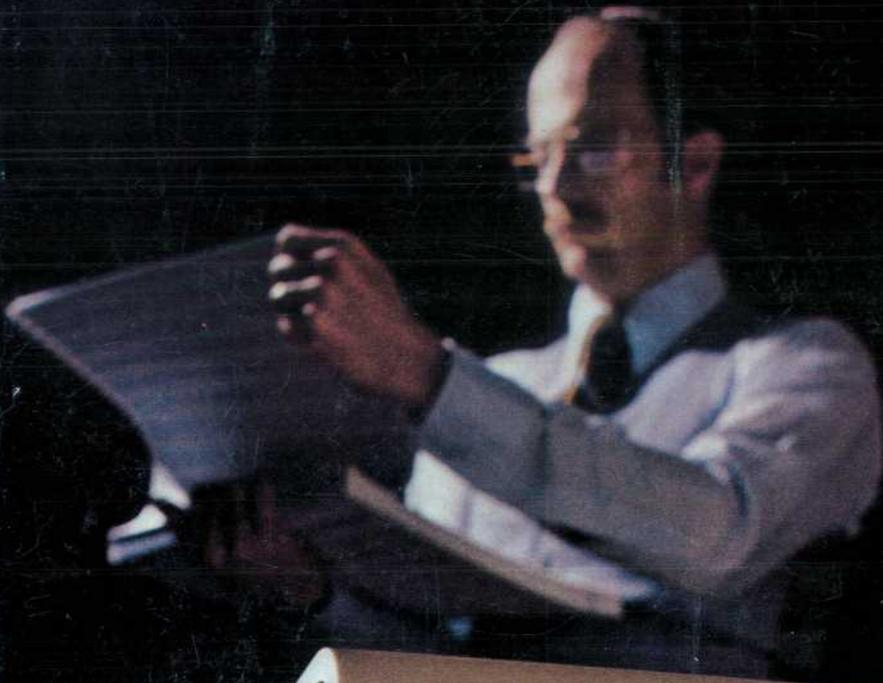


# HEWLETT-PACKARD JOURNAL

JANUARY 1978



# Versatile Low-Cost Graphics Terminal Is Designed for Ease of Use

*HP's newest computer CRT terminal combines sophisticated graphics and alphanumeric capabilities with easy-to-use, system independent, automatic plotting.*

by Peter D. Dickinson

**H**IGH-PERFORMANCE GRAPHICS capabilities are made available at relatively low cost by Hewlett-Packard's new system oriented, general-purpose, interactive graphics display terminal, Model 2648A Graphics Terminal (Fig. 1). Through its microprocessor-based architecture and raster scan technology, the 2648A Graphics Terminal provides a powerful combination of graphics and alphanumeric capabilities. By offering many off-line and system independent features, it helps take the burden off both the user and the host computer to make graphics applications more efficient and productive.

The primary purpose of a graphics terminal is to help the user process and display graphical information. Since the display is the user's primary interface to the product, the quality of the display is particularly important. The 2648A uses the same high-

resolution raster scan monitor that has been used in the entire 2640A family,<sup>1</sup> resulting in a bright, high-contrast, easy-to-read display. Other features made possible by the use of raster scan technology include area shading, selective erase, interface to external monitors, and matrix hardcopy compatibility.

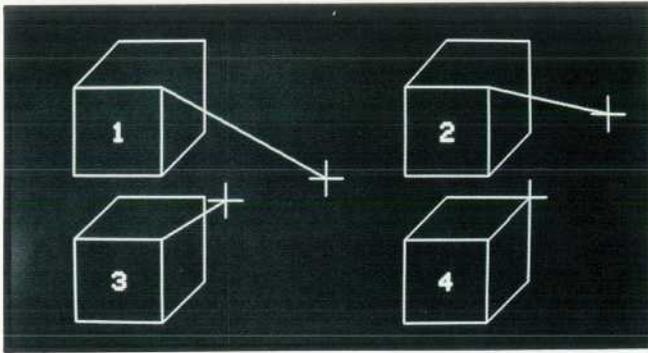
The 2648A's keyboard is the same as that of other members of the 2640 family except that the numeric keypad of other 2640 terminals is replaced by a graphics control group that controls the graphics cursor and display (see Fig. 2). Next to this group is the usual display control group that controls the alphanumeric cursor and display.

Pictures can be generated manually from the keyboard, read from optional cartridge tape units, or transmitted to the 2648A from the host computer. Information is communicated using ASCII charac-



**Fig. 1.** Model 2648A Graphics Terminal has both graphics and alphanumeric capabilities. Raster scan technology provides such features as area shading, selective erase, and compatibility with matrix printers. A comprehensive self test verifies operation and helps identify the defective module or component.



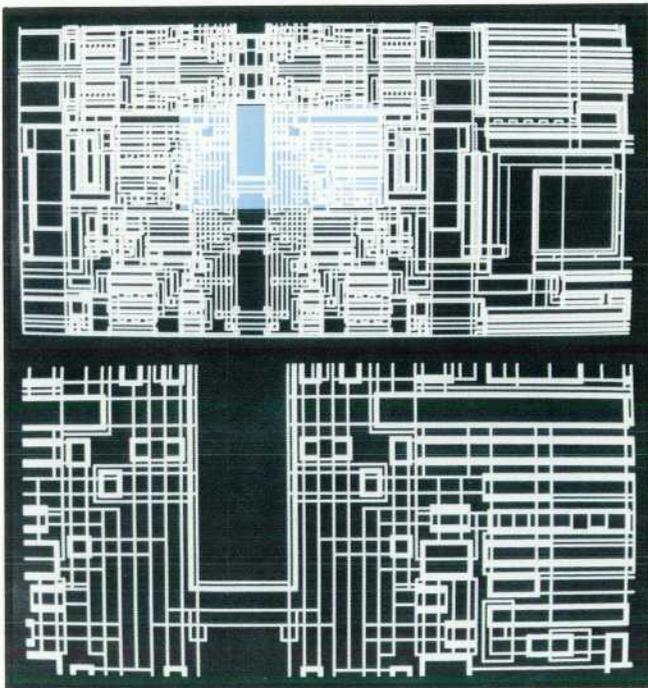


**Fig. 3.** A rubber band line helps the user draw pictures. One end of the line is fixed and the other moves with the graphics cursor (cross).

the graphics memory, so neither obscures the other. The two independent memories are also used effectively by a system independent feature called AUTO-PLOT.

### System Independent Graphics

One of the primary reasons computer graphics has not had wider application is that nearly all graphics applications differ, so each application requires special software specifically tailored to it. Furthermore, graphics has not been very popular among non-technical users because even simple plotting has frequently required the user to write computer programs or learn to use programs written by others. The AUTO-PLOT feature of the 2648A makes data plotting easy. It



**Fig. 4.** Zoom and pan features allow the user to magnify a portion of the image and scan the magnified window across the image. Here the lower display shows the indicated section of the upper display magnified  $\times 3$ .

requires no special software or programming knowledge, and is therefore system independent.

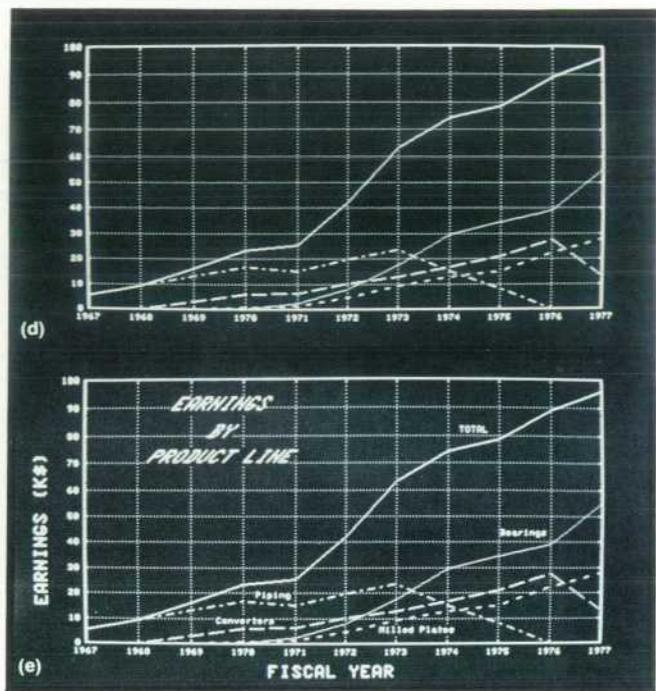
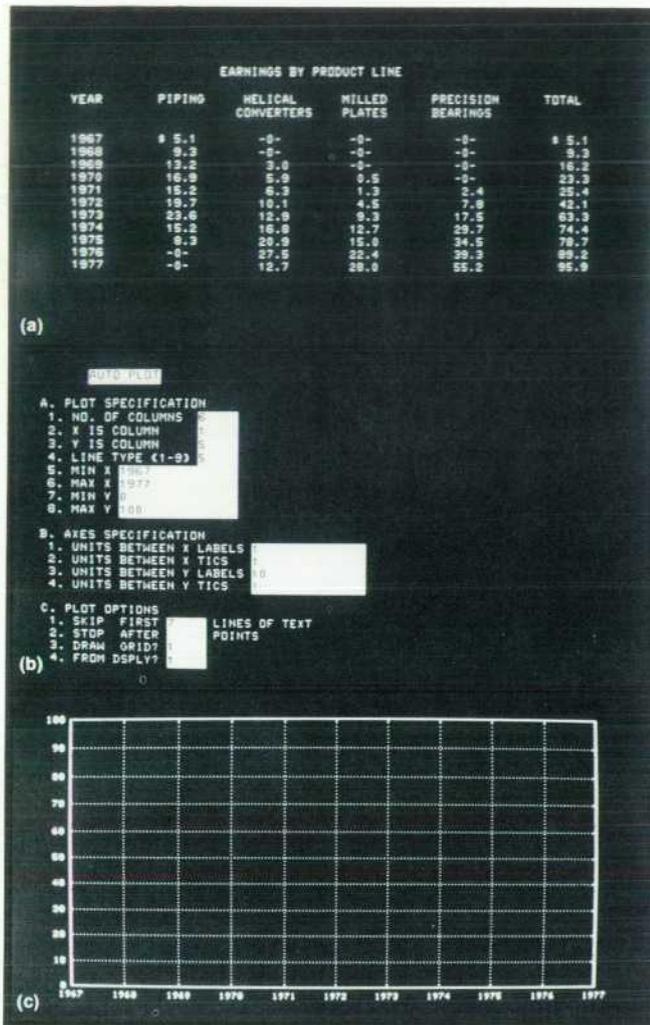
Data to be plotted may come from any source—for example, an existing application program, output from a BASIC or other high-level-language program, keyboard data entries, data read from cartridge tape, an inquiry to a data base, and so on. Fig. 5a shows some typical tabular data. To obtain a plot of such data, the user must first provide some information characterizing the data. This allows the 2648A to select appropriate columns, automatically scale the data, and label the axes. This information about the data is obtained from the user's response to questions presented by the "autoplot menu," which is permanently stored in the terminal's memory. Fig. 5b shows the menu with the appropriate responses for this example. Once the menu has been filled in, pressing the AXES key causes the axes to be drawn and labeled as shown in Fig. 5c. The AUTO PLOT key is then used to cause the data to be automatically scanned, scaled, and plotted as shown in Fig. 5d. The user can then use the graphics text features of the 2648A to title the plot, if desired. The finished plot is shown in Fig. 5e. The entire process takes only a few minutes and requires no special knowledge of computers or programming.

### Terminal Architecture

Fig. 6 shows a system block diagram of the 2648A, which is based on the proven 2640A family architecture.<sup>2,3,4</sup> Three plug-in boards contain the hardware and firmware required to implement the new features of the 2648A (see articles, pages 6 and 12). Unlike most other graphics terminals the 2648A uses raster scan technology, and many of its unique features are a direct result of the application of this technology to graphics.

Until recently, nearly all graphics terminals have used some form of directed beam technology, tracing out pictures on the face of a CRT in much the same way as one would with a pencil and paper. This approach yields good line quality but generally requires either a very high-speed (and expensive) vector generator, or an expensive storage tube, or both. If the display is refreshed, the amount of information that can be displayed without flicker is limited. If a storage tube is used, flicker is no longer a problem, but the entire screen must be erased to delete any part of the picture. Furthermore, storage tube displays are inherently less bright and wear out much faster than conventional CRTs. In spite of all their limitations, however, storage tube displays have been popular, because they were the only choice in low-cost graphics terminals.

A raster scan graphics terminal draws pictures in the same general sequence as a television set does, that is, the beam is swept in raster fashion from left to



**Fig. 5.** Autoplot makes data plotting easy and system independent. (a) Typical tabular data to be plotted. (b) The autoplot menu requests information from the user. (c) Axes are automatically drawn and labeled. (d) Data is automatically scanned, scaled, and plotted. (e) Completed plot, with titles provided by user, using the graphics text features of the 2648A.

right and top to bottom across the face of the CRT. The result is a dense matrix of potential points where dots can be placed to form images. To avoid flicker the process is repeated many times a second, usually at the ac line rate.

Until recently, raster scan graphics had been economically unattractive, since a bit of memory is required for each potential point on the display. However, high-density, low-cost semiconductor memories are now available, making raster scan graphics a practical reality. The 2648A uses sixteen 16,384-bit RAMs to store its 720-by-360-dot graphics image array. The hardware and firmware described in the following articles act as the user's interface to this image memory, providing powerful features to maximize its utility and capitalize on the many inherent advantages of raster scan graphics.

#### Acknowledgments

The development of the 2648A involved significant contributions by a large number of people. The author would particularly like to express his appreciation to Jim Elliott for his contributions to Autoplot and his

enthusiastic product management; Mike Child for his cooperative production engineering; Dwayne Murray



#### Peter D. Dickinson

Peter Dickinson is section manager for graphics at HP's Data Terminals Division. He was born in Morristown, New Jersey, received his BSEE and MEng degrees from Cornell University in 1969 and 1970, and joined HP in New Jersey in 1970, doing CRT terminal design. Moving to California, he designed ICs and served as project manager for three HP personal calculators. He's a member of the ACM Special Interest Group on Graphics and is named as an inventor on seven patents pertaining to calculators and CRT terminals. Peter is married, has a son, and lives in Monte Sereno, California. He plays guitar, works with stained glass, dabbles in real estate, and enjoys backpacking. He's also a private pilot and an advisor to an HP-sponsored Junior Achievement company.

