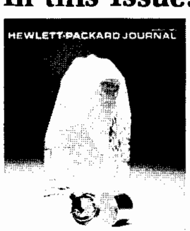
MARCH 1981
HEWLETT-PACKARD JOURNAL

HEWLETT-PACKARD JOURNAL
Technical Information from the Laboratories of Hewlett-Packard Company

MARCH 1981 Volume 32 • Number 3

Contents:

In this Issue:



The shiny rock on this month's cover is a piece of cultured (laboratory-grown) quartz. The thin transparent disc mounted in its holder in front of the rock is a quartz crystal of the type used for frequency control and timing in many electronic devices, including the quartz wristwatches that some of us wear (not all crystals are as large as this one). When these thin slices of quartz are subjected to an alternating voltage, they vibrate, and they vibrate much more strongly at one frequency than at any other. It's this property that makes them useful as frequency and time references.

Quartz is a crystalline material, which means that its atoms line up in a regular pattern or lattice. Thin crystal discs for frequency control are taken from large quartz rocks by slicing the quartz at specific angles to the crystal lattice. Different angles produce different sets of desirable properties. The new HP crystal oscillator described in the article on page 20 derives its frequency stability from a crystal that has been cut to make it relatively insensitive to temperature variations. This property and some state-of-the-art circuit design give the new oscillator, Model 10811A/B, better stability, lower power consumption, and faster warmup than earlier HP crystal oscillators. Model 10811A/B is designed to serve as a highly stable frequency or time reference in precision laboratory instruments, especially those that have to operate for long periods without adjustment.

Pages 3 through 19 of this issue describe a new computer terminal, Model 2626A Display Station, that meets a need of many computer users for a terminal that can handle relatively complex operations. Internally the 2626A can be set up to function as up to four separate "virtual" terminals. The operator can see on the screen what's happening in one, two, three, or all four of these virtual terminals at the same time, and can affect what's happening in one virtual terminal at a time, using the keyboard. Any two virtual terminals can communicate with different computers or the same computer at the same time. This kind of flexibility opens up many new possibilities for computer application systems.

Wrapping up the issue is an article about the 10023A Temperature Probe, a simple device that helps circuit designers find and eliminate circuit hot spots that may indicate problems or likely failure sites.

-R. P. Dolan

New Display Station Offers Multiple Screen Windows and Dual Data Communications Ports

This versatile computer terminal can act like four virtual terminals. It's designed for data entry and program development.

by Gary C. Staas

NEW DATA ENTRY and program development capabilities are provided by a new HP CRT terminal, Model 2626A Display Station, which lets the user display, compare, and combine data from two different computers and four different memory workspaces simultaneously. The new terminal (Fig. 1) has dual data communications ports for connection to computers or peripherals and the user can divide both the display memory and the display screen into as many as four independent work areas. Other features are line widths up to 160 characters with horizontal scrolling, screen-labeled softkeys, an optional built-in thermal forms-copy printer, programmable tones for audio cues, and keys for interactive forms design.

Design Objectives

The objectives that guided the design of the 2626A Display Station are reflected in many of the characteristics of the terminal. An important characteristic is compatibility with the HP 2645A Terminal. Terminal drivers and application programs that work with the 2645A will also work with the 2626A, thereby protecting users' software investments.

The new terminal is also reliable and is easy to build, check out, and service. The reliability goal was 8,000 hours MTBF (mean time between failures). An extensive set of self-tests is built into the terminal, and some of these tests can isolate failures to the component level. One test, which repeatedly executes most of the other terminal tests, is used after terminal assembly to spot failures in the factory. To make the terminal easy to build, there are very few options and the 2626A has many components in common with other terminals in the 262X line. The factory does very little configuration of the terminal. Since that is an easy process,

Fig. 1. *Model 2626A Display Station is a multi-workspace, multi-window computer terminal that has dual data communications ports. Its capabilities may be configured dynamically as four logically independent virtual terminals. The terminal can handle line lengths up to 160 characters and offers foreign language options and an optional built-in printer.*

it is almost totally left to the user.

