGHT: 23.5 kg (51 lb) without stand.
OPTIONAL SECOND CHARACTER SETS: Swedish/Finnish, Norwegian/Danish, French, German, Spanish, United Kingdom, Cyrillic, Katakana, Extended Roman, Math symbols.
CONTROLS AND INDICATORS
PRINT MODE switch: EXPANDED, NORMAL, COMPRESSED.
VERTICAL PITCH switch: 6/8 lpi (sets default value). 1,2,3,4 and 12 lpi selectable through escape sequences imbedded in data stream.
DISPLAY FUNCTIONS button and indicator (prints control characters but does not execute them).
ON LINE/OFF LINE button and indicator.
SELF-TEST button
FORM FEED button
RESET button.
LINE FEED button.
POWER ON indicator.
POWER switch, fuse, and line voltage selector on rear panel.
PAPER OUT audible alarm.
INTERFACE:
STANDARD: General purpose differential line driver compatible with HP 12845B interface board for 2100 and 1000 series computers.
OPTIONAL: General-purpose TTL interface for HP 2640 series CRT terminals, RS232C/CCITT V.24 with or without modem control, 20 mA current loop, and IEEE 488-1975 interface bus.
PRICE IN U.S.A.: $3150.

HP Model 2631G Graphics Printer

Same as Model 2631A Printer except that it accepts raster-format graphics display data from HP 2647A and 2648A CRT Graphics Terminals, has a more flexible vertical forms control, can have multiple secondary character sets, has available an optional line-draw character set and an optional high-density character set.

HORIZONTAL PITCH: 5, 7.2, 10, and 14.4 characters per inch.
RESOLUTION IN GRAPHICS MODE: 72 × 72 dots/inch.
INTERFACE: IEEE 488-1975 interface bus.
PRICE IN U.S.A.: $4250.

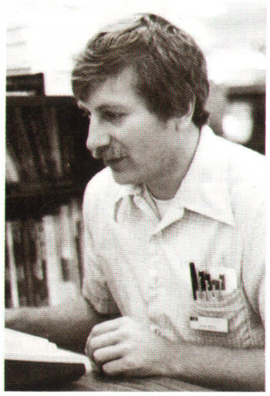
HP Model 2635A Printing Terminal Only

SIZE: 215 mm H × 640 mm W × 595 mm D (8½ × 25¼ × 23 in).
WEIGHT: 25½ kg (56 lb).
KEYBOARD: Full ASCII keyboard with typewriter compatible key locations. Caps lock and repeat keys. 15-key numeric pad. Enhanced 2-key rollover.
CONTROLS AND INDICATORS
TERMINAL RESET Button
SELF TEST button
SET TAB button
DISPLAY FUNCTIONS
DATA SET READY indicator
REMOTE/LOCAL Switch
AUTO LINE FEED button
CLEAR TAB button
POWER ON indicator
CLEAR TO SEND indicator
POWER switch, fuse, and line voltage selector on rear panel
INTERFACE: RS232C asynchronous interface for type 103 modems.
DATA RATE: 110, 150, 300, 1200, 2400 baud switch selectable.
TRANSMISSION MODE: Full or half duplex, switch selectable.
PARITY: odd, even, or none, switch selectable.
OPTIONS: 20-mA current loop (full duplex only) and RS232C for type 202 modems.
PRICE IN U.S.A.: $3450.

Model 2639A Printing Terminal

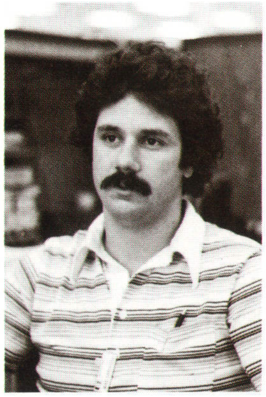
Same as Model 2635A except has flexible interface that makes RS232C and current loop selectable with split transmit/receive baud rates, and also has asynchronous ASCII interface with standard 1-K byte input buffer; "printer busy" state can be selected at 100, 90, 80, 70, 60, 50, 40, or 30 percent of buffer full, or "printer not busy" state can be selected at 0, 10, 20, 30, 40, 50, 60, or 70 percent of buffer full. Supports either X-ON/X-OFF or ENQ/ACK protocols.

PRICE IN U.S.A.: $4025.
MANUFACTURING DIVISION: BOISE DIVISION
11311 Chinden Boulevard
Boise, Idaho 83707 U.S.A.



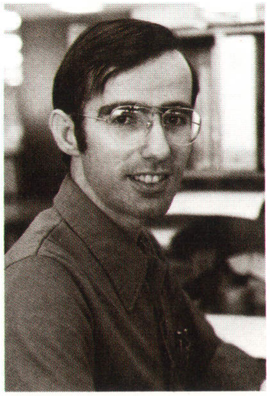
Kenneth B. Wade

With HP since 1975, Ken Wade was responsible for the main firmware for the 2630 family. Prior to HP, Ken worked in logic and software design. Born in Aberdeen, Washington, he received his BSEE degree from the University of Washington in 1970 and his MS in computer science from the University of Santa Clara in 1978. Ken lives with his wife and three children in Boise, Idaho, and is involved in church activities, cub scouts, basketball, volleyball and chess.



Michael J. Sproviero

An HP employee since 1972, Mike Sproviero is project leader for the 2631G Printer. He started as production engineer on disc and magnetic tape drives at HP's Mountain View Division, then moved to the Boise Division. Born in Gustine, California, he received his BSEE degree in 1972 from the University of Santa Clara and MSEE degree in 1978 from the University of Idaho. Mike lives in Boise, Idaho, with his wife and son, and enjoys tennis, camping and skiing.



Phillip R. Luque

Phil Luque joined HP's Santa Rosa Division in 1973 where he worked on the 8754A network analyzer before moving to to HP's Boise Division in 1975 where he designed the lead screw servo for the 2631A/G printer. Born and raised in Boise, Idaho, Phil is a 1972 BSEE and a 1973 MEEE graduate of Brigham Young University. Phil lives with his wife and three children (ages 1, 3 and 5) in Boise, and keeps active doing church work, folk dancing and working around the house.

John J. Ignoffo, Jr.

John Ignoffo was electronic hardware designer for the 2631A/G printers and also designed the character sets. With HP since 1975, John is a 1971 BSEE and a 1973 MSEE graduate of the University of Santa Clara. Between graduation and HP, he worked in printer research. Born in San Francisco, John was recently married and lives in Boise, Idaho. He is fond of skiing, and listening to music, and is a member of the board of directors of a local theater group.

Versatile 400-lpm Line Printer with a Friction-Free Mechanism that Assures Long Life

This medium-speed line printer writes dot-matrix alphanumerics and graphics with a mechanism that has no sliding parts or bearings to wear out. It's also versatile, with a capability for printing with a variety of character sets.

by F. Duncan Terry

THE TRANSFER OF INFORMATION from a computer to a printed page can be done with any of several different techniques at various speeds in a variety of modes. For most small computer systems, especially those directed towards the scientific and small business markets, the impact printing technique is favored because it allows printing on multicopy forms using standard paper. Impact printers commonly used in small systems print with speeds ranging from 120 characters per second to as high as 1000 lines per minute.

Described in the previous articles in this issue is a serial-character printer (Model 2631A) using a horizontal-scan, dot-matrix, impact printing technique capable of printing at speeds of 180 characters/second. This article describes a new medium-speed line printer (Model 2608A) that uses a combined horizontal- and vertical-scan, dot-matrix technique to achieve print speeds of up to 400 lines per minute, equivalent to nearly 800 characters per second. With this speed, the new printer (Fig. 1) can fulfill the majority of printing needs in the small computer field. It also offers the system programmer a new dimension in system interaction with a line printer.

Basic Operation

The printing technique used in the new printer is similar to that used in the HP Model 2607A Printer but a new mechanism was designed to achieve higher speed and greater ruggedness. The basic printing operation is the energizing and release of a cantilever steel tine (hammer) by an electromagnet (Fig. 2). To print a dot, the magnet draws the tip of the hammer away from the paper and then releases it. When released, the hammer snaps forward and a tungsten-carbide sphere welded near its tip impacts the ribbon against the paper and platen, printing a dot 0.45 mm (0.018 inch) in diameter.