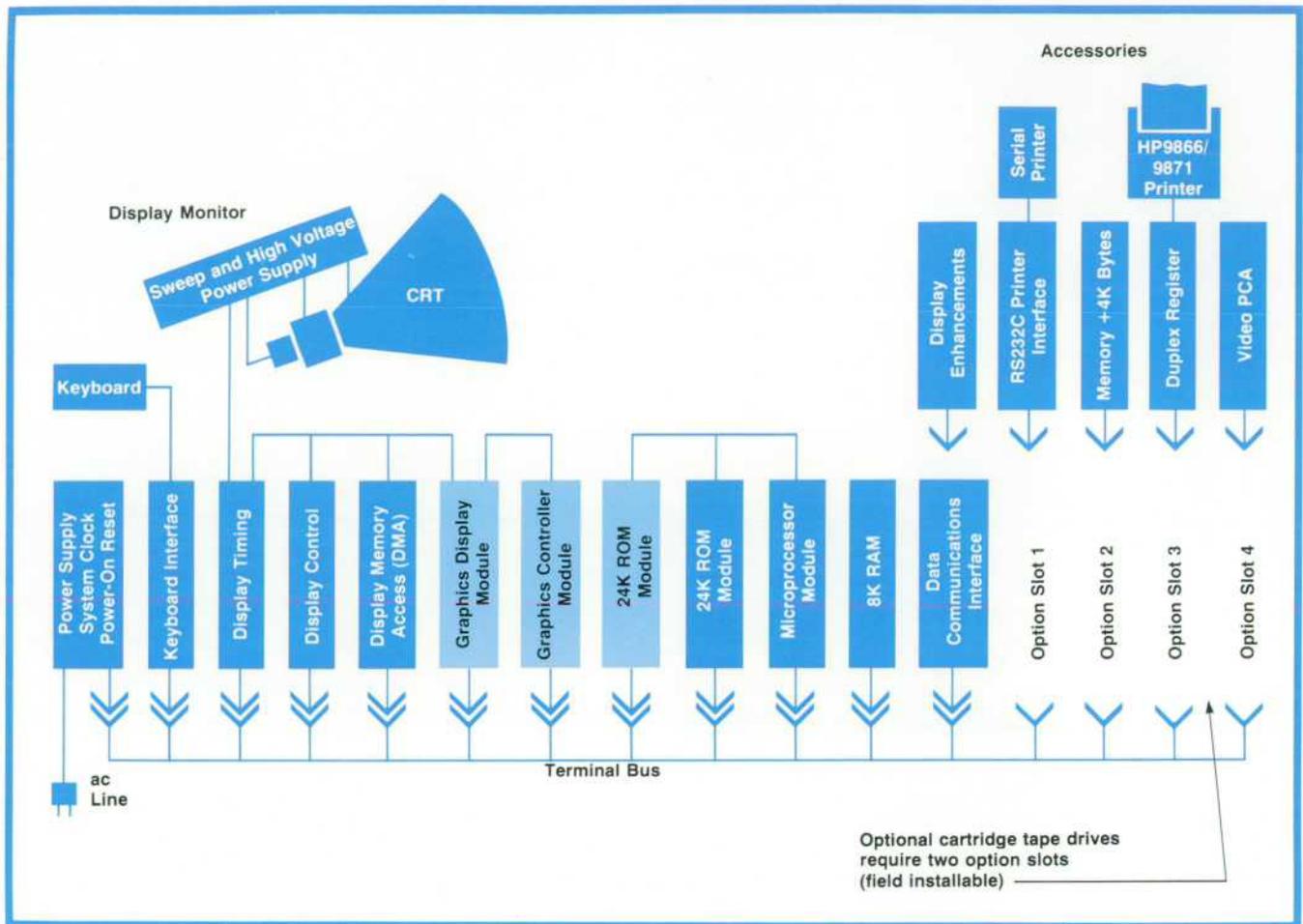


Fig. 6. 2648A architecture is based on the modular 2640 family architecture. Three new modules (shaded) contain the graphics firmware and hardware.

and Mike Caldwell for the user's and reference manuals respectively; Rich Ferguson and Steve Berman for demo and training materials; Dave Goodreau for promotional material; Jack Noonan for his early leadership of the project; and especially to Oty Blazek, John Moyer and Mike Raynham, whose contributions speak for themselves in the articles that follow, for their dedicated efforts to make the 2648A a timely success. 

### References

1. J. Roy, "A High-Resolution Raster Scan Display," Hewlett-Packard Journal, June 1975.
2. J.A. Doub, "Cost-Effective, Reliable CRT Terminal Is First of a Family," Hewlett-Packard Journal, June 1975.
3. A.B. Lane, "A Functionally Modular Logic System for a CRT Terminal," Hewlett-Packard Journal, June 1975.
4. R.G. Nordman, R.L. Smith, and L.A. Witkin, "New CRT Terminal Has Magnetic Tape Storage for Expanded Capability," Hewlett-Packard Journal, May 1976.

## Raster Scan Graphics with Zoom and Pan

by Otakar Blazek and Michael B. Raynham

**T**HE 2648A GRAPHICS TERMINAL displays black and white graphics on a 720-by-360-point raster array using one bit of memory for each point on the screen. This graphics image memory is part of the graphics display module (GDM) and is contained

in sixteen 16,384-bit random-access memory (RAM) chips.

Since the most common method for generating graphics is as a series of vectors, the 2648A has internal hardware vector generation. Vectors are drawn in

the image memory by the graphics controller module (GCM) by modifying the bits in the array that best approximate the desired line. The graphics hardware is also responsible for refreshing the dynamic RAMs and generating zoomed displays in real time without modifying the contents of the image memory. The GDM and GCM hardware needed to accomplish these tasks is on two plug-in printed circuit assemblies.

The 2648A expands on the established 2640 terminal family. The graphics hardware interfaces to the alphanumeric display and uses the same timing. It accepts the 21-MHz raster clock, 60-Hz frame rate, and 22.5-kHz horizontal line rate from the display monitor. The resulting output from the graphics hardware is a 21-MHz serial data stream sent to the display circuitry where it is combined with a similar bit stream from the independent alphanumeric hardware.

### Memory Organization

The graphics image memory contains one bit for every point on the 720-by-360-point display. If this memory were organized as a two-dimensional X,Y array, it would require  $2^{10}$  bits (X) by  $2^9$  bits (Y), or  $2^{19}$  bits to store the image. By assigning each image bit a number, it is possible to store the image as a one-dimensional linear list  $720 \times 360 = 259,200$  bits long (see Fig. 1). A memory size of  $2^{18}$  or 262,144 bits is then sufficient, reducing the memory requirement by half.

This linear list is organized as 16,200 16-bit words. Each of the sixteen 16K RAM chips contributes one bit to each word. Points adjacent on the screen are not necessarily adjacent in the memory. As Fig. 2 illustrates, there are eight possible memory displacements between adjacent screen points. Successive memory addresses correspond to screen dots along a horizontal line. Consequently, a complete scan line (720 dots) can be displayed by reading 45 contiguous words

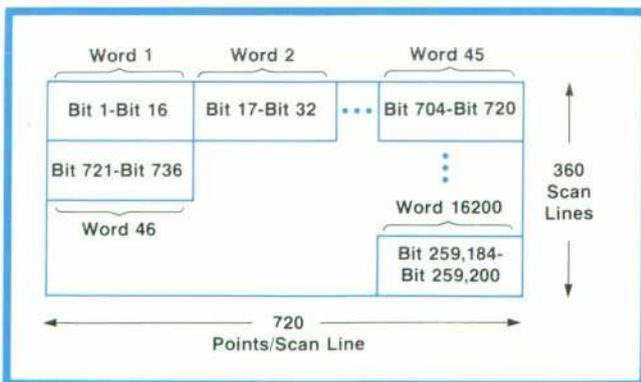


Fig. 1. The image memory contains one bit for every point on the display. It is organized as a linear list of 16,200 16-bit words.

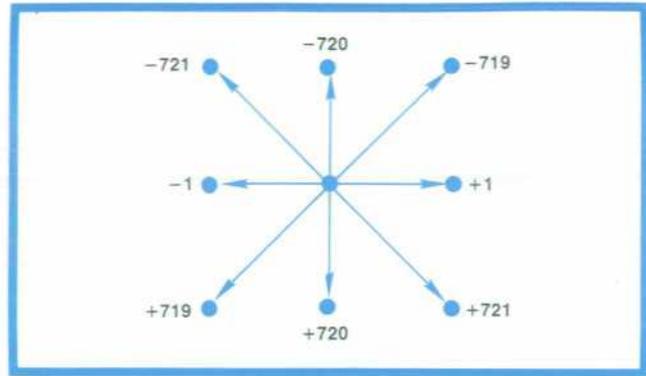


Fig. 2. Points adjacent on the screen are not necessarily adjacent in the image memory. There are eight possible bit displacements.

from the memory. A dot directly above another on the screen will be offset by 720 bits, or one scan line, in the memory. Note that moving upward on the screen corresponds to a negative displacement. Since the raster sweeps top to bottom, the raster origin is taken to be the upper left hand corner of the screen, with increasing Y pointing downward. Because the conventional graphics origin is the lower left hand corner of the screen, the graphics screen coordinates X, Y are converted to a memory bit address by the relation:

$$\text{Bit Address} = (359 - Y) \times 720 + X$$

The Y value is subtracted from 359 to compensate for the shifted origin.

### Display Refresh

The basic hardware functions of the 2648A are described by the flow chart in Fig. 3. When the power is first turned on, the microprocessor clears the cursor, zoom, and vector flags on the GCM. The GCM then waits for a new frame to start by looping on the vertical retrace signal sent by the display circuitry. Since the screen dots are stored in a linear array, displaying one horizontal line requires reading 45 words, each 16 bits wide, out of the image memory and converting them to a serial stream directed to the display monitor.

The GCM has two buffers, A and B, each containing sixteen 12-bit words. The B buffer can be loaded by the microprocessor via the 2648A terminal bus. When displaying a frame, the GCM maintains three variables in the A buffer: the read address, the word count, and the line count. The read address is an absolute word memory address, 14 bits long and stored in two locations, pointing to a word to be displayed. Since there are 16,200 words covering the whole screen, address zero points to the first 16 bits in the upper left corner and address 16,199 corresponds to the last 16 bits in the lower right corner of the

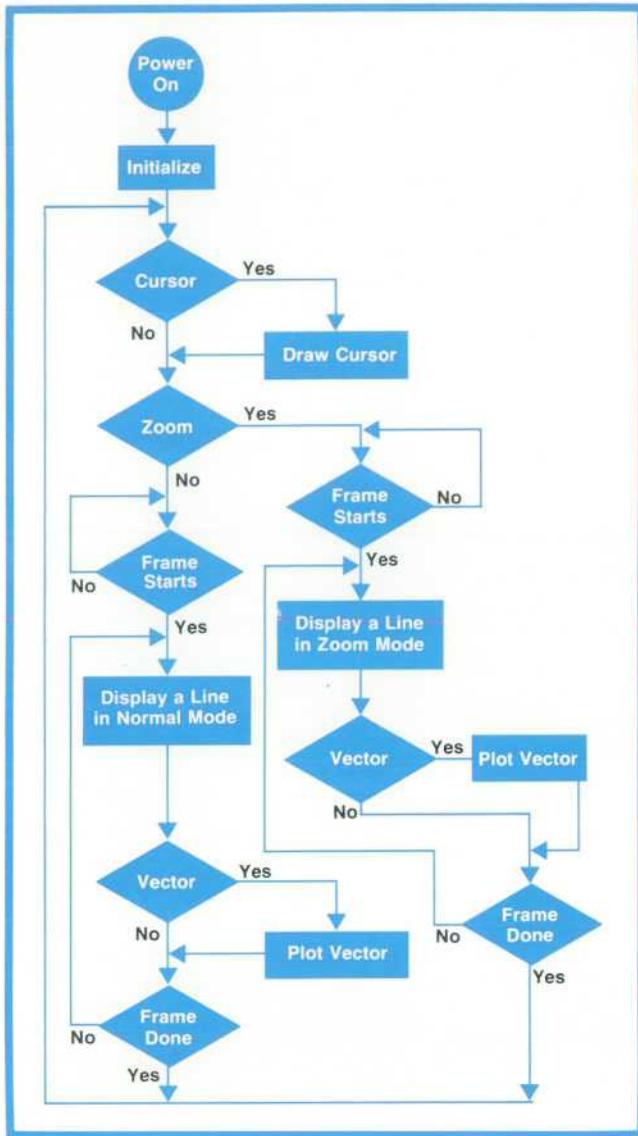


Fig. 3. Flow chart showing the basic graphics hardware functions of the 2648A.

screen. The word count counts the words displayed in one line. When a count of 45 is reached, the line is complete. Similarly, when the line count reaches 360, the frame is complete.

The process of displaying a frame then consists of the following:

- Step 1. While the GCM waits for the raster to begin a new frame, it initializes the read address RA, the word count WC, and the line count LC to zero.
- Step 2. Wait for a new line to start.
- Step 3. Read a word at RA and serialize it. Increment  $RA \leftarrow RA + 1$ .
- Step 4. Increment  $WC \leftarrow WC + 1$ . If  $WC = 45$  then proceed to step 5; otherwise go to step 3 and read another word.

- Step 5. Increment  $LC \leftarrow LC + 1$ . If  $LC = 360$  then the frame is finished. If  $LC < 360$ , set  $WC = 0$  and go to step 2.

### Zoom

The zoom feature displays image memory bits for a given magnification,  $M$ , in the form of  $(M-1) \times (M-1)$  dots, followed by one blank row and one blank column, as shown in Fig. 4. Repeating a dot horizontally on the screen is achieved by dividing the shifting frequency of the parallel-to-serial converter by  $M$ . Vertical repetition is achieved by reading the same line  $M-1$  times. In the zoom mode, only a portion of the image memory is read and displayed. Changing the zoom starting address causes the magnified portion of the image memory to pan across the display. Since only a portion of the image memory is being read, all memory rows must be refreshed during the blank horizontal line between magnified dots.

In the zoom mode the microprocessor outputs the zoom starting address ZASTR, the magnification  $M$ , and the word count per line  $K$  into the GCM's B buffer. The GCM maintains the zoom start address, the current zoom address pointing to the word being displayed, the line zoom address indicating the first displayable word of the current line, and the repeat count that keeps track of how many times a line has been displayed. The word and line counts keep track of words per line and lines per frame.