User sophistication varies considerably, from the data entry clerk to the very knowledgeable OEM customer. Each user needs to take advantage of a different set of features. A department manager of a data entry operation, for example, needs to configure the terminal for use by data entry clerks. The terminal provides configuration menus for this purpose. On the other hand, these configuration menus can be locked out and made unavailable to the clerks to avoid confusing them with details they don't need.

Among many new concepts embodied in the terminal is the ability to handle more than one job at a time. Like several sheets of paper on a desk, each relating to a different task, the 2626A Display Station splits its display screen into as many as four windows, each with independent data. The dual datacomm ports allow these windows to communicate with more than one computer program at once.

The optional built-in thermal printer avoids the problems of expensive, distant, large, and noisy impact printers. It was considered essential to be able to print whatever was on the screen, such as forms and special character sets, and the integral printer makes this possible.

To allow a user to draw simple bar charts without graphics capability, one of the terminal's character sets has been expanded to include appropriate characters. The forms-drawing keys make it easy to design data entry forms.

Foreign language support is an important goal in the international marketplace. The 2626A provides six European languages, including mute and overstrike characters.

The 2626A uses HP's silicon-on-sapphire (SOS) large-scale integrated circuit process. This and other design features make it possible to offer a terminal with a much improved and expanded feature set for a cost comparable to an HP 2645A.

Softkeys

To allow easy access to terminal functions, the 2626A has eight softkeys that do not have fixed functions. A two-row-high label at the bottom of the screen just above each softkey indicates its current function. A function key that indicates a terminal mode, such as REMOTE/LOCAL, has an asterisk on its label when the corresponding function is on. The softkeys can be locked so the terminal stays within a group of softkey levels having similar functions.

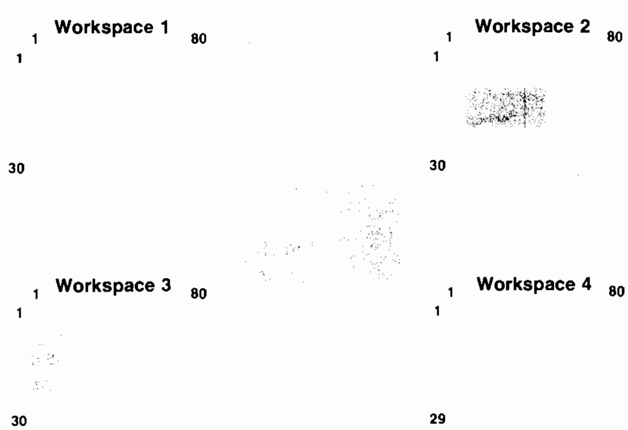


Fig. 2. *In this example, portions of three 2626A workspaces are displayed in windows on the screen. Workspace 4 is not displayed.*

Workspaces and Windows

When the terminal is powered on, it partitions memory into displayable lines of equal sizes. The user can select a length from 80 to 160 characters per line. Lines 132 characters wide, for instance, are useful for holding data to be sent