To print a complete character, the hammer and magnet are moved horizontally five dot positions, and dots are printed in the positions required by that

Fig. 1. *Model 2608A prints 132 columns of characters on standard 14-inch wide computer paper at speeds up to 400 lines/minute when printing upper case only with a 5 × 7 dot matrix, or to 320 lpm when printing upper and lower case with a 5 × 9 dot matrix. As many as 16 different character sets can reside within the printer, any two of which may be used within any line, and the normal dot matrix can be expanded for special print characteristics. Forms with as many as six parts can be printed.*

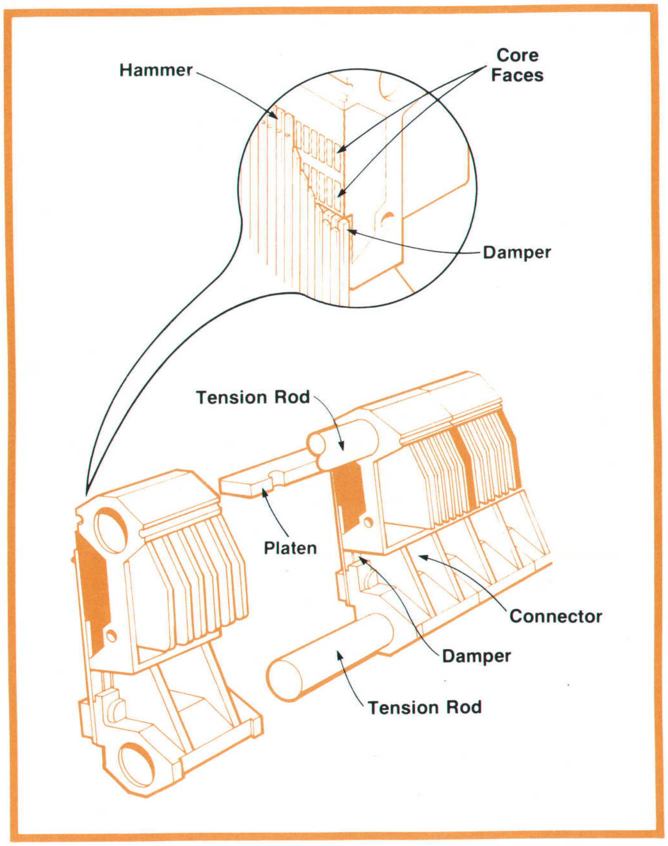


Fig. 2. *Print hammer mechanism. Eleven modules holding twelve hammers each are clamped together by two tension rods to form a rigid bar that is moved back and forth by a linear (voice-coil) motor.*

character in that particular row of the dot matrix. The paper is then advanced 0.353 mm (0.0139 inch), the hammer and magnet reverse direction, and the next dot row of the character is printed. Seven rows of dots complete an upper-case character (Fig. 3). For lower case characters, two more dot rows print the descenders, if required.

The mechanism has 132 hammers, spaced on 2.5-mm (0.1-inch) centers, that move horizontally as a unit. Any combination of hammers can be fired simultaneously, depending on the information being printed. The paper is pulled through the machine in discrete steps, one dot row at a time, and after each advance, the hammer and magnet assembly moves horizontally to print all the dots in that row for all the characters on that line.

New Directions

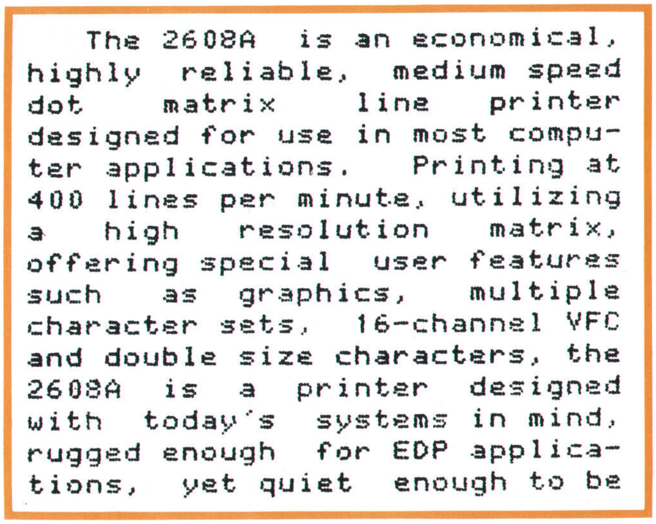
This technique has proven to be versatile, reliable, and cost-effective. It has, in fact, proven to be capable of generating some of the most uniform and readable multicopy print from any type of printer because each dot is printed with the same intensity, a characteristic that is most noticeable when several carbon copies are being made. This is in contrast to full-font printers where a low-density character, such as a period (.), receives the same impact force as a high-density character, such as the number sign (#), with resulting variations in print intensity.

It was felt, however, that the basic mechanism had a potential for higher speed and also for printing graphics. In addition, the use of a microprocessor for control could add a high degree of versatility. Therefore, a major redesign was initiated. To take full advantage of the basic reliability and flexibility of this method of printing, extensive analysis of the printing action was undertaken, leading to major design changes and improvements in the mechanism. The electronics were completely redesigned.

During the design phase, the project focused on three main objectives.

The first was to fully exploit the fundamental advantages of the print mechanism.

High reliability is characteristic of the print hammers because each is a simple mechanical spring with no bearings or sliding parts to wear out. This concept was extended to the horizontal movement of the hammer assembly, which is now mounted on metallic flexures, achieving a virtually frictionless print mechanism. Print uniformity is enhanced by moving the electromagnets horizontally with the hammers rather than moving the hammers only. This maintains a uniform magnetic field and reduces the problem of cross-talk between adjacent characters at the extreme excursions of the hammer assembly. It also allows the hammers to move a greater distance so dots can be placed in the between-character spaces for printing graphics. Also, the hammer assembly is now moved by a friction-free voice-coil-type linear motor, which allows the stroke length to be changed easily for printing graphics. Print speed was more than doubled by



```
   The 2608A  is an economical,
highly  reliable,  medium speed
dot     matrix     line    printer
designed for use in most compu-
ter applications.   Printing at
400 lines per minute, utilizing
a   high    resolution    matrix,
offering special  user features
such    as   graphics,    multiple
character sets,  16-channel VFC
and double size characters, the
2608A   is   a   printer   designed
with   today´s   systems in mind,
rugged enough   for EDP applica-
tions,  yet quiet   enough to be
```

Fig. 3. *Dot-matrix printing performed by the Model 2608A Line Printer is reproduced full-size here.*

driving the hammers near their resonant frequency, by improving the hammer damping characteristics, and by using better materials.

The second was to enhance system flexibility in a cost-effective manner.

Microprocessor control made it feasible to offer multiple character sets, double-sized characters, full-scan graphics, overstriking for underlining or creating APL characters, and a 16-channel vertical format control (VFC) that is fully electronic and can be initialized either from ROM or by software. Self-test routines exercise all features of the printer to give assurance that the printer is functioning properly. A number of status indicators are now available to the host system.

The third was to enable the printer to fit in a wide variety of environments.

The compact appearance disguises the fact that the new printer is a rugged, workhorse machine. Low acoustical noise (72 dBa standard, 68 dBa optional) allows its use in systems that are to be placed in sensitive environments. The 100-metre (~100-yard) ribbon, capable of printing 30 million characters, is contained in an easily changed, smudge-free cartridge. Modular construction minimizes maintenance costs.

These and other characteristics are described in more detail in the article that follows.

Acknowledgments

Many thanks are due Jim Barnes, R and D manager, who guided and supported the project from its inception, and to Don Bowman, who was project leader until transferring to production. In addition to contributions from those mentioned in the following article, significant contributions were made by Tom Baker, who did major portions of the mechanical design and Randy Mazzei, who did the industrial design. Ralph Tenbrink and Al Olson worked out the difficult manufacturing processes. Lamar Goats performed the reliability verification testing and Gary Gapp did the product performance testing. Other contributors were John Meredith, Mike Harrigan, Ross Casey (mechanical), Tom Holmquist (electrical), and Darrell Cox (ribbon).

F. Duncan Terry

With HP since 1969, Duncan Terry was responsible for the design of the 2630 family of printers and the 2608A line printer. Prior to joining HP's Boise Division, Duncan designed logic state analyzers at HP's Colorado Springs Division. The author of several papers on nuclear radiation and electronics, he is also named inventor on two patents relating to test instruments. Born in Lander, Wyoming, Duncan received his BSEE degree in 1962 and his MSEE degree from the University of Wyoming. Duncan is married, has two daughters (ages 14 and 11) and in his off hours, he spends his time camping, hunting, hiking and working on home electronics.

SPECIFICATIONS

HP Model 2608A Line Printer

PRINTING

TECHNIQUE: Impact

CHARACTER FORMATION: Dot matrix, 5×7, 5×9, 7×9.