In the zoom mode a frame is displayed as follows:

- Step 1. While waiting for the raster to begin a new frame, the GCM sets the current zoom address ZA and the line zoom address ZAL to the zoom start address ( $ZA = ZAL = ZASTR$ ), and initializes the line count  $LC = 0$ .
- Step 2. Initialize the repeat count  $RC = 0$ .
- Step 3. Initialize the word count  $WC = 0$  and wait for raster to begin line.
- Step 4. Read a memory word at the current zoom address ZA. Increment  $ZA \leftarrow ZA + 1$  and  $WC \leftarrow WC + 1$ . Serialize the memory word.
- Step 5. Wait until the serial conversion is complete.
- Step 6. If the word count WC is less than K, the speci-

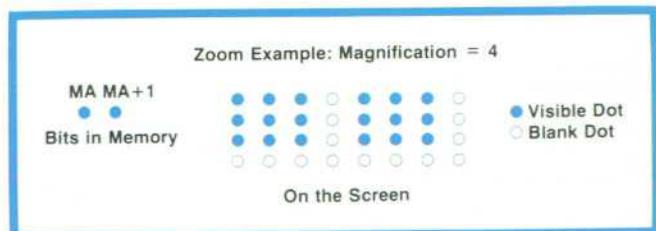


Fig. 4. In zoom mode each memory bit is displaced as a square of  $(M-1) \times (M-1)$  dots, where  $M$  is the magnification. Blank lines and columns separate the squares.

fied word count per line, then go to step 4 and read another word. If  $WC=K$  then proceed to step 7.

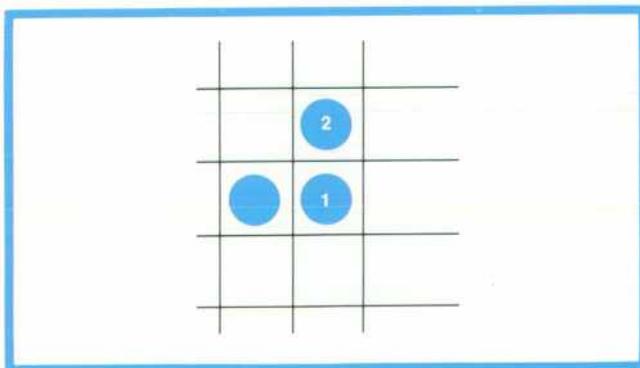
Step 7. Increment the line count  $LC \leftarrow LC+1$ . If  $LC=360$  then the frame is complete. If  $LC<360$ , proceed to step 8.

Step 8. Increment the repeat count  $RC \leftarrow RC+1$ . If  $RC<M-1$  (magnification-1), then set  $ZA \leftarrow ZAL$  and go to step 3 to repeat the line. If  $RC=M-1$  then draw one blank line, update  $ZAL \leftarrow ZAL+45$ , and set  $ZA \leftarrow ZAL$ . Then go to step 2.

### Vector Algorithm

Vectors are generated by computing the memory addresses of the points on the screen that most closely approximate the line between the specified endpoints. An iterative algorithm is used.<sup>1,2</sup> The memory address for a given point is computed by adding a memory displacement to the address of the previous point. For a vector in a given octant, there are only two possible displacements to choose from (see Fig. 5), and the sign of a discriminant determines which of the two to use at each point. After the initial values have been computed, the algorithm uses only addition and subtraction.

The initial values for the algorithm are computed by the microprocessor. These values include the initial starting point converted from X, Y coordinates to an 18-bit memory address, the two memory displacements, the initial discriminant value, two discriminant increments, and the number of dots to be drawn. These values are transferred to registers on the graphics controller module, which then executes the iterative algorithm (steps 2, 3, and 4).



**Fig. 5.** Vectors are generated by computing the memory addresses of the points on the screen that most closely approximate the line between the specified endpoints. At any given raster point, there are only two possible choices for the next raster point. If the slope of the vector is between  $0^\circ$  and  $45^\circ$ , for example, the two choices, as shown here, are 1) over one unit, a memory displacement of +1 bit, and 2) over one unit and up one unit, a memory displacement of +19. The sign of a discriminant determines which to use at each point.

The following description of the algorithm assumes a vector between the points (XSTART, YSTART) and (XFINISH, YFINISH) with absolute slope less than 45 degrees. For vectors of absolute slope greater than 45 degrees,  $\Delta X$  and  $\Delta Y$  are interchanged.

Step 1. Compute the initial parameters and transfer them to the graphics controller module:

$$\Delta X = XFINISH - XSTART$$

$$\Delta Y = YFINISH - YSTART$$

$$\text{Initial memory address } MA = 720 \times (359 - YSTART) + XSTART$$

Look up the memory displacements M1, M2 in a table using the octant determined by

## Designing with 16K RAMs

The 2648A is the first HP product to use the new industry standard 16K RAM chips.<sup>1</sup> The key characteristics of 16K RAMs that are important in this application include:

- high packing density, allowing the entire image memory and associated control circuitry to fit on a single plug-in printed circuit board;
- random access, for maximum vector drawing speed;
- low cost per bit because of wide industry use and multiple sourcing.

The most important design objective for the image memory subsystem was high reliability. Another important consideration was that the design be compatible with the minor differences in specifications among the many vendors of the 16K RAM.

Since the image memory printed circuit assembly contains high-frequency Schottky logic operating at 21 MHz in addition to the actual memory array, the first requirement was to isolate the two sections as much as possible. This was accomplished by using a memory output buffer having low input current and hysteresis to interface the memory array and display register logic.

The noise generated within the memory section was minimized by using a four-layer printed circuit board with internal power and ground planes. Both standard tantalum and distributed ceramic capacitors are used to provide local charge storage for the memory array. All memory input lines are series terminated, since unterminated lines result in overshoot that tends to increase the error rate and can be damaging to the memory chips.

The system was designed to use any 250-ns RAMs that could be qualified using HP's standard test techniques.<sup>2</sup> One limitation on the memory system design was that the total power dissipation had to be kept low for reliable operation at  $55^\circ\text{C}$  ambient temperature, as called for in HP class B environmental specifications. This is normally accomplished in memory system design by having the memory in low-power standby mode most of the time. This was not possible in the 2648A, because the memory is in read mode nearly all the time for the purpose of refreshing the display, so special care had to be taken to minimize the memory system power dissipation through the use of low-power logic components in all portions of the system where speed was not critical.

### References

1. J.E. Coe and W.C. Oldham, "Enter the 16,384-Bit RAM," *Electronics*, February 19, 1976.
2. R.J. Frankenburg, "All-Semiconductor Memory Selected for New Minicomputer Series," *Hewlett-Packard Journal*, October 1974.

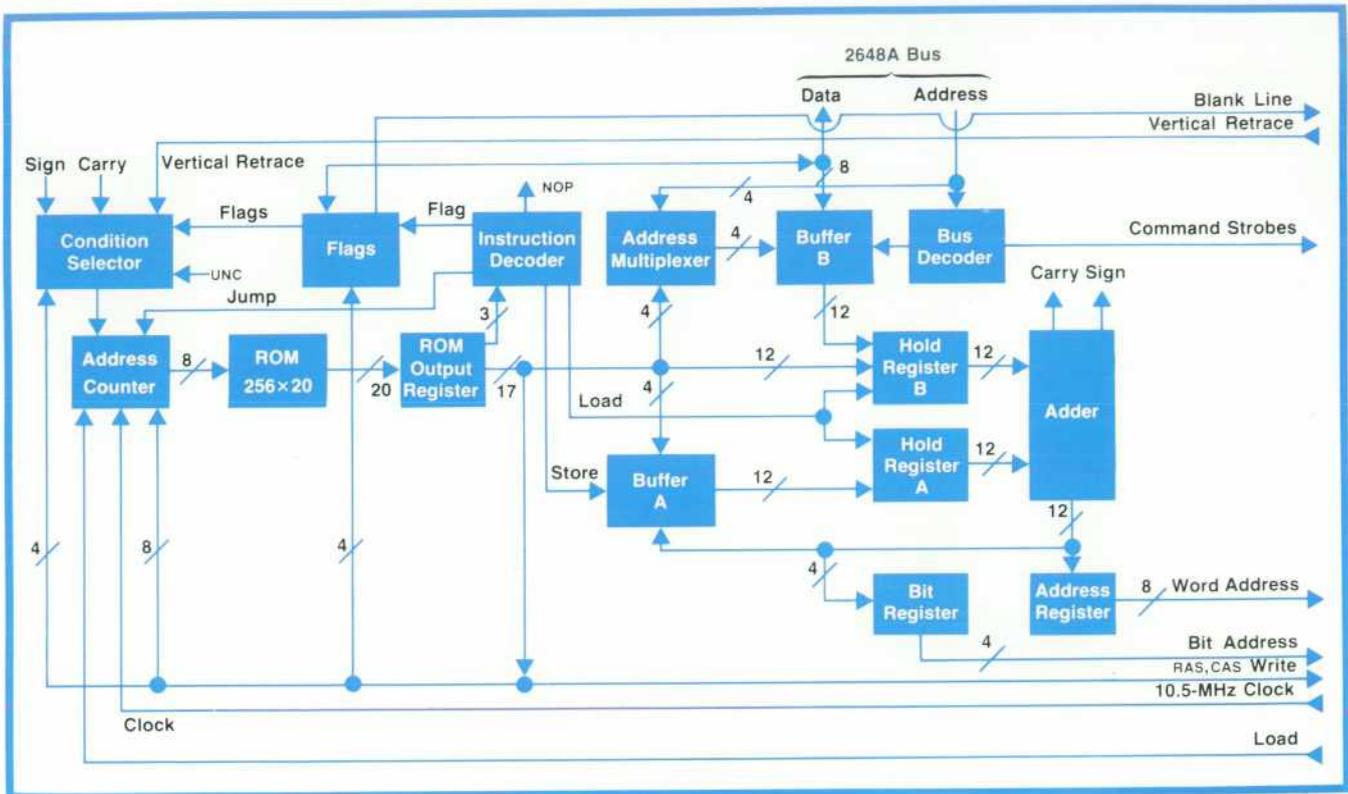


Fig. 6. The 2648A Graphics Terminal's graphics hardware consists of the graphics display module (GDM) and the graphics controller module (GCM), the latter shown here. The GCM is a microprogrammed machine that determines the contents of the image memory, a separate memory from the alphanumeric memory.

$\Delta X$  and  $\Delta Y$  as a key

Initial discriminant  $D = -|\Delta X| + 2|\Delta Y|$

Discriminant increment  $D1 = 2|\Delta Y|$

Discriminant increment  $D2 = 2|\Delta Y| - 2|\Delta X|$

Dot count  $DC = |\Delta X| + 1$

Step 2. Write the bit at memory address MA.

Step 3. Set  $DC = DC - 1$ . If the dot count is 0, then stop, the vector is finished.

Step 4. If the discriminant D is negative,  
Set  $D = D + D1$  (update the discriminant)  
Set  $MA = MA + M1$  (update the memory address)  
Go to step 2.

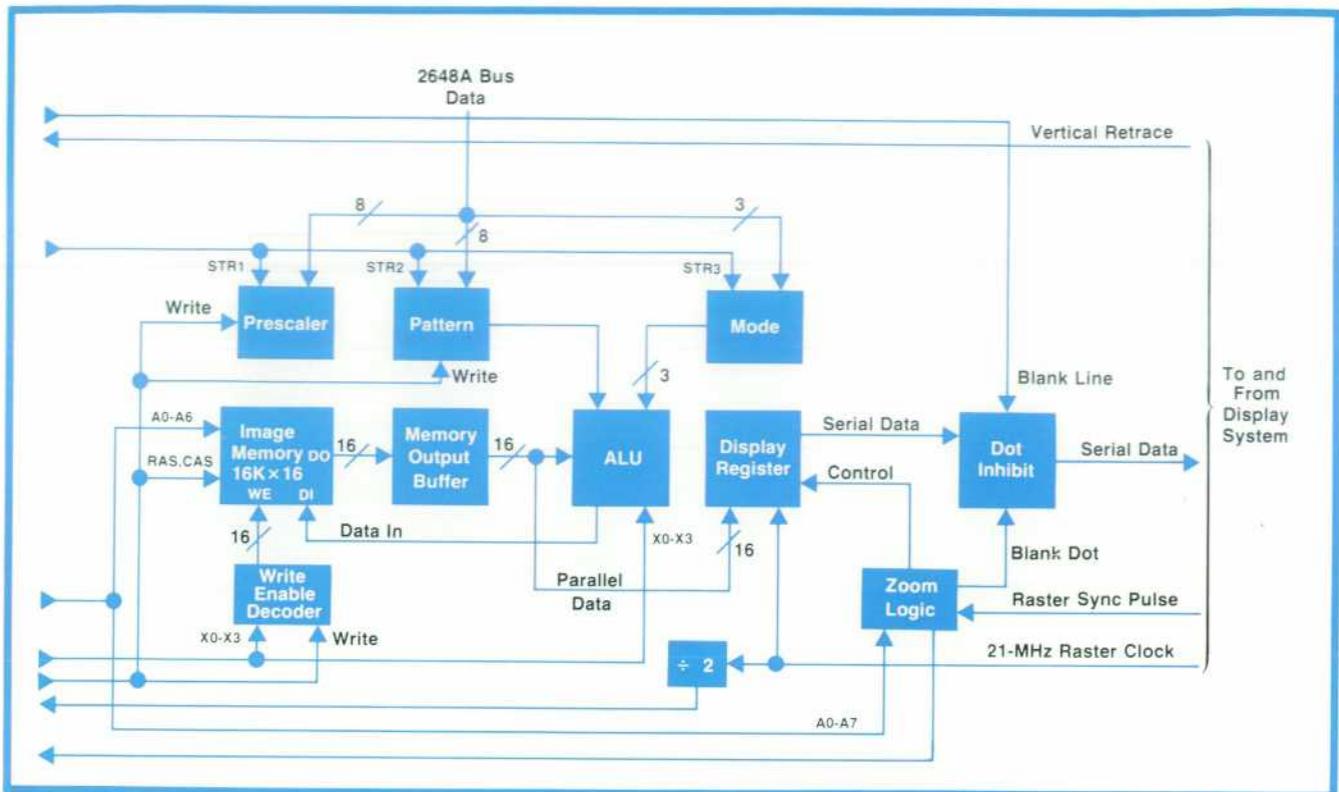
If the discriminant D is positive,  
Set  $D = D + D2$  (update the discriminant)  
Set  $MA = MA + M2$  (update the memory address)  
Go to step 2.