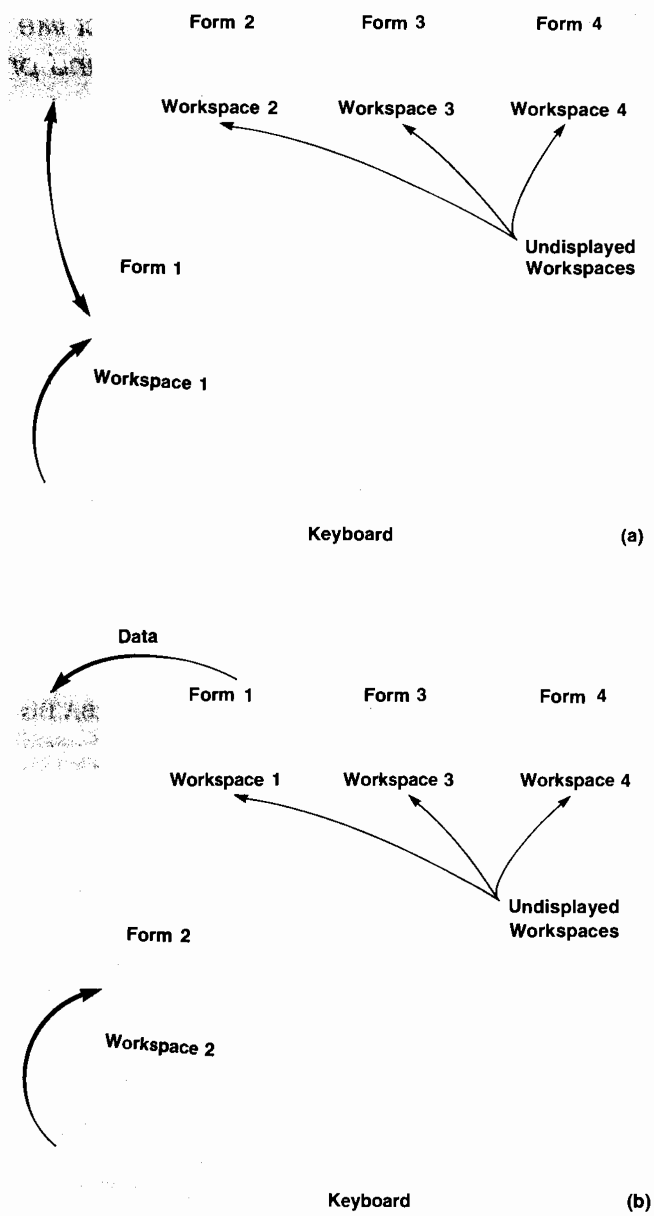


Fig. 3. *(a) In this example, the terminal has four workspaces, each holding a form. Workspace 1 is displayed in the keyboard window and the user can type data into form 1. Workspaces 2, 3, and 4 are not displayed in a screen window. The host computer is connected to workspace/window 1. (b) After the user presses the* ***ENTER*** *key the host displays workspace 2 in the keyboard window and workspace 1 is no longer displayed. The host remains attached to workspace 1 and receives the data just entered while the user fills form 2 in workspace 2.*

to a computer line printer. The user can group these lines into workspaces, each with a fixed number of lines. For example, the user could set up the terminal to have two workspaces of 80-character lines, workspace 1 with 40 lines and workspace 2 with 70 lines. Up to four workspaces are allowed. Multiple workspaces are useful for doing several different jobs on the terminal at the same time.

All workspaces can be displayed on the screen simultaneously to give an overall view of what the terminal is doing. The screen can be divided into display windows that show all or part of the workspaces. To continue the above example, workspace 1 might be viewed in a display window from the first through the tenth screen rows, and workspace 2 might be seen in a window from screen rows eleven through twenty-four. Dotted lines separate windows on the screen to avoid confusion. The terminal also has a vertical border on the screen to allow left and right windows. A workspace need not be displayed to be functional. A workspace that is closed (i.e., has no display window) can receive and send data. Fig. 2 shows an example of a workspace/window configuration.

To avoid ambiguity, all keyboard input affects only one workspace at a time, the one in which the cursor resides. Typed data appears only in this keyboard workspace/window, not in any other window. Local editing keys, such as delete character, take effect only in this window, and so on. The user can change the keyboard window by means of a softkey.

Each workspace operates independently, so that workspaces can be used to perform different tasks with a computer. Most operating modes of the terminal are or can be on a workspace basis. Some of the modes are always associated with a workspace, such as FORMAT mode, which allows a window to be used as a form, limiting the user to typing data into unprotected fields. Other modes are grouped into a terminal configuration, which may be freely attached to workspaces. Suppose that in terminal configuration 1, REMOTE mode is on, and in terminal configuration 2, REMOTE mode is off. If terminal configuration 1 is used for both workspaces, 1 and 2, then both will have REMOTE on. If workspace 1 uses terminal configuration 1 and workspace 2 uses terminal configuration 2, then workspace 1 has REMOTE on and workspace 2 has REMOTE off. There are four terminal configurations so that each workspace can have its own configuration. This is desirable when workspaces are to be used for different kinds of jobs.