GRAPHICS MODE: 924 dots per line at 70 dots per inch × 72 dots per inch vertically. In the graphics mode, the printer interprets each data byte as eight horizontal dot positions and prints dots where 1's occur e.g. 323_8 = 11010011 = ●● ● ●●).

LINE LENGTH: Up to 132 characters.

PRINT SPEED

MATRIX SIZE	LINES PER MINUTE
5×7	400
5×9	320
7×9	250

40 dot rows per second with the maximum 924 dots per line in the graphics mode.

LINE FEED RATE (6 or 8 lines per inch): 15 ms.

FORM FEED RATE: 357 mm (14 in) per second.

VERTICAL FORMAT CONTROL: Electronic control, 16 programmable channels. Standard channel definitions and assignments are available from internal ROM, or forms may be defined and channels assigned by an external data source.

FORM WIDTH: 130 mm (5 in) to 385 mm (16.16 in) edge-to-edge.

PAPER WEIGHT:

SINGLE PART: 20 lb up to 100 lb.

MULTI-PART: Up to 6 copies with 12 lb paper and 7 lb carbons (0.61 mm maximum pack thickness). Multipart forms and card stock should be tested for satisfactory feeding, registration, and print quality.

POWER: 100, 120, 220, 240 Vac +5%, −10%, 48-66 Hz, 700 VA typical, 1500 VA max when printing; 225 VA non-printing.

SIZE (including stand): 1042 mm H × 680 mm W × 555 mm D (41 × 26½ × 22 in)

WEIGHT (including stand): 97 kg (215 lb).

ENVIRONMENTAL

TEMPERATURE

PRINTER: 0 to 55°C (32 to 131°F) operating; −40 to 75°C (−40 to 167°F) non-operating.

RIBBON: 10 to 50°C (50 to 122°C) in use or in storage.

RELATIVE HUMIDITY: 5% to 95% non-condensing. Forms should be tested at high humidity for satisfactory feeding and handling. At low humidity, forms should be tested to determine if static build-up should be eliminated for proper stacking.

AUDIBLE NOISE (using ISO 3744 as measurement standard): 72 dBa operating, 55 dBa in standby. Optional sound cover reduces operating noise level to 68 dBa.

CONTROLS AND INDICATORS

POWER ON indicator
ON LINE/OFF LINE button and indicator
FAULT CONDITION indicators
  PRINT MECH
  PLATEN RIBBON
  PAPER OUT
  TEST FAIL
FORMS ADJUST UP/DOWN buttons
POWER ON/OFF switch on rear panel
POWER ON CONDITIONS
  6 or 8 LPI
  PRIMARY LANGUAGE (1 of 16)
  SECONDARY LANGUAGE (1 of 16)
FORM FEED button
LINE FEED button
6/8 LPI button and indicator
RESET button
SELF TEST button and indicator

INTERFACE:

STANDARD: Differential line driver compatible with the HP 26099A interface board for 2100 and 1000 series computers, and the 30209A interface board for 3000 series computers.

OPTIONAL: HP-IB, HP's implementation of IEEE Standard 488-1975 and ANSI MC1.1.

OPTIONS: Sound cover. Language options include Arabic, Cyrillic, Katakana, and Draw; character options include APL, French, Ferman, Swedish/Finnish, Norwegian/Danish, Spanish, British, Japanese ASCII, Roman extension.

PRICE IN U.S.A.: $9250.

MANUFACTURING DIVISION: BOISE DIVISION
11311 Chinden Boulevard
Boise, Idaho 83707 U.S.A.

Optimizing the Performance of an Electromechanical Print Mechanism

by Everett M. Baily, William A. McIlvanie, Wallace T. Thrash, and Douglas B. Winterrowd

THE SPEED OF ANY MECHANICAL SYSTEM is usually limited by effects that result from controlling the acceleration and/or deceleration of critical parts. Finding the causes of certain effects and minimizing them so printing speed could be increased was a major design goal for the Model 2608A Line Printer.

Details of the print hammer mechanism in the 2608A are shown in Fig. 1. The longer beam (or tine) is the hammer and the shorter beam is the damper. To print a dot, the electromagnet is energized, pulling the hammer back against the core face. This stores enough energy in the hammer so when the electromagnet is de-energized, the hammer flys forward and strikes the paper through the ribbon with enough impact to leave a crisp dot. The hammer then rebounds and collides with the damper on the return stroke, reducing excessive residual motion of the hammer.

Between release and impact, the beam moves freely at a rate determined by its natural resonant frequency, which is about 400 Hz. The principal advantages of allowing the hammer beam to impact the platen at its natural frequency are speed and the relatively low power required—if the core is energized again to print an adjacent dot, the hammer will already be moving towards the core face.

The disadvantage is the tendency for the hammer to "overstrike", that is, to rebound from the damper and print an unwanted dot. In earlier implementations of this printing technique, this problem was handled by making the print speed slow enough to place overstrike dots roughly in coincidence with the printed dot.

One way to allow an increase in print speed would be to increase the resonant frequency of the hammer beam. The relationships involved are:

$\omega : h/l^2$, where ω = the resonant frequency;
h = hammer thickness;
l = hammer length.

$\sigma : hy/l^2$, where σ = stress;
y = displacement.

$E : \sigma^2 lh : y(h/l)^3$, where E = strain energy.

To raise the resonant frequency, either the thickness can be increased, or the length can be reduced. Because of a number of material-availability, processing, and tolerance problems, the hammer thickness is limited to a practical value. Since a certain amount of strain energy needs to be imparted to the hammer for good print quality, reducing the length of the hammer would increase the stress, raising it above the maximum permissible for good fatigue life. Thus, there are practical limits to raising the resonant frequency.

Another way would be to improve the damping mechanism so overstrike would be eliminated altogether. Specifically some means had to be found to limit residual motion in the hammer to something less than that needed for a visible overstrike.

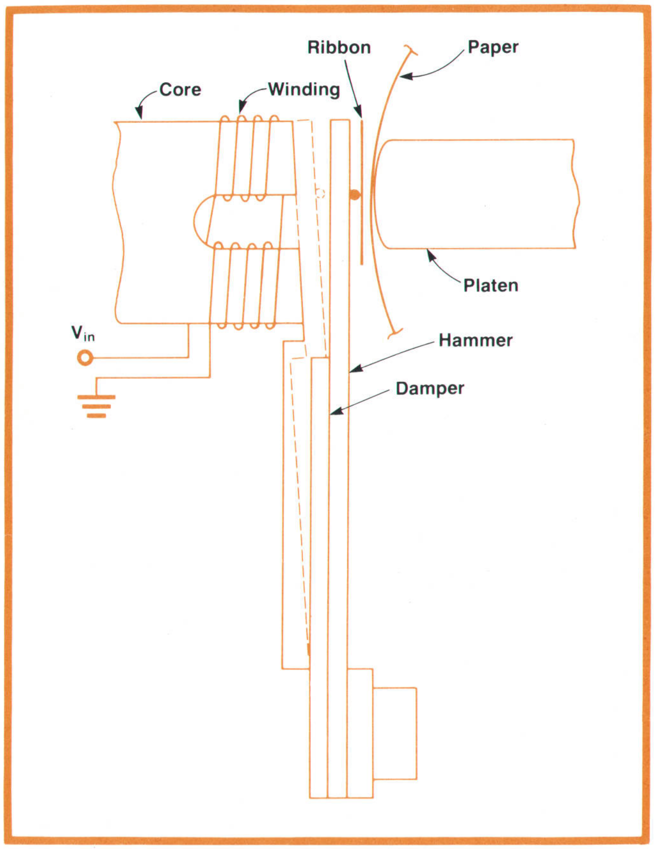


Fig. 1. *Details of the model 2608A's print mechanism. To print a dot, the electromagnetic core is pulsed, pulling the hammer back against it and then releasing it to fly forward, printing the dot.*

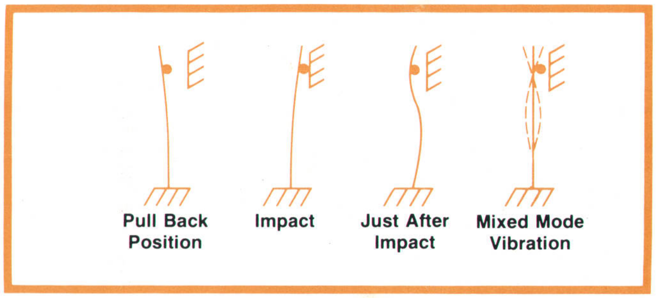


Fig. 2. *Basic hammer motion. Impact redistributes energy in the fundamental resonance mode to higher modes.*

Analyzing Overstrike

The main variables affecting residual motion are impact location, expressed as a percentage of hammer length, and the damper-to-hammer length ratio. Other variables are thickness and width but these had already been determined by overriding criteria.