Communication between the microprocessor and the GCM is via a flag. When the flag is reset the microprocessor loads the B buffer and sets the flag, indicating that all the vector parameters have been specified. After the vector is completed the GCM clears the flag. Memory bits can be modified only when the beam is in horizontal retrace, which lasts ten microseconds, long enough for the graphics hardware to modify four dots.

### Graphics Hardware Organization

The graphics controller module (GCM) is designed as a microprogrammed machine (see Fig. 6). Its architecture includes eight instruction types and 256 words of control store, 20 bits wide. The instruction types include four load, one store, one flag, one conditional jump, and one NOP instruction. The load instructions load the B hold register with either the contents of a B buffer location or a ROM constant, and load the A hold register with an A buffer location. This allows adding an A buffer location and a B buffer location, or an A buffer location and a ROM constant. The store instruction returns the result of the addition back to the specified location in the A buffer, or it can optionally load it into the address and/or bit registers. The address and bit registers hold the image memory address during line display and vector generation.

The address counter, which is driven by a 10.5-MHz clock, addresses a word in the read-only memory. The control word read out of ROM is loaded into the ROM output register and decoded by the instruction decoder. To allow branching within the code a conditional jump is provided. The possible jump conditions, as determined by the condition selector, are unconditional jump, jump on carry, sign, vertical retrace, or jump on the state of one of six hardware flags. To save hardware, testing for zero is not done. Instead, the appropriate variables are loaded as nega-



**Fig. 7.** Graphics display module (GDM) contains the image memory, which consists of sixteen 16K random-access memory (RAM) integrated circuits. The 21-MHz serial data stream from the GDM goes to the terminal's display circuitry, where it is merged with a similar data stream from the alphanumeric hardware.

tive values in two's complement form, incremented, and tested for carry. Internal states in the program can be remembered using the six flags provided, and the states of these flags can later be used as branch conditions. A flag instruction sets or clears any or all of these flags. Some of these flags can be set and read by the microprocessor and are used for communication between the microprocessor and the GCM. The flag instruction can also halt the address counter until restarted by the load signal from the graphics display module (GDM). This serves to synchronize the GCM and the GDM. The bus decoder decodes strobed commands when the microprocessor loads registers on either the GCM or the GDM.

An instruction cycle takes two clocks, fetch and execute, except for a successful jump, which takes three clocks. The GCM works in pipelined fashion, that is, while one instruction is being executed, the next instruction is being fetched. Since a new instruction is loaded into the ROM output register each 10.5-MHz clock period (every 95 ns), the image memory timing is controlled directly from that register.

The graphics display module (see Fig. 7) contains the image memory, which is capable of storing  $720 \times 360$  dots. It also contains the logic for bit modification, the logic for shift and zoom, and the parallel-to-serial converter that outputs the serial bit stream to the display. The GDM accepts the memory address,

memory timing signals, and decoded commands from the GCM and provides the GCM with a 10.5-MHz clock by dividing the raster clock by two.

The memory array accepts the multiplexed row and column addresses and strobes from the GCM. The memory output is buffered to separate high-frequency logic from the memory array (see box, page 9). The 16-bit buffered words are loaded into the ALU (arithmetic/logic unit) and into the display register for parallel-to-serial conversion and transmission to the display. The write enable decoder generates write enable signals to one of the sixteen memory chips when vectors are being drawn. The appropriate chip is selected by signals X0-X3 sent by the GCM. The decoder is enabled by the write signal.

The ALU logic accepts 16-bit memory data, selects one data bit according to signals X0-X3, and modifies this bit as specified by the mode register. The result is sent to the DATA IN inputs on all 16 memory chips, but it is written only into the RAM chip that receives a write enable signal. A pattern memory stores eight-bit patterns for use in area shading and generation of special line types. The pattern memory can be stretched up to  $16 \times$  by the prescaler, which is a variable modulus counter. When enabled, it controls the shift rate of the pattern. The pattern memory and prescaler are loaded by the microprocessor via the 2648A terminal bus.

The mode register specifies the function that the ALU performs on the selected image memory bit: it can do nothing to the bit, clear the bit, set the bit, complement the bit, use a pattern bit, clear the bit if the pattern bit is a one, set the bit if the pattern bit is a one, or complement the bit if the pattern bit is a one.

The display register is a 21-MHz universal shift register that converts the memory data to a serial bit stream. Before this stream is passed to the monitor, it is fed to the dot inhibit logic where selected bits or an entire horizontal line can be blanked. The zoom logic loads and controls the display register. The load pulse sent to the GCM enables the address counter if it was previously halted by a flag instruction, establishing a handshake operation between the GCM and the GDM when an image memory word is read and displayed. In zoom mode the zoom logic controls dot blanking and determines how many times each bit is repeated according to the magnification. If the zoomed picture does not start on a word boundary, the word is pre-shifted in the display register under control of the zoom logic. The magnification and pre-shift counts are loaded from the microprocessor into the GCM's B buffer and are then transferred via the address register to the zoom logic.

#### Michael B. Raynham



Mike Raynham was born in Winnersh, England and received his HNC degree in electrical engineering from Luton College of Technology. He joined HP Ltd. in 1963, eventually moved to the U.S.A., and contributed to the hardware design of the 2116A and 2100A Computers and the 2644A and 2648A CRT Terminals. He's listed as an inventor on two patents and several patent applications related to those products. He received his MS degree in electrical engineering from the University of Santa Clara in 1971. Now a resident of Los Gatos, California, Mike hopes to move soon into a solar-powered house in the Santa Cruz mountains that he and his wife are designing and building. The Raynham's have two sons.

#### Acknowledgments

Thanks are due to those who helped with the design ideas: Dick Zimmerman and Paul Hughett for the initial concepts of raster scan graphics and Pete Showman for ideas on hardware vector generation. In addition, thanks to those who helped with the product: Bill Ulrey who redesigned the power supply to provide more power; Dave Kenyon who helped to get the graphics modules integrated into our computer-based test system DTS-70; Larry Bricker, production technician, who helped to debug prototypes and who contributed to a smooth pilot run; Pam Ewing, project coordinator, who helped with the project documentation. Special thanks to our PC people whose efforts contributed to the product's having been done on schedule. 

#### References

1. J.E. Bresenham, "Algorithm for Computer Control of a Digital Plotter," IBM Systems Journal, Vol. 4, No. 1 (1965) 25-30.
2. B.W. Jordan and R.C. Barrett, "A Scan Conversion Algorithm with Reduced Storage Requirements," Communications of the ACM, Vol. 16, No. 11 (Nov. 1973) 681-682.
3. W.M. Newman and R.F. Sproull, "Principles of Interactive Computer Graphics," McGraw-Hill Book Co., 1973.

#### Otakar Blazek



Oty Blazek was project leader for the 2648A. Born in Pilsen, Czechoslovakia, he received the equivalent of an MSEE degree in 1963 from the Technical University of Pilsen. After several years as a production engineer in Czechoslovakia, Germany, and the U.S.A., he earned another MSEE degree, from the University of California at Berkeley, in 1971. With HP since 1972, he's contributed to the design of the HP 3000 I/O system, designed the keyboard for the 2640 family, and designed 2648A graphics hardware. Several U.S. patent applications came out of the 2648A project. Oty is single and lives in Sunnyvale, California. He speaks five languages and enjoys tennis and skiing.

## Firmware Control of a Microprocessor-Based Graphics Terminal

by John J. Moyer

**T**HE 2648A is a microprocessor-based graphics terminal whose operation is completely con-

trolled by microcode stored in read-only memory (ROM). This firmware is based to a large extent on

code written for the 2645A Display Station. The 2645A was designed in a modular fashion to simplify extensions for future products. For example, important sections of the 2645A firmware were implemented using tables. These tables were merely expanded to include the new functions of the 2648A. Redefining the numeric keypad of the 2645A as graphics function keys required changes to only one discrete keyboard module. The firmware for the 2645A requires 22K bytes of ROM. Graphics extensions for the 2648A add 18K bytes.

The following paragraphs illustrate how the microprocessor was used in implementing several of the graphics features. In some cases, a task is partitioned between firmware and hardware, while in others the microprocessor interacts with the user to make the terminal easier to use.

### Vector Generation

The user causes the terminal to draw a vector by specifying a single endpoint. The terminal calculates the raster points that most closely approximate the straight line between the new endpoint and the previous endpoint. The microprocessor converts the endpoint from ASCII characters (such as 500,250) or a more efficient packed format (which reduces the endpoint 500,250 to the characters /4':) to binary. If either endpoint of the vector is off-screen, the coordinates of the portion of the vector that is on-screen are computed and substituted as new endpoints. The parameters required by the graphics controller module (GCM), described in the article on page 6, are then computed. Next the microprocessor tests a flag on the GCM to determine whether it has finished drawing the previous vector. When the GCM is idle, the microprocessor transfers the vector parameters and sets a flag that tells the GCM that a new vector is ready. The microprocessor can begin processing the next endpoint while the GCM is drawing the vector.

The microprocessor can set the mode in which a vector is drawn. The bits that make up a vector can be written by setting, clearing, or complementing the image memory. This gives selective erase capability,



**Fig. 1.** The 2648A Graphics Terminal uses an eight-bit pattern byte to specify dotted and dashed lines. A scale factor can be applied to stretch the pattern up to 16x.

as well as the ability to draw either white on a black background or black on a white background. To draw dotted and dashed vectors, the microprocessor can load and enable an eight-bit pattern memory on the graphics display module. Instead of drawing every dot in the vector, bits can be written or skipped over, according to the pattern (Fig. 1). The pattern can be stretched up to 16x by a prescaler.

### Cursor

The graphics cursor is drawn in the image memory as intersecting horizontal and vertical vectors. The microprocessor scans the graphics cursor keys to determine where the cursor should be drawn. The starting addresses for the two vectors are computed so the center of the cursor is in the specified position. If any part of the cursor would go off-screen, a shorter length for the appropriate vector is computed. The microprocessor then loads the GCM with the two addresses and two vector lengths, and sets a flag indicating that

## Graphics Self Test

Self test is an important feature in all HP terminals. It is designed to answer the basic question, "Does it work?" Self test also provides valuable diagnostic information. Since the complexity of the new graphics hardware is comparable to the entire digital portion of HP's first terminal, the 2640A, the addition of a comprehensive graphics self test was particularly important. The 2648A graphics self test consists of three tests.

### "Marching Vector" Memory Test

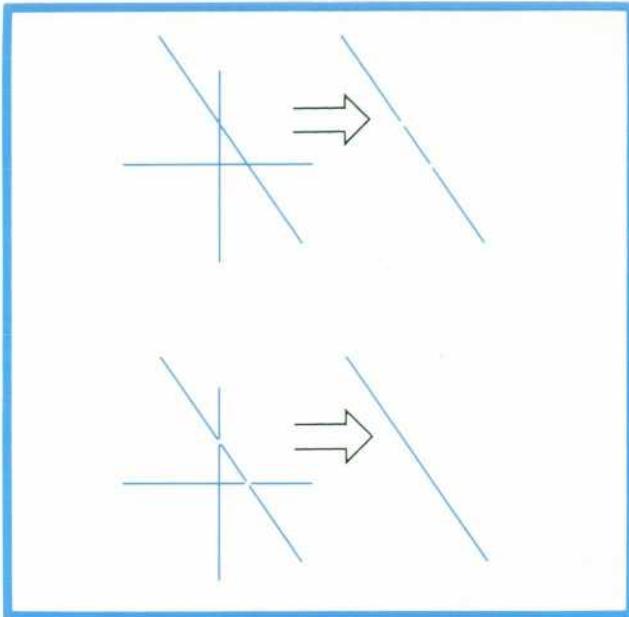
This is analogous to a "marching 1's" and "marching 0's" memory test. However, unlike a normal diagnostic, its operation can be viewed on the CRT in addition to being tested by the graphics controller. It verifies that the graphics vector generator operates in all four quadrants and that the graphics memory contains 259,200 uniquely addressable bits that can be set to a 1 or a 0. If any memory errors are discovered, they are reported on the display. The location of the failed memory pack is also indicated, allowing simple replacement of the socket-mounted part.

### Display Test

This is a visual test only, since the processor does not have access to the 21 MHz video bit stream. First ZOOM is tested by displaying a succession of numbers corresponding to different zoom factors. The numbers are written in sizes that are complementary to the zoom factors used, and consequently should all appear the same size on the screen. PAN is then tested by moving a checkerboard pattern across the center of the screen past the crosshair graphics cursor. If the pattern traverses the screen smoothly, then virtually all the zoom and pan hardware must be operating correctly.

### Alphanumeric Self Test

Since the 2648A features are a superset of the 2645A Display Terminal's features, the remainder of the 2648A self test is identical to the 2645A self test. Firmware-implemented graphics features are also tested here, since part of the test checks all of the ROMs to ensure that their stored bit patterns are intact.



**Fig. 2.** The 2648A's graphics cursor is drawn by complementing bits in the image memory and is erased by recomplementing the bits (bottom drawing). This avoids the problem shown in the top drawing, where erasing the cursor leaves a gap in any line that intersects it. However, drawing the cursor by complementing, as shown in the bottom drawing, leaves a gap at an intersection with another line. To remedy this the cursor is recomplemented every frame.

a cursor is to be generated. The GCM draws the cursor during vertical retrace, while the display is blanked.