The workspace/window configuration can be set locally on a menu or remotely by escape sequences sent from the host computer.

A data communications port can be attached to any one of the workspaces at a given time. Data flow is to and from that workspace only; other workspaces are unaffected. The data entry example in Fig. 3 illustrates how this feature is used. A data entry application program in the host computer first creates four workspaces in the 2626A terminal and attaches its host's data communications port to each workspace in turn to transmit a form to it. After it sends all the forms, it displays the workspace that has the first form and makes it the keyboard workspace so the user can begin entering data into the form. When the user presses the **ENTER** key, the program displays the second workspace with the next form and attaches the keyboard to it. The host program then attaches itself to the first workspace to receive the data just entered on the form. The user continues entering data into the second form while this transfer is in progress. This concurrency results in greater throughput and operator productivity.

To fine-tune a window configuration, the user can move the horizontal and vertical borders with softkeys. Another softkey moves the cursor to the next screen window, and another displays the next workspace that is not currently displayed on the screen. Whenever the line length is greater than the screen width of a window, the data may be scrolled

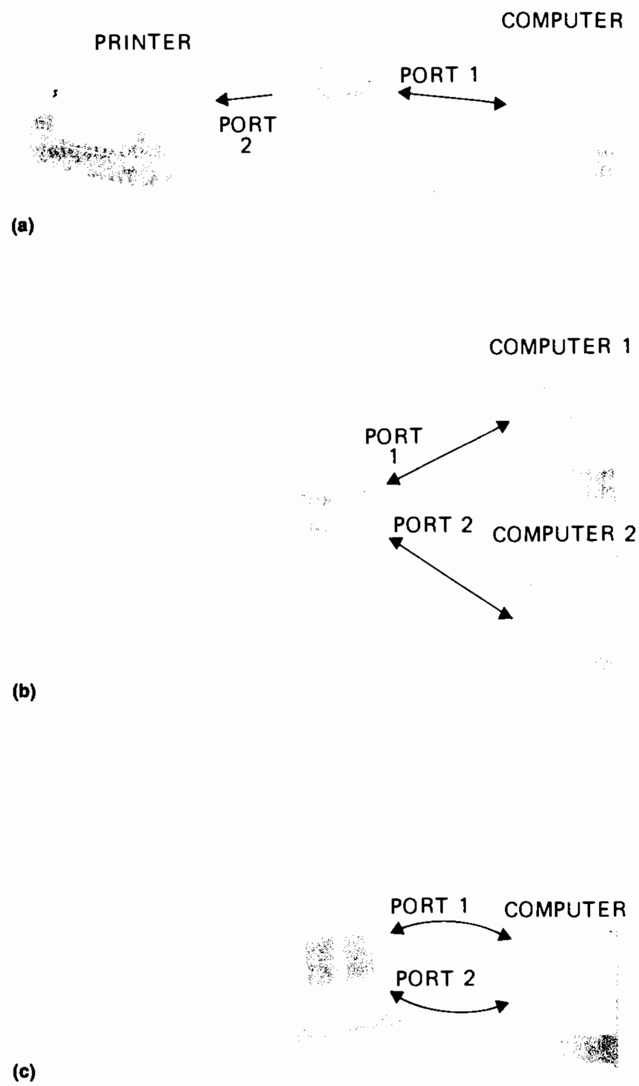


Fig. 4. *Uses of the dual data communications ports. (a) Port 1 is attached to a computer. Port 2, when not in use as a datacomm port, can be attached to a serial printer or other RS-232-C device. (b) Each port is attached to a different host computer. (c) Both ports are attached to the same computer.*

horizontally. If the vertical border were at screen column 30, for example, a right window would have only 50 columns displayed and horizontal scrolling would be necessary to view all the data.

Dual Data Communications Ports

To further enhance its multitasking capability, the 2626A has two ports for data communication. When a port is attached to a workspace, it means that data received from that port is processed and/or displayed in that workspace. If the user enters data in that workspace (by pressing **ENTER** while that workspace is the keyboard workspace/window, for example) the terminal sends data through that port. A port, workspace and terminal configuration together make up a virtual terminal entity.