Ordinary steel rules make useful large-scale models for observing the residual motion of a hammer. As shown in Fig. 2, the impact location, which acts as a temporary node point, is in the vicinity of the natural second node point for a cantilever beam. The action of impact redistributes the residual energy in the hammer beam into higher modes, reducing the amplitude of the fundamental motion. Experimentally it was found that the optimum impact location from the tip is 13% of the hammer length, 9% above the second node point. Furthermore, the model suggests that maximum damping occurs when the second-mode frequency of the hammer is a multiple of the dominant fundamental mode of the damper (not shown in Fig. 2). Experiments showed that the optimum multiple is 2. With this configuration, overstriking would be minimized over a practical range of parameter variations.

In going from the scaled-up thin-section model to an actual size thick-section model, however, performance was not the same. There was enough residual fundamental motion after impact to cause occasional overstriking. Sensitivity to parameter variations was much greater too. These performance differences might be explained by the rate at which energy in the hammer is dissipated, primarily in the form of acoustic energy coupled to the air. As the equations show, the strain energy increases as the cube of the thickness, but since energy dissipation into the air is proportional to frequency, which increases linearly with thickness, the damping ratio goes down in proportion to thickness. Intuitively speaking, the thicker hammer is stiffer and less affected by its surrounding medium.

It was therefore proposed, and confirmed by experiment, that the introduction of a magnetic damping pulse would obtain the results desired. The pulse is timed to occur as the hammer rebounds from the damper and is on its way towards an overstrike.

Further Analyses

The complexity of hammer motion led to further studies using a high-speed camera (40,000 frames/s). This provided excellent qualitative information, but quantitative data was hard to derive so the next step was to design a computer model of the hammer mechanism and simulate the dynamic processes digitally. This allowed us to simulate different combinations of operating conditions and parameters.

A finite element technique was used for the analysis. The hammer and damper beams were modeled as one-dimensional beam elements, i.e., only one degree of deflection and rotation were allowed at each node, and the deflection due to shear was neglected.

According to the simulation, the hammer imparts energy to the damper when they collide, but the collision propels the hammer back towards an overstrike. Experimentation with the location of the impact point, the ratio of the fundamental frequencies of the hammer and damper, and the gap separation between

Acoustic Design of the Model 2608A Line Printer

With as many as 132 hammers impacting the paper and platen during printing, it was inevitable that the Model 2608A's printing mechanism would generate some noise. In fact, prototypes demonstrated 90+ dBa sound-pressure levels. To make the printer a welcome participant in an office environment, a noise abatement program was initiated.

Because of the rigidity required in the printing mechanism, no attempt was made to apply any vibration isolation to the working parts. Instead, attention was directed towards attenuating acoustic radiation from openings in the printer's enclosure. Hence, the access cover was designed with a labyrinthine-foam seal around the edges. The stand is foam lined and totally enclosed, except for vents near the floor for cooling air input. The paper box is contained within this stand so noise leaving the machine by way of the paper inlet slot is attenuated by the stand. The cooling fan inlet is also in the stand and the cooling vents are at the rear, away from the operator.

An optional removable shroud encloses the paper outlet at the top of the machine and effectively moves this sound escape outlet to the rear of the machine.

To prevent the taut paper from acting as a soundboard where it leaves the platen, a plastic cover clipped on to the top of the core bar presses into the paper just above the platen (see the diagram in the box on page 26). This not only damps paper vibration, but it also reduces the area of paper affected by hammer action.

Although the resulting external sound levels are highly room and geometry dependent, they are in the range of 72 dBa. With the optional shroud, the sound level is 68 dBa, about the same as an electric typewriter and quieter than many nonprinting computer peripherals.

-Lynn Hessing

them confirmed that the combination of parameters found empirically was the optimum for suppressing overstrike.

The magnetic damping pulse was also implemented in the simulation. The simulation showed that the timing was critical; firing the pulse too late would provide insufficient damping, and firing it too soon would delay but not prevent overstrike. Simulation showed that for a given amount of magnetic core force, a pulse fired 3.3 ms after the print pulse and sustained for 0.5 to 0.7 ms would be most effective in suppressing overstrike. This was close to the optimum values found experimentally.

Driving Circuits

With the hammer design optimized, design of the hammer drive circuits could be completed. The design goal was efficiency with cost as a tempering factor. The number of drive circuits (132) dictated that each drive circuit have minimal complexity.

The approach chosen was to use a single high-power pulsed source and to use silicon-controlled rectifiers (SCRs) to switch the current to the appropriate hammer drive coils. The SCRs are driven at the TTL level by a shift register.

Pulsed drive was chosen so SCRs could be used. Since these are inherently latching devices, turn-on information needs to be present only at the beginning of the power-pulse cycle. This makes the remainder of the cycle available for loading the next dot information into the shift register.

The optimum drive voltage for the hammer coils is 18-20V for 1.5 ms. The coil current peaks at 1.5A, and since any number of coils up to 132 may be energized simultaneously, the total power-pulse current can be as high as 200A. The energy losses in generating, filtering, and switching 200A at 20V would have been prohibitively high using conventional techniques, not to mention the size of the necessary filter capacitors. The method chosen was to use a coarsely-regulated (±5%) 270V dc supply and an iron-core step-down transformer. Two power Darlington transistors switch the 270V to the transformer to generate the power pulses. The power switching thus takes place at a relatively low current level (<15A).

A block diagram of the core driving circuit is shown in Fig. 3. The dot control information is transferred from the microprocessor in serial form and held in an input register until the logic circuits shift it into the trigger register. As soon as the power pulse has been initiated and the indicated hammers fired, the dot information is shifted out of the trigger register into the damping register. At the same time, the previous dot information held in the damping register is shifted back into the trigger register. A delayed trigger is then applied to SCRs controlling those coils that were energized during the previous dot position but not the present one. This provides damping for those hammers now rebounding from the damper beams. The delayed trigger has no effect on SCRs that are fired in both the present and previous dot positions because they would be turned on already.

Mechanical Construction

The magnetic cores for the hammer drive are made of Hyperco-50 (vanadium permendur), a material that

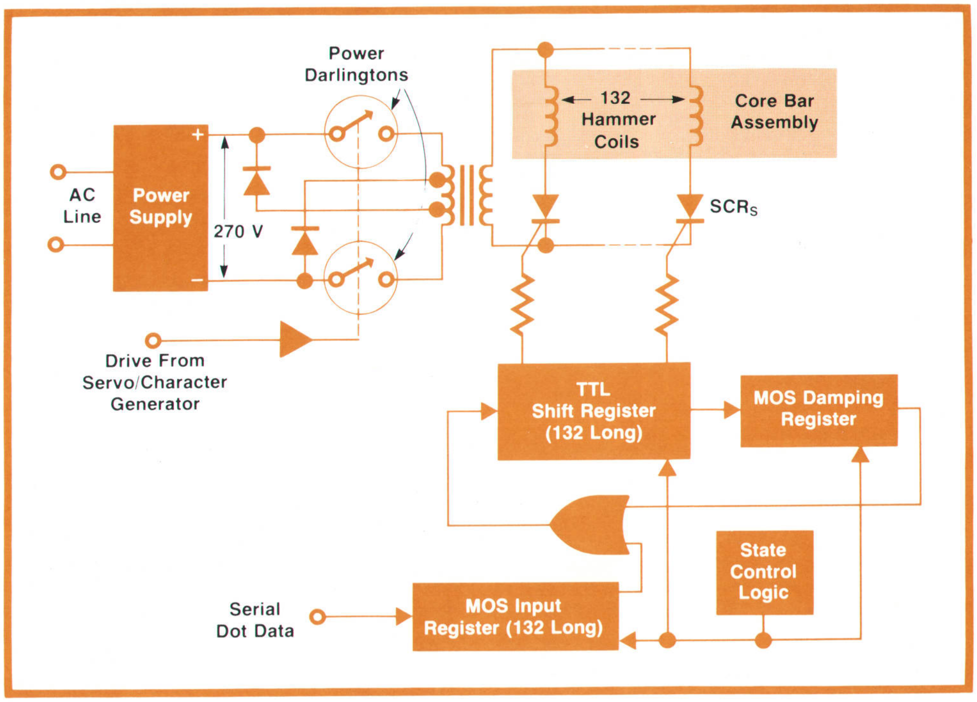


Fig. 3. *Block diagram of hammer drive circuits. A power pulse is applied to all hammer coils simulateously but only those coils that are switched on by the SCRs will be energized.*

provides one-third higher flux density than ingot iron. Since the same magnetic field intensity could be supported with less of this material than of iron, the weight of the hammer-core assembly was reduced by using this material. Because of its relatively high fatigue strength, this material is also well suited for the hammers.