When the cursor is moved, the cursor at the old position is erased before a cursor is drawn at the new position. As Fig. 2 illustrates, if a line is erased by clearing bits in the image memory, gaps will be left in any line it intersects. If the cursor were erased this way, large parts of the display would be erased as the cursor moved across the screen. Consequently, the cursor is drawn by complementing bits in the image

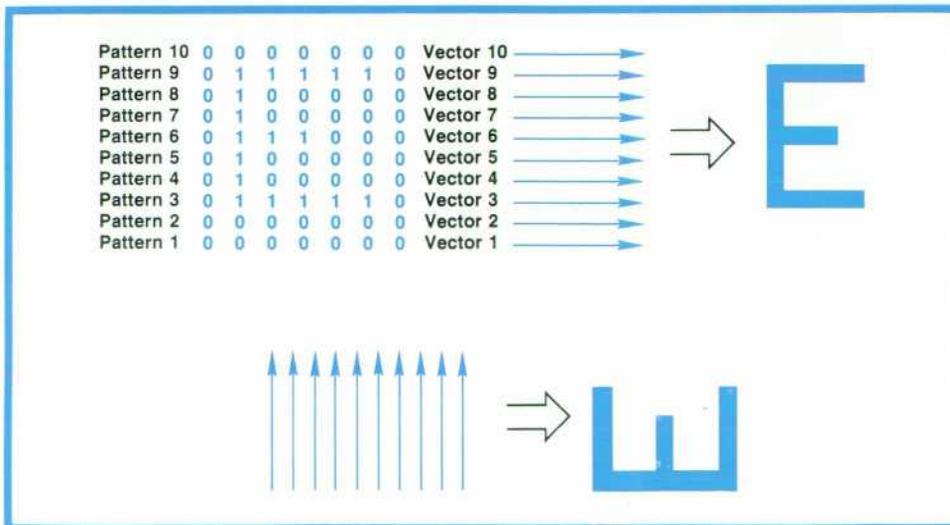
memory. To erase it, the identical bits are complemented again. Complementing a bit twice restores it to its original state. Complementing also insures that the cursor will always be visible, regardless of the background. However, as seen in Fig. 2, gaps will appear in vectors intersecting the cursor when the cursor is drawn. To remedy this, the cursor is recomplemented every frame. The resulting cursor appears half-bright because it is only visible every other frame, but it does not cause gaps when placed on top of other vectors.

### Zoom

Zoom allows the user to select a subset of the image memory and magnify it to fill the entire display. The center of the area to be zoomed is selected with the graphics cursor. The microprocessor uses the cursor coordinates and the desired magnification to determine the memory address of the first bit that will be displayed in the upper-left hand corner of the zoomed area. If this address is not on an image memory word boundary, the number of bits in the first word read that are not to be displayed is determined. The number of words to be read from the image memory, which decreases as the magnification increases, is also computed. The microprocessor loads these parameters into the proper buffer locations on the GCM, and sets a flag indicating that zoom mode is to be turned on. The GCM changes into or out of zoom mode only during vertical retrace.

### Graphics Text

To provide different text sizes and orientations, the microprocessor can draw dot matrix characters directly into the image memory. The smallest character is defined in a cell seven dots wide by ten dots high. It is generated by drawing ten vectors, each seven dots long. Before a vector is drawn, an appropriate pattern



**Fig. 3.** Graphics characters are drawn as a series of adjacent vectors, using a different dot-dash pattern for each vector. The smallest character cell consists of ten vectors, each seven dots long. Characters are rotated or slanted by changing the direction of the vectors.

byte specifying the dot-dash line pattern is read by the microprocessor from a table stored in ROM. To draw characters at different angles, the direction in which the vectors are drawn is changed (Fig. 3). Larger characters are generated by increasing the size of each dot in the dot matrix representation. For example, multiplying the vector length and pattern prescale by three and repeating each pattern three times will draw each point in the matrix as a three-dot-by-three-dot square. The microprocessor can also left justify, right justify, and center strings of graphics text.

### Autoplot

Autoplot allows plots to be made directly from tabular data. The user enters simple parameters about the data into a menu. These menu entries tell the terminal how many columns of data there are, which column is to be used for X data and which for Y, and what the minimum and maximum values are. If tick marks are desired, the spacing between them must be given also. Using this information, the microprocessor will draw the axes and tick marks, with labels and a grid if desired, select the proper data values, scale them, and plot them.

Axis generation is straightforward. The microprocessor reads the values stored in the menu, checks them for possible errors, then uses them to determine where the axes and tick marks should be drawn. When generating tick mark labels, the format of the menu entry is used to determine the format of the label. If the spacing given in the menu entry has no decimal point, the tick labels are written as integers. If the menu entry contains a decimal point, the tick label is rounded to the same number of places after the decimal.

When autoplot mode is turned on, the microprocessor scans all incoming data one character at a time, reconstructs complete numerical values from appropriate ASCII characters, and determines which of the numbers it has built should be used for X and Y data points. The flow chart in Fig. 4 illustrates the process. The scanner starts building a number when a numeric character (0-9, +, -, or .) is detected. Succeeding numeric characters are concatenated onto the value being built. When a non-numeric character arrives, the string being built is terminated. A column counter is then incremented to determine which data column the string is in. The column count is compared with the menu fields for the X and Y data columns, and if a match is found, the string is converted from ASCII to a floating-point representation and stored. When both X and Y values have been received, they are scaled using the values in the MIN and MAX menu fields, and plotted. When the column count exceeds the value in the NO. OF COLS. field, it is

reset to 0.

Only the relative position in the data stream is used to determine which data column a number belongs in, not the physical position on the screen. Con-

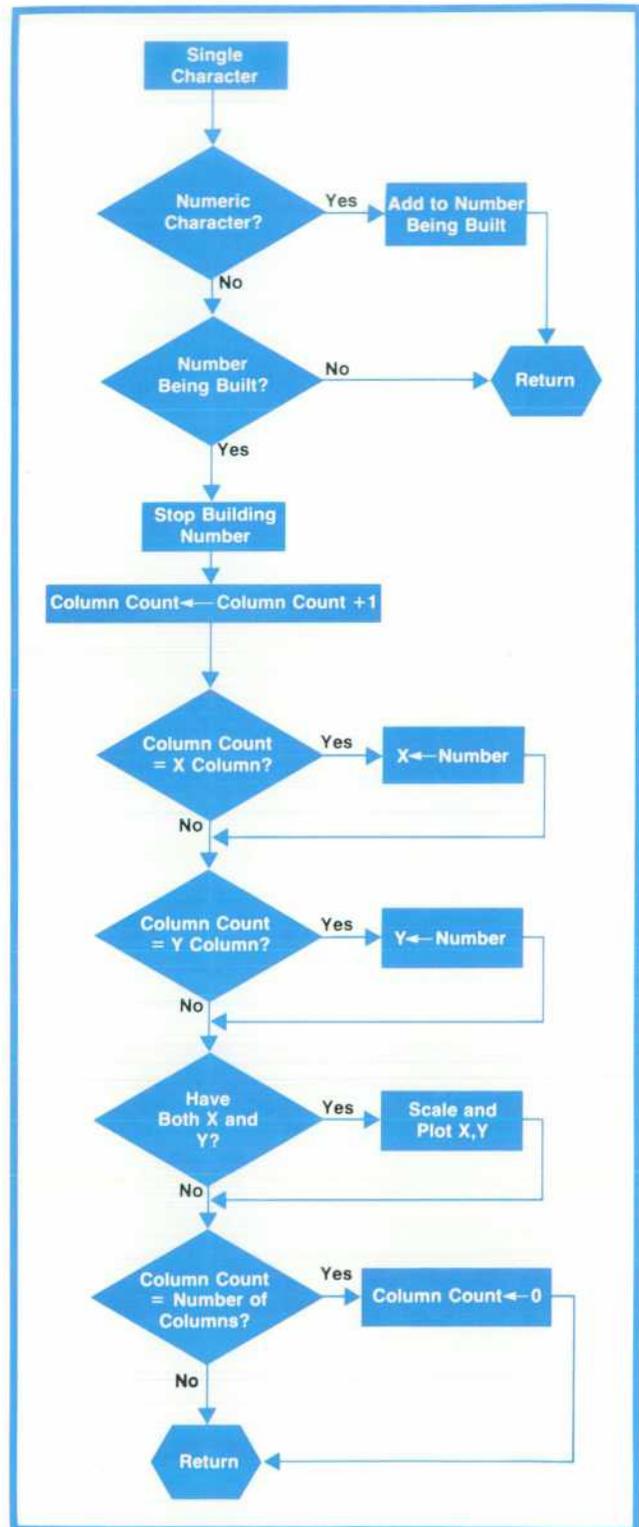


Fig. 4. A flow chart illustrating how the autoplot scanning routine picks out selected columns from tabular data.

sequently, data formatted for 132-column line printers, which will be split across two of the 2648A's 80 character lines, is correctly scanned. Intervening text or blank lines are ignored. By entering twice the number of data columns in the menu, every other point can be plotted.

The source of the plot data can be selected as either the data communications module, the cartridge tapes, or the data being displayed on the screen, which is available to the microprocessor from the terminal's display memory.

### Acknowledgments

I would like to express my appreciation to Ed Tang, Warren Leong, George Hunt, and Rick Palm for their help in interfacing to the existing 2645 firmware; to Pete Showman for his inputs on graphics; to Mike Ramsay and Myron Tuttle for their Q.A. efforts; and to Greg Garland and Bill Woo for their datacom expertise. 



### John J. Moyer

John Moyer received his BA degree in computer science from the University of California at Berkeley in 1975, then joined HP to work on the graphics firmware for the 2648A. He's named as an inventor on several patent applications related to the 2648A. Born in Syracuse, New York, John is single and now lives in Cupertino, California. For recreation, he likes covering distance, either in the air—he's a private pilot—or on the ground, with a pack on his back.

## SPECIFICATIONS

### HP Model 2648A Graphics Terminal

**SCREEN SIZE:** 127 mm (5 inches) × 254 mm (10 in).

**SCREEN CAPACITY:** 24 lines × 80 columns (alphanumeric); 720 dots × 360 rows (graphics).

**CHARACTER GENERATION:** 7 × 9 enhanced (alphanumeric); 9 × 15 dot character cell; non-interlaced raster scan.

**CHARACTER SIZE:** 2.46 mm (.097 in) × 3.175 mm (.125 in) (alphanumeric); 5 × 7 dot character cell (graphics).

**CHARACTER SET:** 128 character (alphanumeric)

**CURSOR:** Blinking-Underline (alphanumeric); Blinking-Crosshair (graphics).

**DISPLAY MODES:** White on black; black on white (inverse video). Optional half-bright, underline and blinking.

**REFRESH RATE:** 60 Hz (50 Hz optional).

**TUBE PHOSPHOR:** P4.

**IMPLOSION PROTECTION:** Bonded implosion panel.

#### MEMORY

ALPHANUMERIC: 37 lines of 80 characters (less enhancements).

GRAPHICS: 720 dots by 360 rows of displayable points.

**OPTION SLOTS:** 4 available.

**KEYBOARD:** Detachable, bit pairing; user-defined soft keys, 18 control and editing keys; graphics pad; cursor pad; auto-repeat, n-key rollover; 1.2-m (4-foot) cable.

**CARTRIDGE TAPE (option):** Two mechanisms

READ/WRITE SPEED: 10 ips

SEARCH/REWIND SPEED: 60 ips

RECORDING: 800 bpi

MINI CARTRIDGE: 110-kilobyte capacity (maximum per cartridge)

#### DATA COMMUNICATIONS

**DATA RATE:** 110, 150, 300, 1200, 2400, 4800, 9600 baud, and external. Switch selectable (110 selects two stop bits). Operation above 2400 baud may require nulls or handshake protocol to insure data integrity. External clocking requires a TTL signal 16 × bps.

**VECTOR DRAWING TIME (9600 baud, typical):** 7 ms half screen; 10 ms full screen.

**STANDARD ASYNCHRONOUS COMMUNICATIONS INTERFACE:** EIA standard RS232C; fully compatible with Bell 103A modems; compatible with Bell 202C/D/S/T modems. Choice of main channel or reverse channel line turn-around for half duplex operation.

**OPTIONAL COMMUNICATIONS INTERFACES (consult 3260A/B/C/D Communications data sheet for details):**

Current loop, split speed, custom baud rates

Asynchronous Multipoint Communications

Synchronous Multipoint Communications-Bisync

**TRANSMISSION MODES:** Full or half duplex, asynchronous.

**OPERATING MODES:** On-line; off-line; character, block.

**PARITY:** Switch selectable; even, odd, none.

#### ENVIRONMENTAL CONDITIONS

**TEMPERATURE, FREE SPACE AMBIENT:**

Non-Operating: -40 to +75°C (-40 to +167°F)

Operating: 0 to 55°C (+32 to +131°F)

**TEMPERATURE, FREE SPACE AMBIENT (TAPE):**

Non-Operating: -10 to 60°C (-15 to +140°F)

Operating: 5 to 40°C (+41 to +104°F)

**HUMIDITY:** 5 to 95% (non-condensing)

**HUMIDITY (Tape):** 20 to 80% (non-condensing)

**ALTITUDE:**

Non-Operating: Sea level to 7620 metres (25,000 ft)

Operating: Sea level to 4572 metres (15,000 ft)

**VIBRATION AND SHOCK (Type tested to qualify for normal shipping and handling in original shipping carton):**

Vibration: .37 mm (0.015") pp, 10 to 55 Hz, 3 axis

Shock: 30 g, 11 ms, ½ sine

#### PHYSICAL SPECIFICATIONS

**DISPLAY MONITOR WEIGHT:** 19.6 kg (43 pounds)

**KEYBOARD WEIGHT:** 3.2 kg (7 pounds)

**DISPLAY MONITOR DIMENSIONS:** 444 mm W × 457 mm D × 324 mm H (17.5 in W × 18 in D × 13.5 in H). 648 mm D (25.5 in D) including keyboard.

**KEYBOARD DIMENSIONS:** 444 mm W × 216 mm D × 90 mm H (17.5 in W × 8.5 in D × 3.5 in H)

#### POWER REQUIREMENTS

**INPUT VOLTAGE:** 115 (+10%–23%) at 50/60 Hz (±0.2%)

230 (+10%–23%) at 50 Hz (±0.2%)

**POWER CONSUMPTION:** 115 W to 150 W max.

**PRICE IN U.S.A.:** 2648A, \$5500. 2648A with cartridge tape units, \$7100.

**MANUFACTURING DIVISION:** DATA TERMINALS DIVISION  
19400 Homestead Road  
Cupertino, California 95014 U.S.A.

# Add-On Digital Signal Processing Enhances the Performance of Network and Spectrum Analyzers

*Digitizing and storing the outputs of network and spectrum analyzers enables flicker-free display of slowly swept measurements, corrections for system errors, and direct comparisons of device performance. Additions to the basic storage circuits achieve improved signal-to-noise ratios and increased resolution.*

by Mark D. Roos, Jacob H. Egbert, Roger P. Oblad, and John T. Barr

**T**HE CONTINUING EVOLUTION of digital signal-processing techniques now allows the instrument designer to add powerful capabilities that were previously not practical because of cost. Storage of CRT displays is a case in point. Digital storage allows the user to make swept-frequency measurements at a slow rate, and then display the acquired data repetitively at a fast enough rate to enable viewing the entire sweep without annoying flicker.