Fig. 4 shows different possibilities for using both ports. The two ports can be cabled to the same host or to two different computers because each port can use a different data communications protocol. Fig. 5 shows an example using the two ports attached to two different computers. Both workspaces that are attached to ports can be displayed on the screen at the same time in two windows. To make the terminal truly useful for handling multiple jobs, both ports/workspaces can be receiving and sending data and a user can be entering data in a third workspace simultaneously. A useful single-host application is to use one port and workspace for a console and another as a user terminal, or a programmer might examine the output of a compiler for errors in one workspace and edit the source file in another workspace.

Soft Configuration

The 2626A represents an advance over previous terminals in its flexibility and ease of configuration. There are no hardware configuration switches inside the terminal. A user configures the terminal by choosing values on menus built into the terminal. There are configuration forms for workspace/windows, data communications for each port,

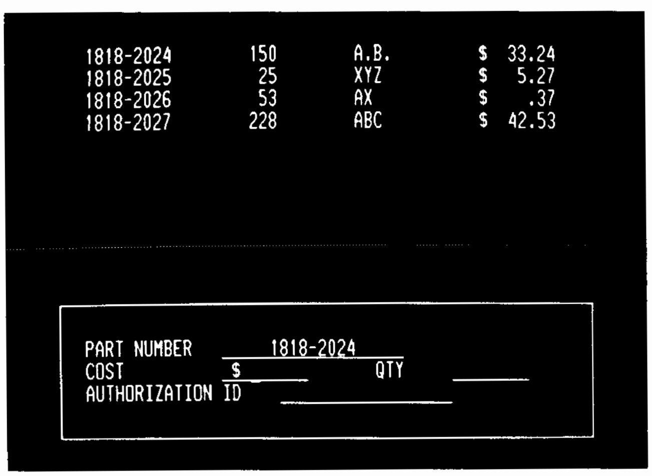


Fig. 5. *In the top window/workspace, a user can examine a data base listing from a computer attached to port 1, and in the bottom window/workspace can get information needed for a data entry program running on a second computer attached to port 2.*

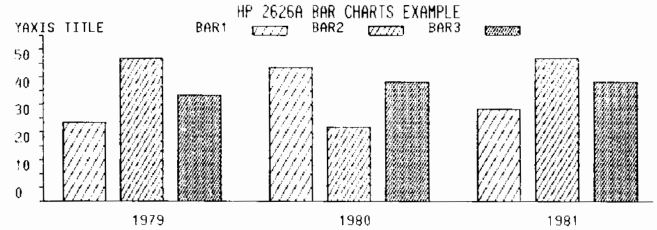


Fig. 6. *New line drawing characters make it easy to draw high-quality bar charts.*

terminal modes for each workspace, and global terminal items. A host computer program can also set any configuration item on any menu. The terminal saves configuration data in battery-powered RAM when ac line power is off. All configurations can be locked to prevent local alteration in applications involving less sophisticated users. If the optional thermal printer is present or if an external printer is attached to port 2, the user can print the configuration menu for further backup.

The 2626A provides both point-to-point and multipoint data communications as standard features. Data communication configuration is done by means of six general menus that cover most full and half-duplex configurations as well as multipoint (multidrop bisynchronous) communications.

Device Control

The 2626A can copy data from one workspace to one or more other workspaces. If the receiving workspace is in REMOTE mode, it will transmit the data just as if it were typed into that workspace. This allows data transfers from one host to another. An external printer can be connected to the second data communications port, which is a standard RS-232-C port when it is not used as a data communications port.

One of the 2626A options is a thermal printer integral to the terminal. The printer is capable of printing all of the terminal's character sets to give exact screen copies. It also provides normal, expanded (40 characters/line), and compressed (132 characters/line) printing modes. When REPORT mode is on, the printer provides three blank lines at the top of each page, 60 lines of text, and three blank bottom lines to format standard $8\frac{1}{2} \times 11$-inch sheets.