The core assembly, or core bar as it is known, is subject to thermal expansion, since its temperature can vary as much as 150°C. Heat is generated by I^2R losses in the coils, and by eddy currents and hysteresis losses in the cores. Improved efficiency was built in by rigidly mounting the magnetic cores and the hammers in a single assembly that moves as a unit. The rigid mounting permits a minimum-reluctance air gap. Although this reduced the required drive energy from 0.07 watt-second per dot to 0.03 watt-second, the dots are now printed at a faster rate with a corresponding increase in heat.

Heat build-up within the core bar presented a problem since the core bar is asymmetric, does not heat uniformly, and is composed of such diverse materials as aluminum, steel, and potting compound. To combat this, the core bar is designed with eleven aluminum modules clamped tightly between two end caps by two tension rods, forming a rigid structure. Each module holds an assembly containing twelve cores and hammers, as shown in Fig. 4. This structure allows the various materials to expand at different thermal-expansion rates without causing out-of-tolerance warping of the assembled core bar.

Heat generated in the core bar is dissipated mainly by fins located on the back of the bar. Air is blown upwards through the fins and out the top of the printer through the paper exit, with a small amount going through the hammers. Calculations show that the fins can dissipate a maximum of 185W and an additional 60 to 90W can be dissipated by other surfaces, assuming a practical air flow of 0.3 to 1 m^3/minute and an allowable wire temperature of 200°C. This is adequate for removing the heat when printing alphanumerics. However, when printing dense dot patterns in the graphics mode, heat can be generated at rates up to 620W. For this reason, the number of dots printed is monitored electronically and print delays are imposed if the dot density averaged over a

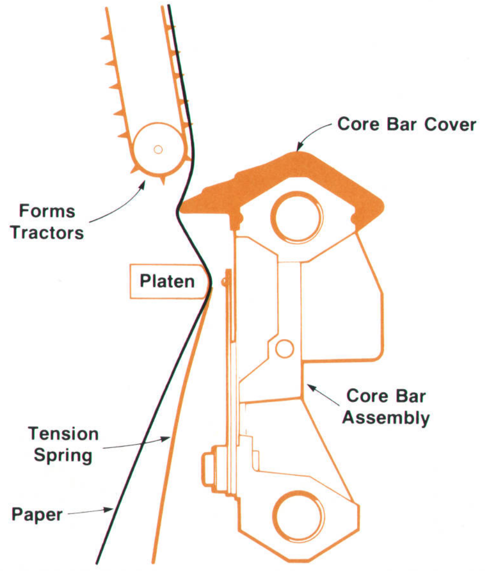
Precise Paper Movement

Good print quality in a dot-matrix printer requires accurate placement of the dots for each character. As described in the accompanying article, servo control of the core bar motion in the Model 2608A Line Printer positions the dots accurately in the horizontal direction. Precise movement of the paper is required to position the dots accurately in the vertical direction. Not only must the paper be moved precisely but the movement must be made and stabilized within 5 ms if print-speed requirements are to be met.

It was found early in the project that step motor drive through a belt-and-pulley arrangement introduced too much inertia into the system to achieve smooth acceleration and deceleration of the paper. Therefore, a custom-made, four-phase step motor that advances 2° per step is used so it can be directly coupled to the tractor drive shaft.

Open-loop control of the step motor achieves the simplest and least expensive method of control. However, to assure that the rotor detent position is reached reliably for worst case loads, more than sufficient torque for overcoming inertia and friction is needed. To improve the motor's rated torque, a two-level voltage drive is used. A 600-μs overdrive pulse from a high-voltage supply causes an initial rapid increase in winding current (to 5A) that is sustained by a low-voltage supply for the remainder of the drive pulse. By this means, torque was increased 25% over a single-level drive while mechanical resonance problems were minimized.

Extensive studies with an electro-optical tracking system on early versions of the printer revealed that hammer action caused the paper to vibrate against the platen. To reduce this motion, the platen face was curved to match the bend radius of the paper, providing better adhesion as the paper is tensioned around it. Tension is provided by a thin (0.010-inch) stainless-steel, cantilever, paper-tension spring that contacts the paper along the entire width of the platen just below the print line. A plastic cover along the top of the core bar protrudes into the paper path to force the paper to conform to the radius of the platen face, as shown in the drawing.



With this paper path, the step motor drive moves the paper 0.0139 inch for each dot row and completes each movement within 5 ms without overshoot or other aberrations, giving excellent print quality.

-Robert Deely
-Lynn Hessing

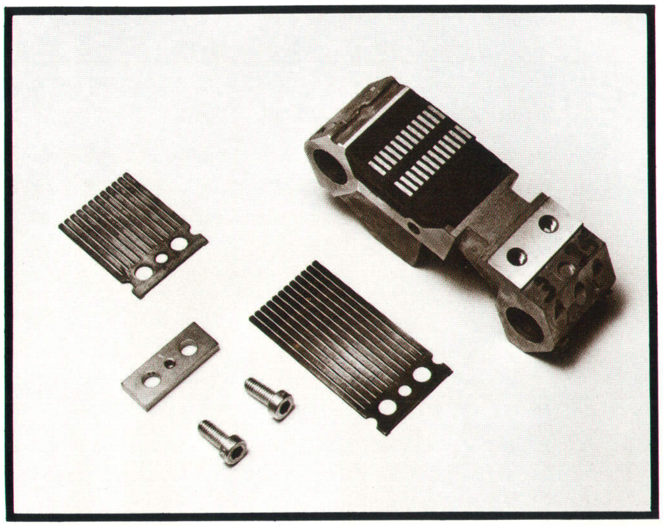


Fig. 4. *One core-bar segment holds twelve hammers, dampers, and cores. The cores are held in place by potting compound that permits the cores to expand thermally at a different rate than the aluminum module.*

page exceeds 37%.

Mechanical Tolerance

A tolerance study revealed that consistent print quality is obtained when hammer-to-paper spacing is 0.038 ±0.08 mm (0.015 ±0.003 inch). Hammer variations alone take up 0.08 mm of the allowed 0.16 mm spacing tolerance. This means that straightness tolerance over the 335-mm (13.2-inch) length of the core must be 0.08 mm, assuming variations are negligible. The core-bar assembly has an initial straightness tolerance of 0.05 mm, and this is maintained by the tautness in the tension rods.

Core Bar Motion

The core bar is attached to the printer casting by two stiff flexure springs that allow lateral motion while precisely maintaining the hammer-to-platen spacing (see Fig. 5). The rest position is aligned with the center dot column of the character matrix. With this configuration, energy stored in the flexure springs by core bar movement left or right aids the linear motor in providing the forces necessary to reverse the core bar direction at turnaround.

As illustrated in Fig. 6, the linear motor housing is also attached to the printer casting by stiff flexure springs. These isolate drive force vibrations from the casting. Although the flexures permit some movement of the linear motor, the closed-loop motion-control system assures precise and controlled movement of the core bar with respect to the platen, independent of motor housing motion.

The lateral motion of the core bar is tightly defined by the action of the hammers. Since these operate at a fixed repetition rate (400 Hz) close to the hammer resonant frequency, the core bar must move across the character spaces at constant velocity to assure equally spaced dots. In addition, core-bar turnaround must be completed in an integral number of dot cycle times. It is also desirable to make the velocity rate of change