Another advantage of digital signal processing and storage is the capability for applying scalar correction factors. Commonly called normalization, the application of correction factors removes frequency-response errors that often mask the true response when swept measurements are made with less-than-perfect microwave test fixtures (Fig. 1). Normalization has been done with computers in automatic test systems, but with the newer, more powerful, low-cost digital circuits that are presently available, this capability can now be designed into instruments used on the bench.

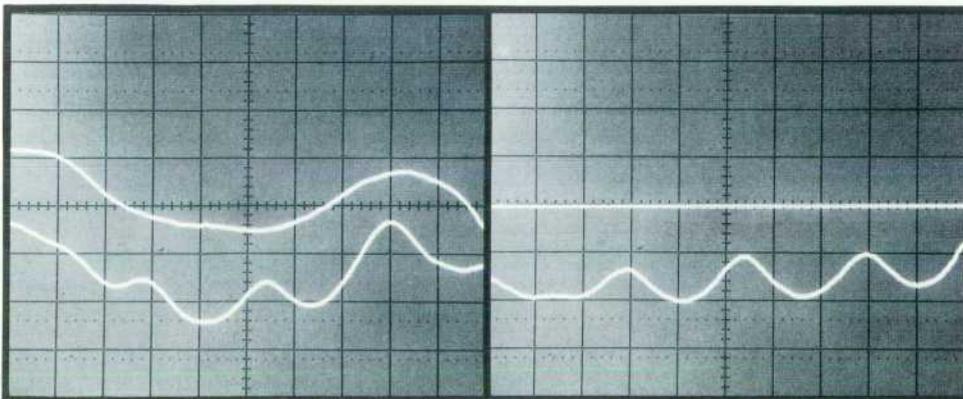
A third useful capability provided by digital storage is the retention of measurement data for compar-

ison with data taken later (Fig. 2). This is useful for matching devices, or for examining characteristics by observing changes in performance while one of the measurement parameters is varied. For example, changes in amplifier gain compression can be monitored as the input signal level is varied.

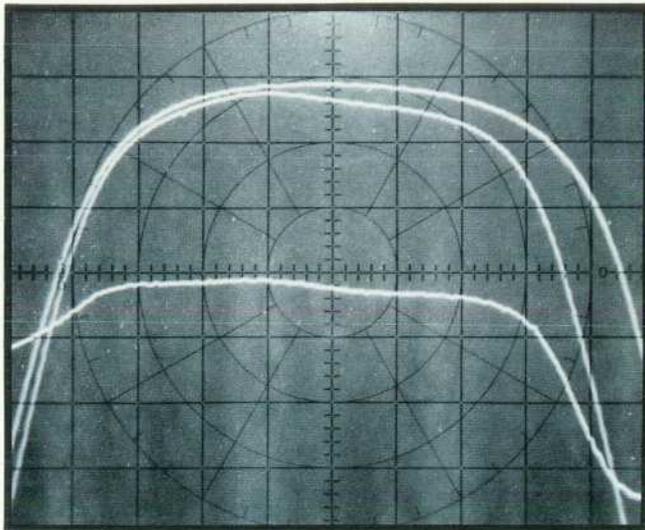
## An Add-on Capability

These and other capabilities have now been designed into two new accessory instruments for use primarily with network and spectrum analyzers. The first of these, Model 8750A Storage-Normalizer (Fig. 3), accepts the X-Y outputs from a network or spectrum analyzer, samples the X-Y outputs during a single swept-frequency measurement, converts the samples to digital words, stores the words, and reads them out repetitively into a digital-to-analog converter. The measurement data is reproduced repetitively at a rate of 167 sweeps per second for flicker-free presentation on the analyzer's CRT. A line generator connects the data points on the display so a smooth, continuous trace is obtained.

Model 8750A can store the data input of two chan-



**Fig. 1.** Digital storage can normalize measurements by retaining a calibration measurement (upper trace in photo at left) and subtracting it from the total system response (lower trace). The result is a display of true system response (photo at right).



**Fig. 2.** Digital storage can retain the result of a previous measurement for comparison with a later measurement (upper two traces). If desired, the difference between the two measurements can be displayed (lower trace). This example shows how well the passbands of two filters are matched.

nels simultaneously for presentation of two quantities, such as amplitude and phase. The stored data can be updated continuously by the analyzer at the measurement sweep rate selected or, when the HOLD pushbutton is pressed, the stored data can be "frozen." The stored data can also be supplied to an X-Y recorder at a rate appropriate to the recorder, giving a hard copy of whatever appears on the CRT screen.

The most powerful capability of Model 8750A is its ability to subtract input data from previously stored data and display the difference. This allows direct comparison of two devices by displaying the difference in their responses, and it allows removal of system residuals from the displayed data (Fig. 1). Formerly, the response resulting from system residuals was usually traced on the CRT with a grease pencil, and the total system response was then compared visually to the pencilled line. When the STORE INPUT pushbutton of Model 8750A is pressed while system residuals are being measured, the input data is stored. Then when the system response is measured and the INPUT - MEM button is pressed, only the difference, i.e., system response minus the residuals, is displayed.

Model 8750A was designed to be compatible with a wide range of HP microwave instruments. It requires only video and sweep inputs and provides outputs compatible with a wide variety of displays. The video input can be supplied by detectors, power meters, or other devices, as well as by network and spectrum analyzers. The X input does not necessarily have to represent frequency but can be power level, position, time or any other parameter that serves as the independent variable.

### Network Analyzer Adjunct

The other new accessory instrument (Model 8501A, Fig. 4), optimized for use with the Model 8505A Network Analyzer,<sup>1</sup> provides the same capabilities plus a number of others. For one, it can store and reproduce polar displays as well as rectangular displays. It can also average the results of several successive measurements to improve the signal-to-noise ratio by as much as 27 dB and it can magnify the data stored in its memory by a factor of up to 10 to improve the resolution of the displayed data (Fig. 5).

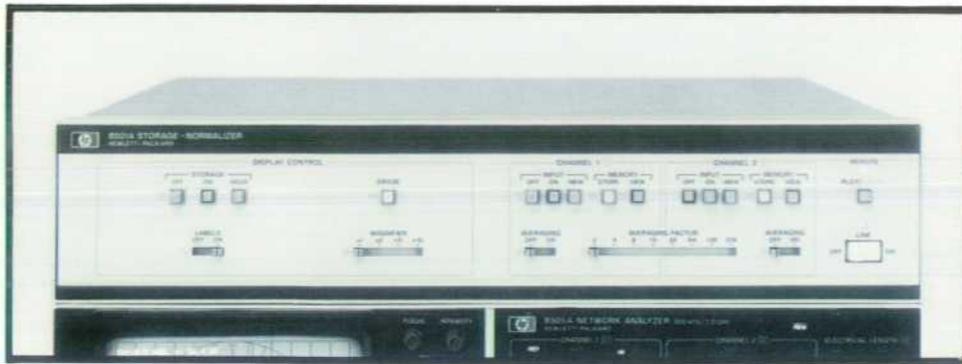
Besides improving signal-to-noise ratios, the ability to derive the average of several readings (see box, page 20) also reduces measurement ambiguities. For example, reflectometers are usually calibrated by making one swept-frequency measurement with the reflectometer output port shorted and another with it open. The user would then average the two measurements to obtain the reflectometer's residual response. With the new Model 8501A Storage-Normalizer, the two measurements can be made quickly, averaged, and then stored for normalizing subsequent measurements (Fig. 5).

Model 8501A Storage-Normalizer is compatible with the HP Interface Bus,\* opening up a whole new range of applications for computer-controlled automatic test systems based on the Model 8505A Network Analyzer. The fast digitizing capability of Model 8501A (500 points in 10 ms) reduces the test

\*Hewlett-Packard's implementation of IEEE Standard 488-1975.



**Fig. 3.** Model 8750A Storage-Normalizer retains the results of dual-channel swept-frequency or other-type measurements for flicker-free display, normalization, and comparison. It works with suitably equipped frequency response sets and spectrum analyzers in HP 180-series mainframes and with the 8407A and 8410A/B Network Analyzers. It can work with spectrum analyzers in HP 140-series mainframes with the addition of a display monitor or oscilloscope. It also works with other types of instrumentation.



**Fig. 4.** Model 8501A Storage-Normalizer is optimized for use with the Model 8505A Network Analyzer. Among other functions, it enables flicker-free display of slow-sweep measurements, normalizes measurements to remove frequency-response errors, and averages repetitive measurements to improve signal-to-noise ratios.

time needed for multiple-frequency go/no-go measurements in a production environment. Model 8501A can also store processed data from an HP-IB system controller and convert the stored information to analog form for display on the network analyzer's CRT. The system is thus able to acquire data in one form and reformat it for display in another form. For example, a swept measurement of reflection coefficient can be reformatted and displayed as input impedance magnitude and phase angle.

Model 8501A has line generators on both the X and Y axes, giving it full graphics capability. Under control of an HP-IB system's desktop controller, the line generators can be used to trace vectors between any two pairs of X-Y coordinates on the network analyzer's CRT, enabling limit lines or complete graticules to be overlaid on measured data. The system controller can aid the operator further by notifying him by an audio or visual message when and where measurement data exceeds limits.

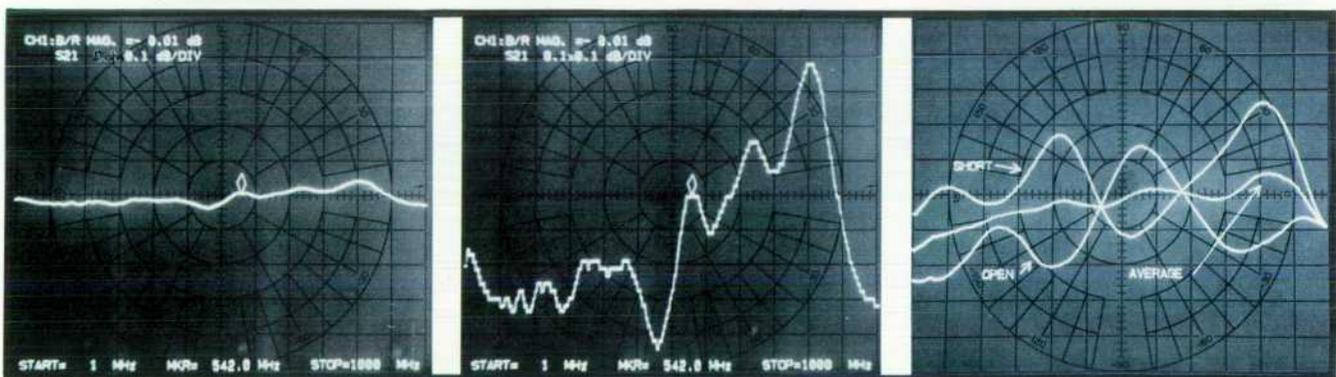
The 8501A also has a built-in character generator that can be used to annotate the displays (serial numbers, dates, etc.) and to present messages to the operator on the network analyzer's CRT (Fig. 6). Up to 22 lines of text can be written using the English and Greek alphabets, numbers, and a complete set of mathematical symbols. Since the controller can also

use the 8501A's graphics capability to generate diagrams of test connections, it is unnecessary to provide written test procedures. Programs for long, involved test procedures can be stored on tape cartridges and entered into the controller as required.

An option enables the new Model 8501A Storage Normalizer to respond to the control settings of a suitably equipped Model 8505A Network Analyzer and, using the built-in character generator, format this information into labels that are displayed on the analyzer's CRT along with the reproduced measurement data. The analyzer's operating parameters may thus be included with the measurement data on CRT photos, an extremely helpful feature for documentation purposes.

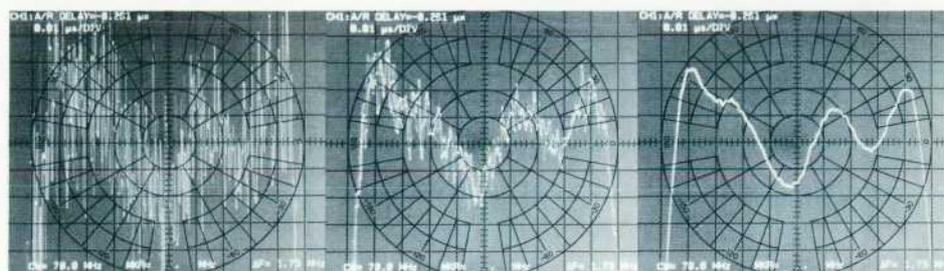
#### Internal Operation

A block diagram of the basic Model 8750A Storage Normalizer is shown in Fig. 7. The vertical (CH1, CH2), horizontal (sweep ramp), and blanking signals from the measurement system enter the 8750A through a plug-in interface card that scales both the inputs and outputs, enabling the 8750A to work with a wide variety of equipment. Two interface cards are provided. One accommodates the requirements of spectrum analyzers and the other accommodates network analyzers (front-panel LED indicators show which



**Fig. 5.** Ten-fold magnification in the vertical direction enables a power splitter's frequency response (photo at left) to be displayed with a resolution of 0.01 dB/div (center photo). Photo at right shows how open and short calibration responses can be averaged (center trace) and retained for normalizing subsequent measurements.

# Signal Averaging Enhances Network Analyzer Performance



**Fig. 1.** Photo at left shows a single measurement of a filter's group delay. Center photo is of a 10-measurement average, showing about a 7.6-dB improvement in signal-to-noise ratio ( $F=4$ ; see Fig. 2). The response in the photo at right has about a 24-dB improvement with the averaging of 500 measurements ( $F=128$ ).

There are many applications, such as some group delay measurements or where the test signal is highly attenuated, in which the signal-to-noise ratio is so degraded that measurement resolution and accuracy are substantially reduced. The Model 8501A Storage-Normalizer makes a major contribution to this class of measurements because of its ability to average the results of several measurements, thereby reducing the effects of noise.

Since the noise present in many measurements tends to average to zero, averaging several successive measurements can reduce its effect (Fig. 1). If "exponentially-weighted" averaging is used, the averaging process can be continuous, with the results of the earlier measurements de-emphasized as new measurement information is added.<sup>1</sup> Adjustments to a circuit can therefore be made while measurements are in progress, and the effect of the adjustment will become apparent.