A data logging mode enables the terminal to record all the data it receives on a printer. When LOG TOP mode is on, the terminal prints lines as they disappear from the top of a workspace to the destination printer(s) when all display lines for the workspace are used up and the data would be otherwise lost. LOG BOTTOM causes the terminal to log lines as they are received.

Ordinarily, one workspace at a time is copied to the destination device(s). SCREEN COPY copies the screen to the printer(s) exactly as it appears, so that all windows displayed are copied.

User-Definable Softkeys

When they are not being used for terminal configuration or control, the 2626A's softkeys can be used to perform user-defined functions. The user-defined functions are enabled by pressing the **USER KEYS** key. Each softkey may be given a definition consisting of as many as 80 displayable characters. Subsequently, pressing a softkey has the same effect as pressing all the keys in its definition. Two eight-character rows on the screen label each softkey; the labels are defined by the user. Video and character set enhancements may be embedded directly in the softkey labels and the 80-displayable-character key definitions. This allows eye-catching labels and more information in the definitions. The **ENTER** and **RETURN** keys are also softkeys. Another key sets the default values of the user softkeys.

Sketch Forms Facility

The HP 264X family of terminals has a line drawing character set to represent forms. The sketch forms facility of the 2626A makes it much easier to use this set by providing a set of keys to draw horizontal and vertical lines. The drawn lines do not overwrite text or fields. When horizontal and vertical lines cross, the correct intersection character is automatically selected. The user initially selects through softkeys the line type desired (single, double or bold) and the video enhancement desired for the line. Another set of keys automatically draws a box around either the margins (left, right, top and bottom) or the displayed data.

Video Enhancements

Video enhancements and character sets were previously selected only by escape sequence. The 2626A user can also select them with softkeys. The line-drawing, math, and large-character sets have all been expanded to 96 characters. With the new line-drawing characters, for instance, the user can create bar charts as shown in Fig. 6. If a portion of the workspace has the new security enhancement, it displays as blanks, no matter what data is there. This enhancement is useful for password fields and for other secure fields. For users who prefer black-on-white lettering, the black background can be switched to inverse video.

Foreign Languages

The 2626A has ISO character sets for the following European languages: Swedish/Finnish, Danish/Norwegian, French, German, United Kingdom, and Spanish. Keyboards are available for all these languages. The terminal is set to operate for a given language by making a selection on the global configuration menu. All the language capability is built into the terminal; a user needs only the appropriate keyboard with the special character keycaps and key placements and the extra character set ROM.

Fig. 7 illustrates the new language option characters. Some of the languages provide a mute and overstrike capability. For example, when the terminal is configured for French with mutes enabled, typing a circumflex displays it without moving the cursor. When the letter a is typed, the character is replaced by the letter a with a circumflex over it. The terminal provides an eight-bit mode for the HP 300A in which characters are shifted into the alternate character set by setting the eighth bit. When the foreign characters or Roman extension set is selected as the alternate, the terminal can operate with an HP 300A doing foreign processing.

	DECIMAL VALUE										
LANGUAGE	35	64	91	92	93	94	96	123	124	125	126
USASCII	#	@	[	\	]	^	`	{	¦	}	~
Swedish/Finnish	#	É	Ä	Ö	Å	ü	é	ä	ö	å	ü
Danish/Norwegian	#	@	Æ	Ø	Å	^	`	æ	ø	å	~
French	£	à	°	ç	§	^	`	é	ù	è	¨
German	£	§	Ä	Ö	ü	^	`	ä	ö	ü	ß
United Kingdom	£	@	[	\	]	^	`	{	¦	}	~
Spanish	#	@	¡	Ñ	¿	°	`	{	ñ	}	~

Fig. 7. *Special characters, including overstrike characters, are available.*

Terminal Tests and Error Messages

The 2626A offers a powerful set of internal tests. Whenever the terminal is powered on, it performs a self-test to give the user an immediate indication of terminal malfunction. A more comprehensive test is used by production during the burn-in time of the terminal. This test logs any errors found into the battery-powered RAM. The data communications test facility is menu-driven and allows a variety of tests on each port. The integral printer test prints a test pattern on the thermal printer. Each ROM in the terminal has identification information that can be displayed on the screen; this is useful to customer engineers in determining the exact version of firmware in the terminal. All of these tests can be invoked by softkeys and do not require downloading a diagnostic into terminal memory. A customer can perform a test and pass on the results to a customer engineer over the telephone, making service calls less frequent and more efficient.