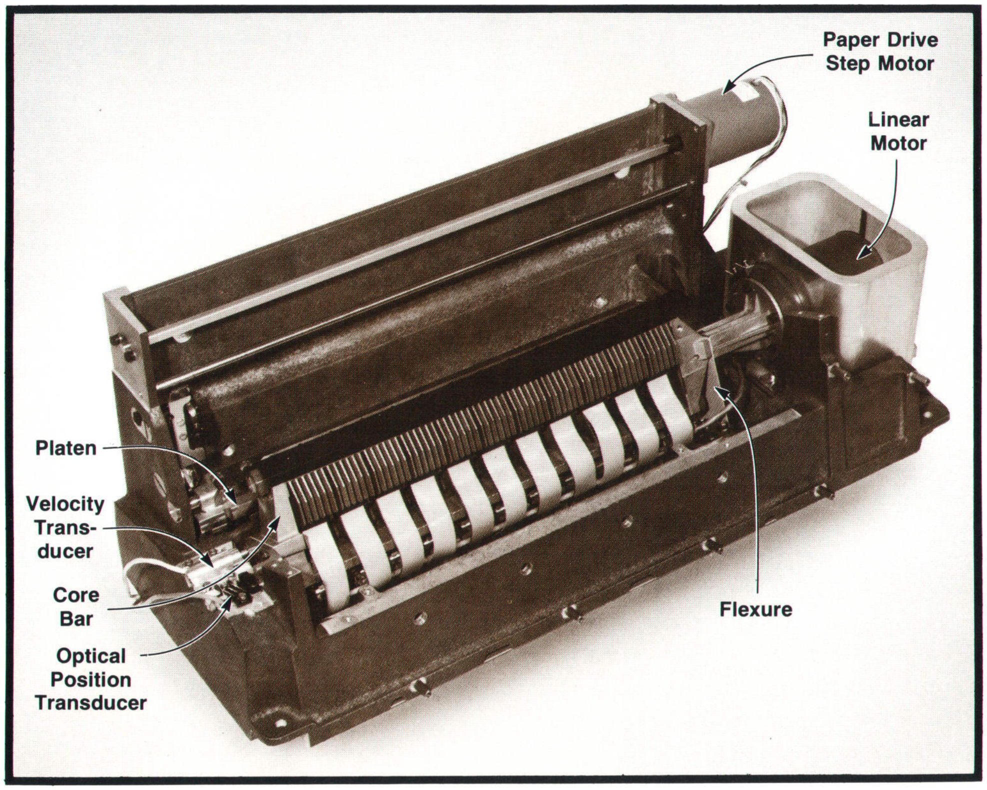


Fig. 5. *Partially assembled printer mechanism.*

constant during turnaround to minimize the magnitude of the forces acting on the bar. Hence, the desired core-bar velocity profile is as shown in Fig. 7.

To achieve the desired motion, the core bar is embedded in a closed-loop motion control system, diagrammed in Fig. 8. The actual velocity of the core bar is obtained from a velocity transducer that has its stationary part fixed to the print mechanism base and its movable part fixed to the core bar. The transducer output is compared to the desired velocity profile obtained from a waveform generator and the error between the two signals is forced to zero by the motion-control system loop. Thus, the actual velocity closely matches the velocity profile waveform.

The equation of core-bar motion is:

$$V_{in}\cdot K_a\cdot K_m = M_1\ddot{X}+C_1\dot{X}+K_1X$$

where X = core-bar displacement
V_{in} = drive amplifier input voltage
K_a = drive amplifier constant
K_m = linear motor force constant
M_1 = core bar mass
C_1 = effective viscous friction
K_1 = flexure constant.

With this as the model describing the core bar and drive components, a compensator that provided appropriate loop dynamics was designed using classical root-locus design. The compensator used a third-order pole at the origin to achieve a system capable of following the input velocity profile ramp on turnaround (classical type-2 control system) and three zeros were included in the left half plane for stability.

When this compensator was added to the system and the loop closed, however, the system proved to be unstable. It was found that the model did not give an accurate representation of the core bar over the total range of frequencies involved. Although the core bar oscillates at about 30 Hz during the printing of alphanumerics, its motion has a high harmonic content because of the nonsinusoidal nature of the velocity profile. At frequencies above 400 Hz, the core bar, which at that time was a ribbed steel bar with the cores and hammers mounted along the top, no longer behaved as a rigid body but was subject to some twisting and bending. The twisting and bending motions were coupled into the velocity transducer.

Frequency response plots of the core bar system were made with an HP Model 5451B Fourier Analyzer to determine what additional compensation would be needed to stabilize the loop. These plots disclosed the existence of several resonance peaks that rendered the system uncontrollable for the response speeds of interest. An accelerometer probe was then used with the Fourier analyzer to find the sources of the resonances. Therefore, changes were made to the core-bar design. The changes were iterative in nature with each change followed by further probing with the accelerometer and analyzer to determine the extent of the resonance attenuation achieved.

As a result of this process, several modifications were made. For example, the coupling structure between the linear motor coil form and the core bar was strengthened to prevent excessive flexing. The print mechanism base, originally made of half-inch aluminum members bolted together, was replaced by a one-piece, ribbed aluminum casting, and the core bar was redesigned with the cast-aluminum modules shown in Fig. 4 for increased rigidity. Finally, the velocity transducer was relocated to a point where any twisting or bending modes of the core bar would have minimum effect. The frequency response plots also indicated a need to improve the linearity and speed of response of the drive amplifier.

With these changes, plus the addition of lead-lag and roll-off compensators to the electronics, the desired stability was achieved.

Position Control

For proper synchronization between hammer operation and core-bar motion, it is necessary to generate

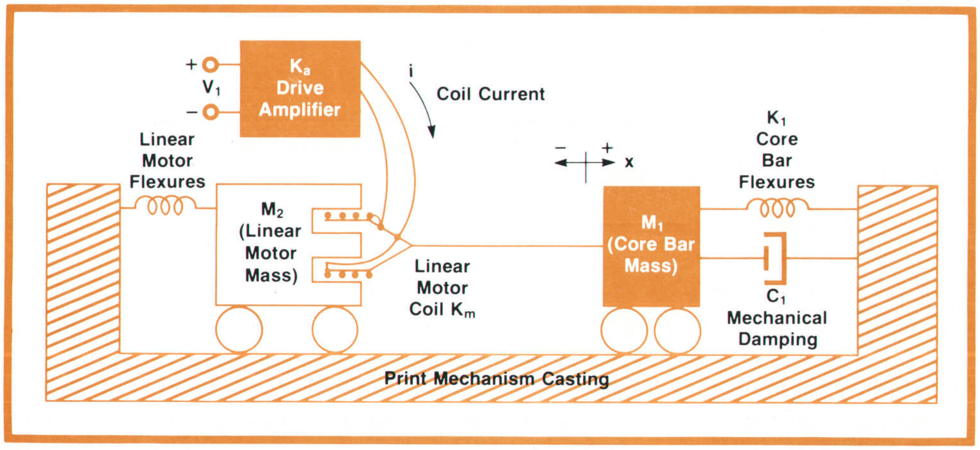


Fig. 6. *Mechanical diagram of the core bar and linear motor.*

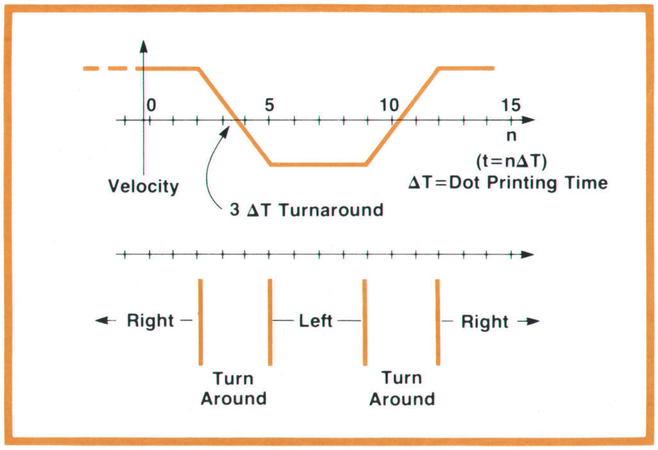


Fig. 7. *Velocity profile of the core-bar for obtaining even spacing of the printed dots and a constant rate of change during turnaround.*

information pertaining to the core bar position. This is done by applying the output of the velocity transducer to an op-amp integrator and applying the resulting position waveform to a bank of comparators that have equally spaced reference voltage levels. Each time the waveform crosses a comparator reference level, the resulting output edge triggers a pulse generator, signifying that a dot column has been encountered.

A second position circuit is included for controlling the stationary "home" position of the core bar when the printer is in the STANDBY mode. An LED and a phototransistor are mounted on the base casting with a vane attached to the core bar passing between them. The vane partially blocks the LED light when the core bar is in the home position so the output of the phototransistor is proportional to core bar position over a narrow range. This signal is switched into the motion-control feedback loop during STANDBY to place the core bar in the home position. An initial adjustment of the vane is made so the home position coincides with the mechanical rest position of the flexures.

The output of the position integrator is also included in the home-position feedback loop. The integrator provides an accurate, dynamic position-feedback signal for returning the core bar to the home position in situations where the core bar is remote from the home position with the photo-detector circuit in a limiting state. A block diagram of the composite motion and position control system is shown in Fig. 9.

When the core bar is operating in the oscillatory print mode, the output of the phototransistor is used to generate a short pulse each time the core bar crosses the home position. This pulse closes a switch in parallel with the position integrator capacitor to provide a momentary reset, preventing any slight offset in the velocity transducer circuit from causing the position waveform to drift with time.