The exponentially-weighted algorithm is expressed as follows:

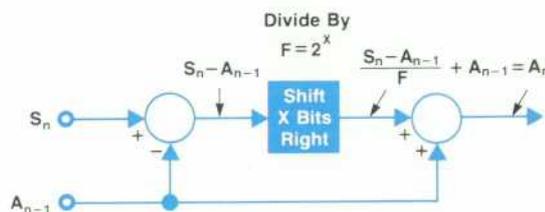
$$A_n = A_{n-1} + \frac{S_n - A_{n-1}}{F}$$

where  $A_n$  = the latest measurement average,  
 $A_{n-1}$  = the previous measurement average,  
 $S_n$  = the current measurement, and  
 $F$  = a fixed integer, the same for all  $n$ .

In other words, the difference between the previous average and the current measurement is obtained and divided by  $F$ . The result is then added to the previous average to obtain the new average.

It can be seen by inspection that if  $F$  is small, the averaged signal quickly adapts to changes in  $S_n$ . If  $F$  is large, the average responds very slowly to changes in  $S_n$  but it also exhibits a much

greater signal-to-noise improvement. The user can thus make a tradeoff between dynamic response and signal-to-noise improvement. It can be shown,<sup>1</sup> that the maximum possible signal-to-noise improvement using this algorithm is  $\sqrt{2F}$ . Fig. 2 shows the theoretical signal-to-noise improvements possible.



**Fig. 3.** Implementation of exponential averaging algorithm.

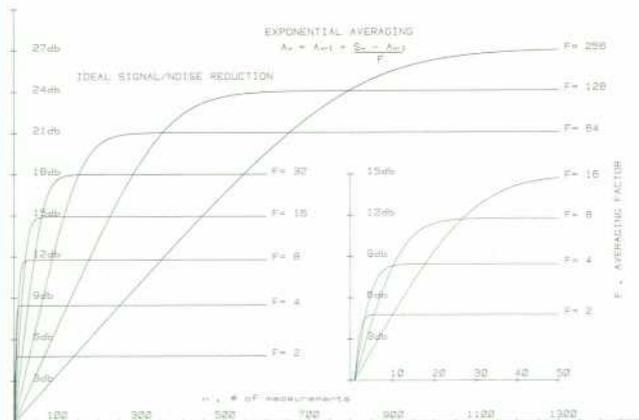
### Implementation

Since this algorithm was to be implemented in digital hardware,  $F$  is chosen such that  $F = 2^x$ , where  $x$  is an integer. The divide operation is then implemented by a right shift of  $x$  bits. A block diagram of the algorithm is shown in Fig. 3.

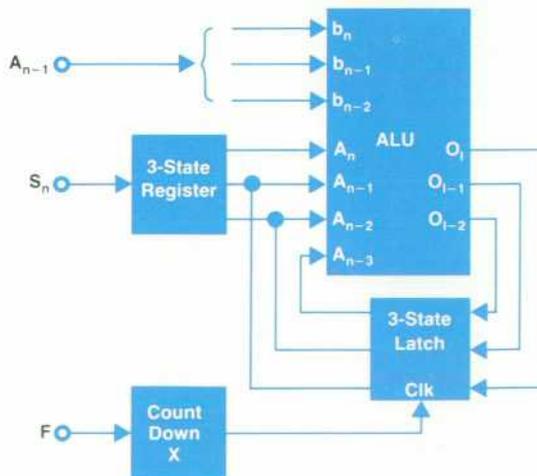
The add and subtract operations are accomplished readily by the standard ALU chips already included in the 8501A for other operations, e.g. normalization. The divide function is accomplished by feeding the ALU outputs back to its input shifted one bit towards the LSB (Fig. 4).

### Reference

1. J.E. Deardorff and C.R. Trimble, "Calibrated Real-Time Signal Averaging," Hewlett-Packard Journal, April 1968.



**Fig. 2.** Theoretical signal-to-noise reduction as a function of the number of measurements and the averaging factor,  $F$ .



**Fig. 4.** Implementation of the divide-by- $F$  function.

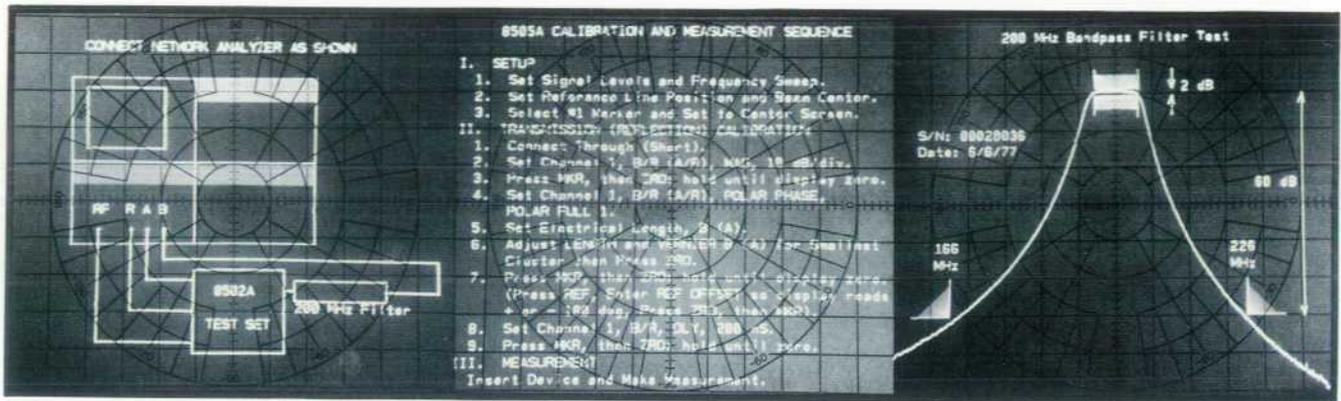


Fig. 6. When the 8501A Storage-Normalizer is used in an HP-IB system, test set-ups (left) and user instructions (center) can be stored by the systems controller on tape as part of a test program and then displayed at the appropriate steps in the program on the analyzer's CRT. Photo at right shows a typical measurement, annotated with limit information, that the user will see during an automatic test sequence.

card is in use).

The channel 1 and channel 2 vertical input signals are processed on alternate sweeps. During a spectrum analyzer sweep, the signal in the selected channel goes to a sample-and-hold peak detector circuit that retains the peak value encountered during a sampling interval in case the test frequency sweeps past the peak of a spectral line during this interval. When the network analyzer interface card is in use and an average-responding display is desired, the peak detector is bypassed.

The X (sweep) input determines both the sampling interval and the memory address where the sampled data is to be stored. It does this by comparing the sweep ramp to the output of a staircase generator (Fig. 8). When the sweep voltage exceeds the staircase

voltage, the comparator fires the one-shot multivibrator. This in turn triggers the sample generator, increments the write address counter, and switches the current source to capacitor C. Capacitor C charges for the duration of the multivibrator's pulse, increasing the reference voltage supplied to the comparator. When the sweep ramp reaches the new reference voltage level, the cycle repeats. This continues until 256 samples have been taken, then everything resets for the next sweep.

To prevent leakage from capacitor C between samples—of special concern during slow sweeps (up to 100 seconds per sweep)—an FET connected as a diode (the lowest-leakage diode available) is used to switch the charging current. Also, the circuit node where the capacitor is connected is enclosed by a

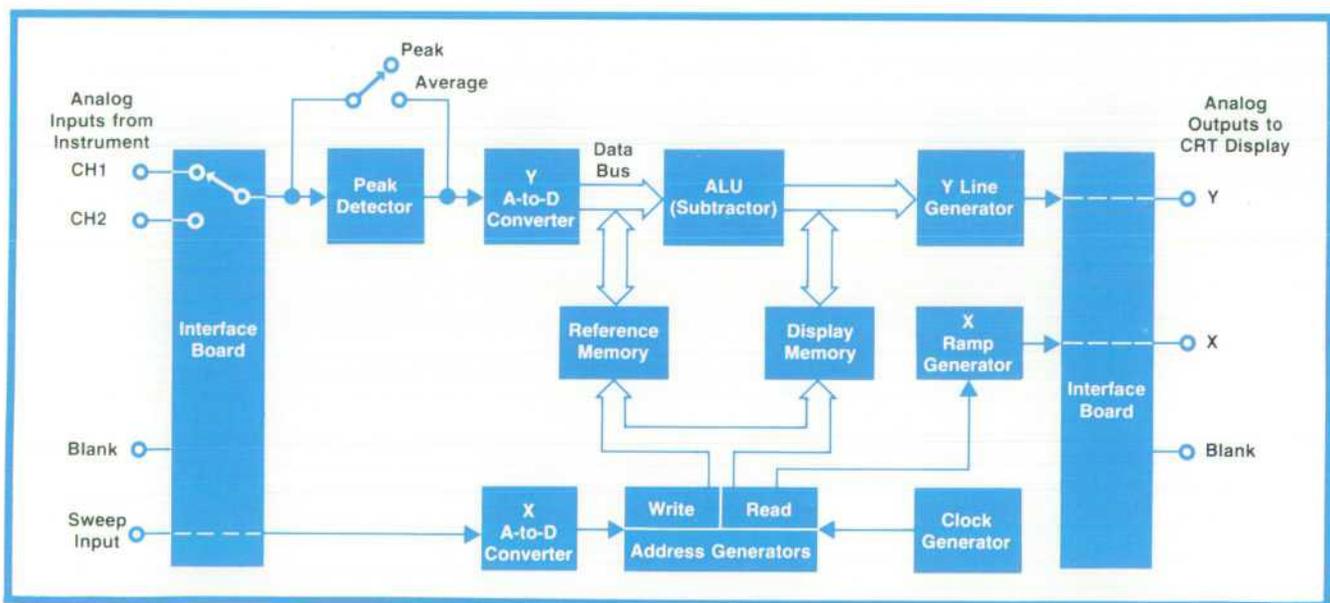


Fig. 7. Block diagram of Model 8750A Storage-Normalizer. Inputs and outputs pass through an interface board that scales the signals according to the instrument being used with Model 8750A.

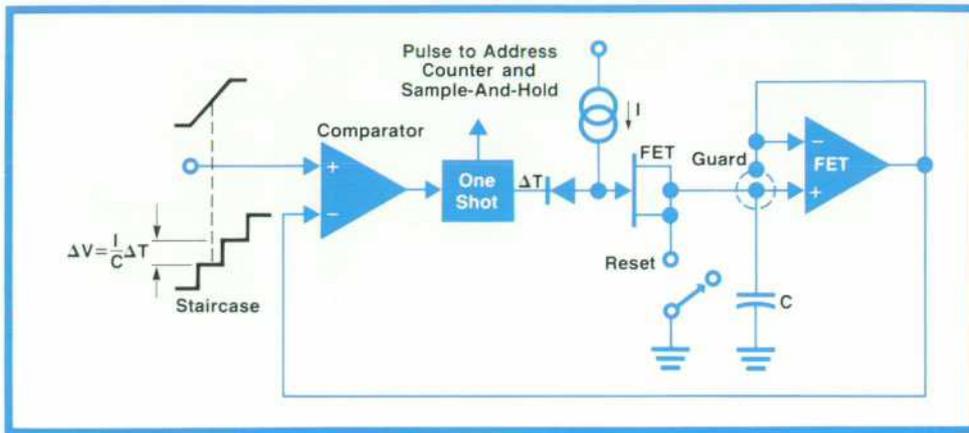


Fig. 8. Staircase generator automatically adjusts its rate, and hence the sampling rate, to the sweep ramp generated by the associated network or spectrum analyzer.

guard ring to prevent leakage across the circuit board.

The Y input voltage is sampled each time the sweep comparator fires and the sample is digitized in a successive-approximation A-to-D converter and sent through the ALU to the memory. During display of the stored information, a 50-kHz clock generator increments the read address register that reads the data out of memory through the line generator to the Y output. The X output is simply a fixed ramp generated in an integrator whose start is synchronized with the readout of the first address in memory.

#### Organization of the 8501A

The basic concept of Model 8501A is similar to that of the 8750A but the 8501A has additional blocks for display annotation and the HP-IB interface. Also, because of the need to process a variety of display formats, including polar data, the full graphics capability was implemented.

The information that must be handled and displayed by the 8501A Storage Normalizer comes in three forms from the 8505A Network Analyzer: rectangular data, polar data, and display annotation information. The display annotation basically contains the front-panel and marker information from the 8505A and is coded in binary form. It is brought to the 8501A over separate signal lines.

The 8501A block diagram is shown in Fig. 9. The rectangular or polar data from the 8505A Network Analyzer is sampled and digitized by the analog-to-digital converter block. The data is then processed by the algorithmic state machine (ASM) and stored in memory. The display section reads the data from memory, formats it, and transfers it to the line generators for display on the CRT.