Microprocessor Control

The printing of the dots is synchronized with the motions of the core bar and the paper by the machine's microprocessor. This is a control-oriented, eight-bit, NMOS microprocessor designed and manufactured by HP's Loveland Instrument Division. A second function of this microprocessor is to control the conversion of the ASCII-coded input characters into the dot patterns required for printing these characters.

The core-bar position control system provides position information to the microprocessor. Using this information, the microprocessor decides when to reverse or stop the motion of the core bar, when to advance the paper, which portion of the dot patterns to use, and when to fire the print hammers.

The core bar is moved according to the largest matrix to be used in a given line. In the alphanumeric mode, there are three sizes of dot matrices— 5×7, 5×9, and 7×9— according to what is being printed. For example:

E g _ (underline).
5×7 5×9 7×9

If upper-case ASCII only is being printed, seven dot rows with five dots per row in each character are printed at a speed of 400 lines per minute (lpm). If a lower-case character is included in the data stream, then nine dot rows per line are printed and the speed is 320 lpm. If underline is added, nine dot rows with seven dots per row in each character position are printed and the speed is then 250 lpm. In the graphics mode, core-bar motion across seven dot columns is used and printing proceeds at a rate of 40 dot rows (0.572 inches) per second.

The ASCII-coded characters sent to the printer by the external system are processed through the inter-

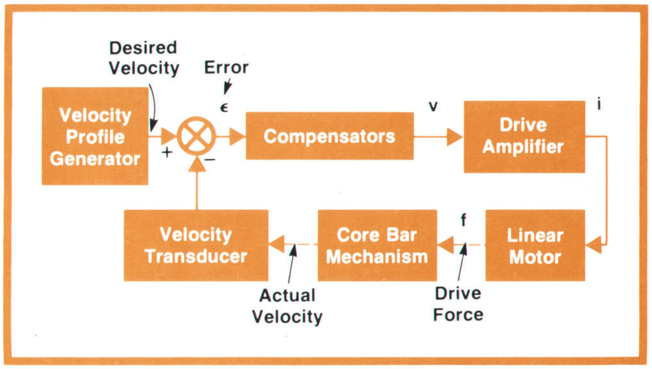


Fig. 8. *Closed-loop motion-control system forces the core bar velocity to match the velocity profile.*

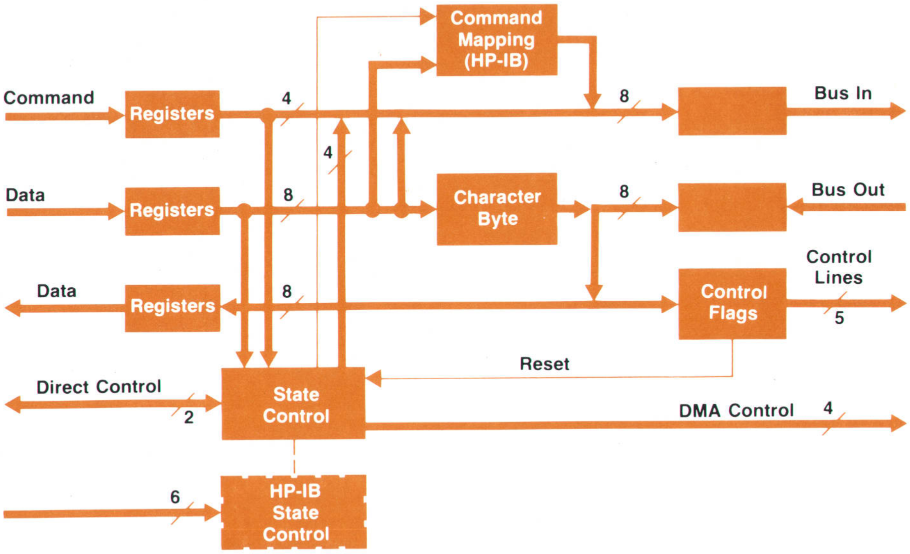
Interface I/O for a 400-lpm Printer

The design of the interface I/O for the Model 2608A Line Printer is in keeping with the general philosophy of flexibility in the printer's electronics. The selection of which functions are to be performed by the I/O and which by the main processor in the printer allows the communications flow between the printer and the host computer to be relatively independent of the main processor control.

The standard interface to the host computer is a TTL-level, direct-coupled, differential, parallel bus with eight line pairs carrying input data, eight carrying output data, four for commands, and one line pair in each direction for control. Except for the control lines, these connect to registers in the I/O, as shown in the diagram below.

Another function of the ASM is the processing of control code bytes that are to be acted upon by the printer. Character data bytes are scanned for the shift-out, shift-in, and backspace characters and these operations are performed, unless the printer is in the transparency (print control codes) mode.

The HP interface bus I/O is implemented with a CMOS SOS integrated circuit developed specifically to provide a logical interface to the interface bus defined by IEEE-488-1975. The HP-IB has bidirectional buses and control lines so information on these lines is sorted out by the HP-IB IC and applied to the appropriate unidirectional bus feeding into the standard parallel I/O, as shown in the diagram. The I/O's state control for the HP-IB has two serial linked ASMs.



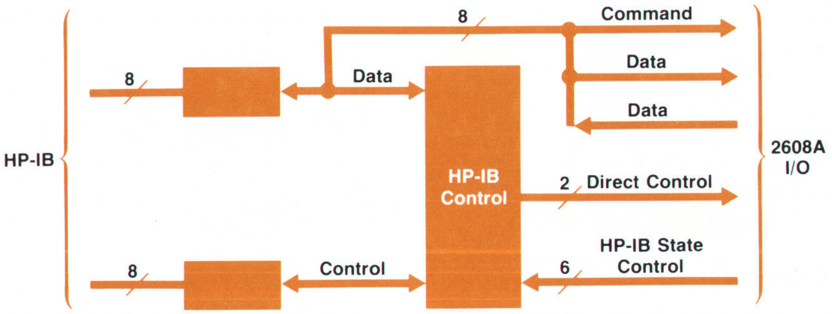
The heart of the I/O control is the algorithmic state machine (ASM) shown in the diagram. Strobe logic within the ASM syncs the asynchronous incoming information to the printer's clock. Incoming information is decoded to determine whether it is data or information related to a command. If it is character data, the ASM takes control of the bus into the main processor RAM by turning off the processor clock (stealing processor cycles) during the processor fetch phase. This gives the ASM direct access to the main processor's RAM by way of the processor's *outgoing* data bus. The ASM stores the character data at the appropriate address, returns control to the main processor, and then waits in its idle state for the next data or command byte. Input data burst rates up to 500K bytes/second can be handled this way.

Printer command bytes with I/O flags and data bytes associated with commands are routed onto the main processor's incoming bus.

The HP-IB I/O also enables three additional control codes to be acted upon by the printer. These are carriage return, line feed, and form feed.

The flexibility of the I/O system design allows adaption to other interface requirements. The standard interface is easily changed to a positive- or negative-true TTL interface. A serial interface is being designed.

--Stanley G. Peery



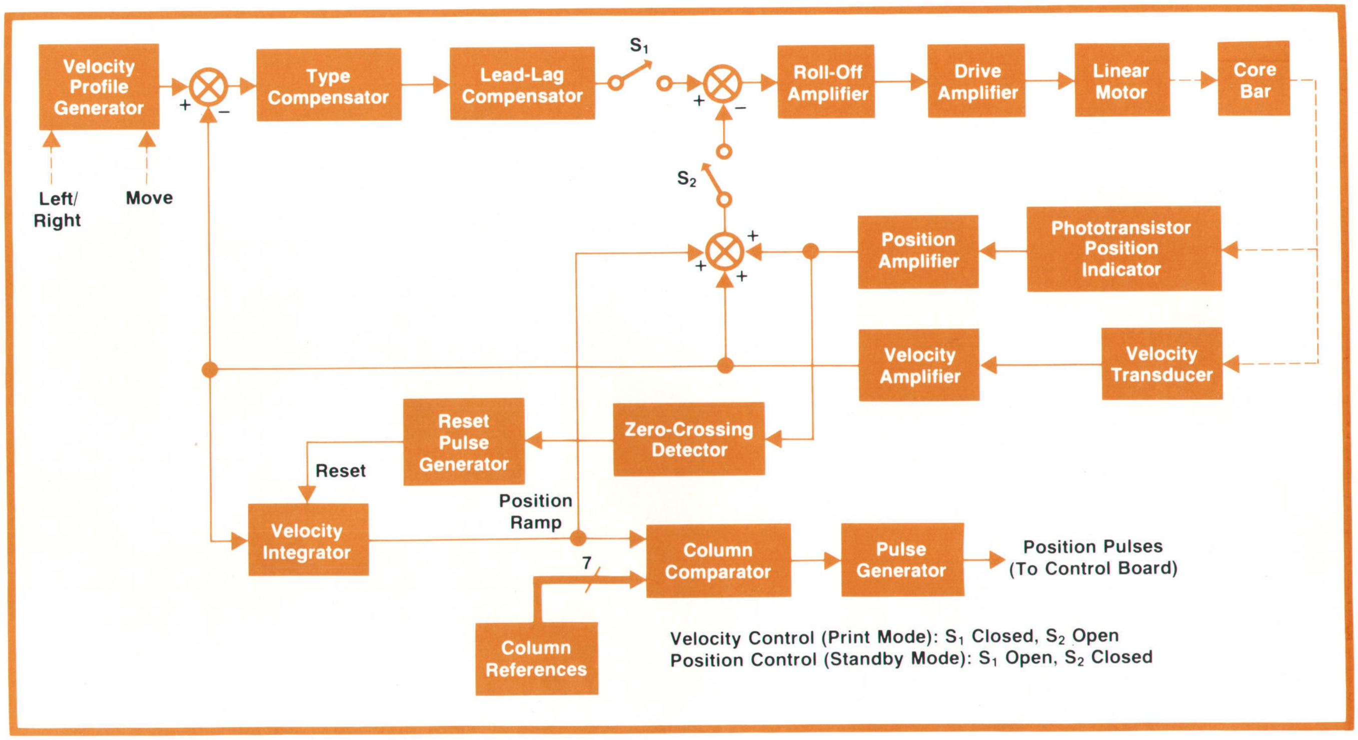


Fig. 9. *Combined motion and position system.*

face board and stored in a buffer RAM (read-write memory) using direct memory access (Fig. 10). Actually, there are two buffers so characters can be loaded into one while the microprocessor converts characters stored in the other into dot patterns.

Dot-pattern conversion is performed with the use of a table stored in ROM. Inputs to this conversion are:

1. 7-bit ASCII-coded character;
2. 1-bit code extension (i.e., shift in/shift out);
3. 4-bit language code (16 languages possible);
4. 3-bit dot-row number;
5. 3-bit dot-column number.

This implies a virtual memory space of 256K bits. However, because many characters are duplicated in the languages, the printer's actual memory space is 40K bits with 16 languages.

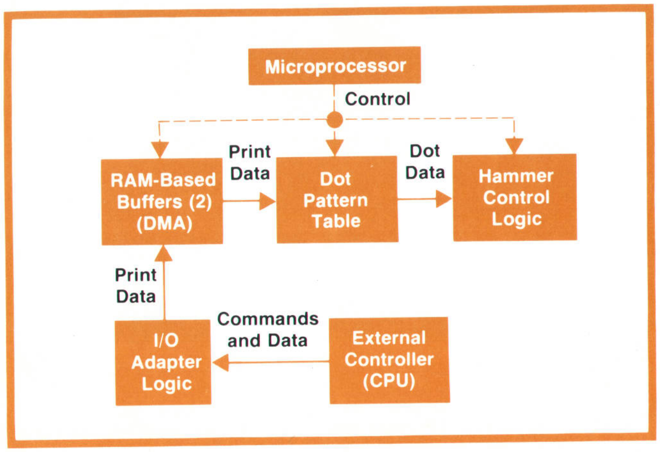


Fig. 10. *Block diagram of the microprocessor control system.*

The microprocessor moves the ASCII-coded character data directly from RAM to the dot pattern generation table. From there, it goes to the hammer control logic. This decoding is done once for every dot in each of the 132 character positions per line.

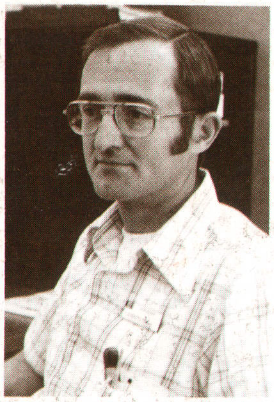
In the graphics mode, each eight-bit byte sent by the external system is presented to the hammer-control logic as eight contiguous dots with the 1's printed and the 0's left blank. Since there are 924 (7×132) dots per dot row, 116 (924÷8) data bytes are needed to describe one row. Thus, to turn the printer into a convenient, user-oriented graphics device, external software is required. Such software has been written for use with the Graphics-1000 software on the RTE-IV operating system for the HP 1000 series computer system.[1] The Model 2608A Printer can thus be an AGL-compatible plotter supported by the 1000 system.

Acknowledgments

The computer simulation of hammer motion was developed by C.S. Chan.

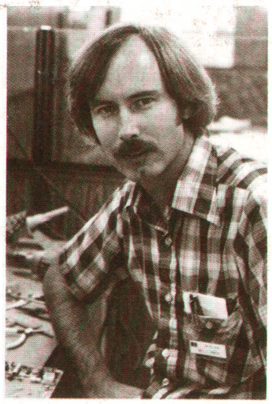
Reference

1. R.K. Juncker, "Higher-Performance HP 1000 Computer Systems, "Hewlett-Packard Journal, October 1978.



Everett M. Baily

Everett Baily received his BSEE degree in 1961 and MSEE degree in 1964 from the University of Idaho and PhDEE from Stanford University in 1968. An HP employee since 1974, Everett was production engineer for the 7970 Tape Drive before undertaking the design of the electronic drive circuitry for the 2608A Line Printer. Prior to joining HP, he was a professor of electrical engineering at the University of Idaho. Born in Twin Falls, Idaho, Everett now lives in Boise, Idaho, with his wife and daughter, 11 and son, 9. In off hours, Everett likes to hunt, fish, camp, take photographs and work at restoring his Model A Fords. He is also a United Way volunteer, is active in a local church group, and is vice chairman of the Boise IEEE chapter.



Wallace T. Thrash

Wally Thrash began work at HP's Boise Division as a senior lab technician, working in the investigation of the feasibility of the 2608A Printer, and later in design of the hammer-drive circuits and linear-motor drive amplifier for the 2608A. Before joining HP, he worked as a lab technician for a disc memory manufacturer. Born in Sacramento, California, Wally is completing his degree in applied physics at Boise State University. Married with two children (ages two and four), Wally lives in Boise, Idaho, and enjoys building his own microcomputer as well as camping, snow skiing and partridge and grouse hunting in the Idaho uplands.

Douglas B. Winterrowd

Doug Winterrowd received his BSEE degree in 1973 and MSEE degree in 1974, both from Montana State University. He joined HP shortly after graduation and has since worked on design of the main control electronics and firmware for the 2608A line printer. Born in Conrad, Montana, Doug is married and lives in Meridian, Idaho. Like many other Idahoans, Doug spends much of his leisure time hiking and crosscountry skiing.

William A. McIlvanie

Bill McIlvanie is a 1972 BSME and a 1974 MSME graduate of Washington State University. An HP employee since 1974, Bill was development engineer for the hammer mechanism on the 2608A Line Printer. He is also named inventor on a patent for a rock drill, a project he undertook while doing graduate work for WSU. Bill is single, lives in Boise, Idaho, and enjoys a variety of outside interests including skiing, fishing, golfing and playing on a Boise city league softball team. He is also fond of listening to classical music.

Hewlett-Packard Company, 1501 Page Mill
Road, Palo Alto, California 94304

Bulk Rate
U.S. Postage
Paid
Hewlett-Packard
Company

HEWLETT-PACKARD JOURNAL

NOVEMBER 1978 Volume 29 • Number 15

Technical Information from the Laboratories of
Hewlett-Packard Company

Hewlett-Packard Company, 1501 Page Mill Road
Palo Alto, California 94304 U.S.A.
Hewlett-Packard Central Mailing Department
Van Heuven Goedhartlaan 121
Amstelveen-1134 The Netherlands
Yokogawa-Hewlett-Packard Ltd., Suginami-Ku
Tokyo 168 Japan

Editorial Director • Howard L. Roberts
Managing Editor • Richard P. Dolan
Art Director, Photographer • Arvid A. Danielson
Illustrator • Susan E. Wright
Administrative Services, Typography • Anne S. LoPresti
European Production Manager • Dick Leeksma

0200032503&&&HARR&JA00
MR JULIAN A HARRIS
CHAYO ELECTRONICS LTD
COMMUNICATIONS DEPT
1575 NO 20TH AVE
PENSACOLA FL 32503

CHANGE OF ADDRESS: To change your address or delete your name from our mailing list please send us your old address label (it peels off). Send changes to Hewlett-Packard Journal, 1501 Page Mill Road, Palo Alto, California 94304 U.S.A. Allow 60 days.