To process data, the ASM controller detects the sweep start and sets the sweep D-to-A converter output to zero volts. When the sweep input exceeds the

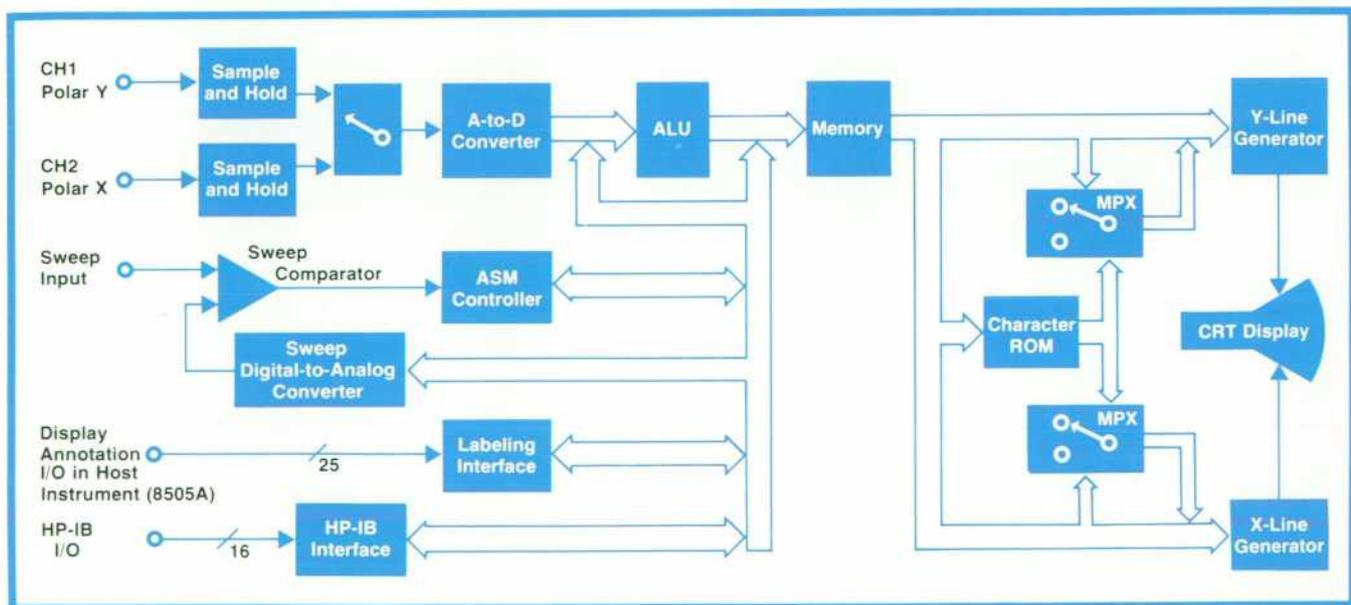
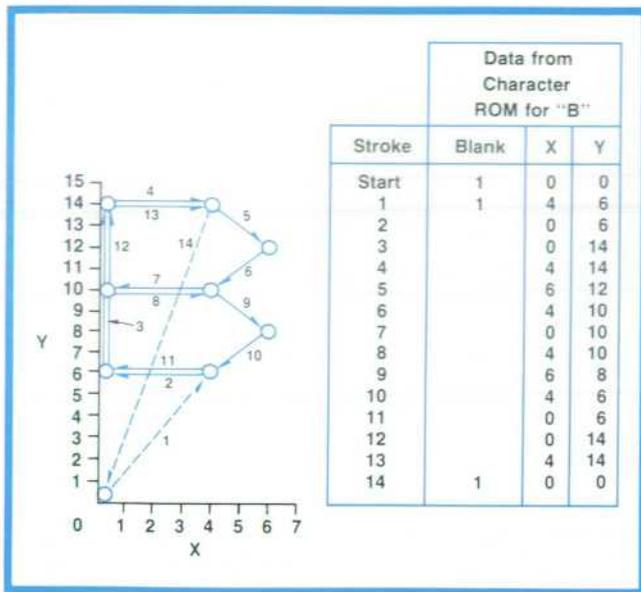


Fig. 9. Block diagram of Model 8501A Storage-Normalizer. A 32-bit algorithmic state machine (ASM controller) was designed to do all the data processing carried out by Model 8501A.

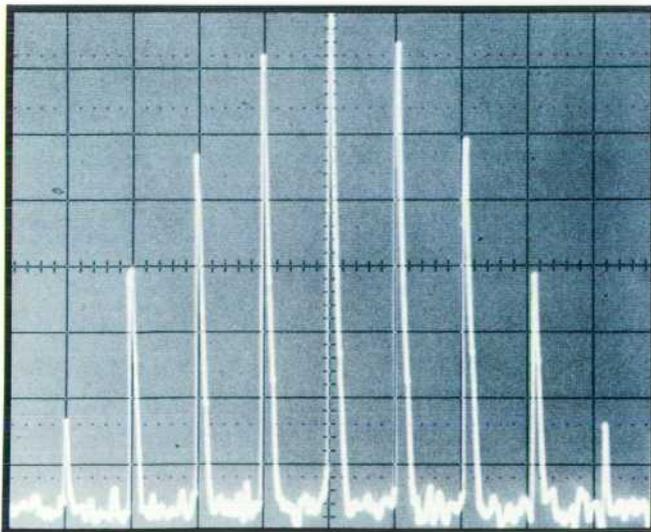


**Fig. 10.** A character is generated by tracing vectors between end points stored in ROM. The longer vectors are traced twice on the CRT to match their intensity to that of the shorter vectors.

D-to-A converter output, the comparator fires. Then the ASM sets the sample-and-hold circuits to the hold mode and initiates an A-to-D conversion. Upon completion of the A-to-D conversion, the ASM processes the data, stores it in memory, increments the sweep comparator register, then waits for the comparator to fire again. This occurs 500 times per sweep.

#### Character Generation

Conversion of the front-panel information from the 8505A Network Analyzer into English messages by the storage-normalizer for CRT display is broken into



**Fig. 11.** Photo of stored spectrum analyzer display shows how the line generator connects data points to give a smooth, continuous trace.

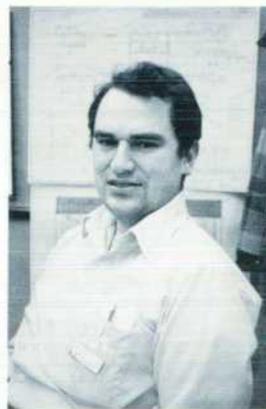
two steps. The first step decodes 8505A front-panel settings and converts them to a string of words and symbols coded in ASCII form. The position information for the letters is also included in this string. The second step takes the ASCII codes and actually draws the characters.

Rather than require that each line segment needed to construct a character be put in the display memory, a special ROM was programmed with all the strokes (short vectors) necessary to implement a 190-symbol



#### Jacob H. Egbert

Graduating from the University of Oklahoma with a BSEE degree in 1969, Jake Egbert completed course work for an MSEE degree before leaving to join a computer firm where he designed bus systems. He joined Hewlett-Packard in 1971, initially working on the 8500A System Console and related systems, then the 8501A. Jake enjoys all outdoor sports, playing in the Santa Rosa city basketball and softball leagues, and enjoying golf and skiing. He has a wife and three daughters, ages 8, 6, and 3.



#### Roger P. Oblad

A native of Salt Lake City, Utah, Roger Oblad obtained a BSEE degree from the University of Utah in 1972 and then joined Hewlett-Packard. At first he worked on the IF detectors in the 8505A Network Analyzer and then moved on to the 8501A project. In the meantime, he earned an MSEE degree from Stanford University in the HP Honors Co-op program. Married, and with four children ages 1 to 6, Roger enjoys camping with the family and swimming.



#### John T. Barr

John Barr joined Hewlett-Packard in 1971 upon getting a BSEE degree from the Georgia Institute of Technology. He worked on the HP-IB interface for the 8505A Network analyzer before moving to the 8501A project, and obtained his MSEE from Stanford in 1974 in the HP Honors Co-op program. John likes to relax by gardening or reading science fiction, or taking his family on camping trips. He has a wife, and a 4-year old daughter with another child expected any moment now.



**Mark D. Roos**

Mark Roos joined Hewlett-Packard in 1973 and worked on the 8505A Network Analyzer (tracking sources) before taking on responsibility for the 8750A Storage-Normalizer. Mark obtained a BSEE degree in 1973 from the California State Polytechnic University, Pomona, and an MSEE degree in 1976 from Stanford University in the HP Honors Co-op program. Outside of working hours, Mark enjoys the outdoors, going in for bicycling, camping, and skiing.

**Acknowledgments**

Special thanks are due Hugo Vifian, Network Analyzer Program Manager, for guiding these projects to completion, and to Doug Rytting and Rit Keiter for leadership and support during the design phase. Thanks are also due Oleg Volhontseff for the product designs, Dave Eng for the 8750A front-panel industrial design, Roy Church for the 8501A industrial design, Al Knack for the design of a ruggedized, shielded cable connector for the inter-instrument cabling, John Ruff for a smooth transition to production, Bob Schaeffer, Sy Ramey, and Jim Fitzpatrick for 8501A applications software support, and Bruce Jones, Steve Hundley, Ken Wong, Mike Heiselman, and Chuck Compton for the instrument retrofits needed to use the 8750A.

character set. When the display-mode control circuit determines that memory data is to be interpreted as text information, it is passed to the ROM and from it the strokes necessary to draw the required character are extracted. An example is shown in Fig. 10.

**Reference**

1. H. Vifian, "A Direct-Reading Network Analyzer for the 500-kHz-to-1.3-GHz Frequency Range," Hewlett-Packard Journal, July 1976.

**ABRIDGED SPECIFICATIONS**

**HP Model 8750A Storage-Normalizer**

**Display**  
**HORIZONTAL MEMORY RESOLUTION:** Two display channels, 256 points per channel.  
**VERTICAL MEMORY RESOLUTION:** 512 points displayed full scale (0.2% of full scale, 10 bit word) plus a 50% overrange (256 points) both above and below full screen.  
**HORIZONTAL INPUT SWEEP TIMES:** 100 s max/10 ms min.  
**DISPLAY REFRESH RATE:** 6 ms.

**Input/Output**

**A/D INPUTS:**  
**HORIZONTAL INPUTS:**  
 NETWORK ANALYZER: 0 to 10V nominal. Offset = 0.5V and gain adjust 6 to 15V.  
 SPECTRUM ANALYZERS: ±5V nominal. Offset = 0.5V and gain adjust = 4.5 to ±5.5V.  
**VERTICAL INPUT:**  
 NETWORK ANALYZER: ±0.6V min. (nominal) and ±2.25V max. (nominal) with continuous gain adjustment. Offset = 0.3V.  
 SPECTRUM ANALYZER: 0 to 0.8V or 0 to -0.8V nominal. Offset = 0.1V and gain adjust = 80 mV.

**D/A OUTPUTS:**  
**HORIZONTAL OUTPUT:**  
 NETWORK ANALYZER: Gain adjustment from 1 to 3V nominal. Offset adjustment allows ±1.5V or 0 to 3V sweep output.  
 SPECTRUM ANALYZER: 0 to 3V nominal. Offset = 5V and gain adjustment from 0.7 to 3.5V.  
**VERTICAL OUTPUT:** Same as Vertical Input with ±10% adjustment range.  
**X-Y RECORDER OUTPUTS (rear panel):**  
**HORIZONTAL RANGE AND ACCURACY:** 0 to 20mV to 1V nominal.  
**VERTICAL RANGE AND ACCURACY:** ±4V ±3%.  
**SWEEP TIME:** 30 s per displayed trace.  
**PEN LIFT:** 30V maximum.

**General**

**POWER:** Selection of 100, 120, 220, or 240V +5% -10%. 48 to 440 Hz and <math>\leq 20VA</math> (<math>\leq 20</math> watts).  
**DIMENSIONS:** 102 mm high, 212 mm wide, and 280 mm deep. (4.0 x 8.4 x 11.2 in).  
**WEIGHT:** 2.72 kg (6 lbs).  
**INCLUDES:** Both network and spectrum analyzer plug-in interface cards are included with internal storage for card that is not in use. One 60-cm direct interface cable for direct connection to 182T, 160TR, 8412A, 8505A, and 8565A. Opt 001 deletes direct interface cable and adds BNC Interface Adapter which allows 8750A and an oscilloscope to be connected to systems like HP 140 series spectrum analyzer.  
**PRICES IN U.S.A.:**  
 8750A STORAGE-NORMALIZER, \$1450  
 OPTION 001 BNC INTERFACE ADAPTER (deletes direct interface cable). No charge.

**HP Model 8501A Storage Normalizer**

**Display**

**RECTANGULAR DISPLAYS**  
**HORIZONTAL MEMORY RESOLUTION:** Two display channels, 500 points per channel (0.2% of full scale).  
**VERTICAL MEMORY RESOLUTION:** 500 points displayed full scale (0.2% of full scale) plus 50% overrange (250 points) both above and below full screen.  
**POLAR DISPLAYS**  
**RESOLUTION:** Two display channels, 250 points per polar display.  
**NORMALIZATION:** Two traces can be stored in memory (one for each channel).  
**DISPLAY MAGNIFIER:** When used, display is magnified by factor of 1, 2, 5 or 10 giving a display resolution of 500, 250, 100 or 50 points in Y (rectangular) and in X and Y (Polar).  
**HORIZONTAL INPUT SWEEP TIMES:** 100 s max/10 ms min.  
**CONVERSION TIME:** 10 ms max for 500 ±2 data points.  
**DISPLAY REFRESH TIME:** Nominally 20 ms depending upon information displayed.

**Input/Output**

**CHANNEL 1 AND 2 INPUTS:** -2.5 to +2.5 V nominal for full scale.  
**SWEEP VOLTAGE:** 0 to 13 V nominal.  
**TRIGGER SIGNAL:** Initiates processing of data when it goes from high to low (standard TTL levels).  
**AUXILIARY OUTPUTS XYZ (BNC female connectors on rear panel):**  
 X: 1 V full screen, 83 mV/div (12 div).  
 Y: 1 V full screen, 100 mV/div (10 div).  
 Z: -1 volt blanks display, +2 volt unblanks display.  
**OFFSETS:** X, Y, and polar display offsets can be adjusted over ±10% range.

**HP-IB Interface**

**LEARN MODE:** Outputs current instrument state to a computing controller.  
**INPUT DATA:** Data for graphics or other purposes can be sent to 8501A at rate of 600 points per second (ASCII mode) or 10000 points per second (Binary mode).  
**OUTPUT DATA:** Data can be read from 8501A at rate of 800 points per second (ASCII mode) or 8000 points per second (Binary mode).  
**GRAPHICS:** Data for graphics can be read into 8501A and viewed in two ways. TEXT DISPLAYS: 22 lines of text with 54 characters per line. Up to 500 characters can be displayed at one time without flicker.  
**VECTOR DISPLAYS:** Lines can be drawn between any two points with resolution of 432 points in x and 360 points in y (nominal).

**General**

**POWER:** Selection of 100, 120, 220, or 240 V +5% -10%. 50 to 60 Hz and <math>\leq 140 VA</math> (<math>\leq 140</math> watts).  
**DIMENSIONS:** 90 mm high, 426 mm wide, 534 mm deep (3 1/2 x 16 1/2 x 21 in).  
**WEIGHT:** 13.25 kg (27 lb).  
**INCLUDES:** HP-IB cable and processor interconnect cable.  
**PRICES IN U.S.A.:** 8501A Storage Normalizer, \$5300.  
**MANUFACTURING DIVISION:** SANTA ROSA DIVISION  
 1400 Fountain Grove Parkway  
 Santa Rosa, California 95404

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

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

**HEWLETT-PACKARD JOURNAL**

**JANUARY 1978** Volume 29 • Number 5

Technical information from the Laboratories of  
 Hewlett-Packard Company

Hewlett-Packard Central Mailing Department  
 Van Heuven Goedhartlaan 121  
 Amstelveen-1134 The Netherlands  
 Yokogawa-Hewlett-Packard Ltd. Shibuya-Ku  
 Tokyo 151 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 Leekma