All of the self-tests produce error messages indicating any problems found. Some user operations can also produce errors, such as certain illegal settings on the configuration menus. If a user attempts to set such a configuration, the terminal displays an explanatory message. All error messages appear on the bottom two lines of the screen replacing the softkey labels until the message is cleared. The rest of the screen remains visible so the user can more easily determine the error.

Additional Features

Among the 2626A's features is a programmable bell with 15 tones, 16 durations, and two volume levels. Several new kinds of terminal status are also offered. For example, window status shows the current workspace/window config-

Gary C. Staas

Gary Staas was born in Dayton, Ohio and attended the University of Dayton, graduating in 1969 with a BS degree in mathematics. In 1971 he received his MS degree in statistics from Stanford University. After three years developing software for hospital information systems, he joined HP's Data Terminals Division, where his responsibilities have included QA of various terminals, data entry system design, and firmware development and feature design for the 2626A and other terminals. Gary is a member of the ACM and a resident of Scotts Valley, California. He lists his interests as skiing, eastern philosophy, running, backpacking, public speaking, and humor. Recently he left HP to pursue his career with another computer firm.

uration, and terminal ID identifies the terminal as a 2626A.

Modify modes allow a user to modify a line on the screen and retransmit it to the host. When the TAB = SPACES feature is on, pressing the **TAB** key transmits the appropriate number of spaces to take the cursor to the next tab stop rather than sending a tab character. **BACK TAB** sends the appropriate number of backspaces to the previous tab stop. This makes it easy to create text files for applications that do not understand tab characters.

Acknowledgments

I would like to acknowledge the HP 2626A project manager, Prem Kapoor, for his many helpful suggestions on features and continual striving for excellence in the product. Stan Telson contributed much to the self-test features and keyboard hardware. Products of this complexity cannot be realized without the active support of lab and marketing management. I would like to thank Lance Mills, lab manager, Tom Anderson, marketing manager, and Terry Eastham and Ken Blackford, product managers, for their encouragement and support.

Display Station's User Interface Is Designed For Increased Productivity

by Gordon C. Graham

A PRIMARY CONTRIBUTION of the HP 2626A Display Station is its advanced user interface, which allows quick access to the many features of the terminal. An important design goal in this project was to make the terminal easy to use. Because this terminal has more features than previous alphanumeric terminals developed by HP, easy access required a thorough, thoughtful approach to the user interface.

Elements of the User Interface

All user interfaces have four elements in common:

User Model. This is the mental model that the user of a product forms as to how the equipment functions. If the product has been carefully designed, the user automatically develops a good understanding, or model, of how it works, and will use the product's features naturally instead of resorting to the operator's manual every time a new situation arises. This area is a very important part of the total design. Each new feature and every change to an existing feature must be carefully evaluated to ensure that it is completely consistent with the operation of all other features, and that its use is natural to the operator.

Command Language. The command language is closely related to the user model and is in fact a concrete representation of it. It is by means of the command language that the user accesses the features of a product. The commands should all relate to each other in a consistent, systematic way, and should include provisions for aborting commands and handling errors.

Feedback from the Instrument. Feedback from the instrument in response to commands tells the user about the completion status of any command. Proper design of feedback mechanisms inspires confidence that the instrument is being used properly and is performing the desired functions. Feedback is also used to point out errors to the user as soon as they are detected.

Information Presented to User. This is closely related to feedback. The central issue is how to display information in a manner that promotes the most effective interaction between the user and the instrument. Both visual and audible means can be used. Generally, the form that visual information takes must be tailored to the characteristics of the display.

Application of these guidelines to the 2626A is mainly reflected in two areas: screen-labeled softkeys and menu-driven configuration.

Screen-Labeled Softkey

Earlier terminals provided access to their features by means of function keys on the keyboard. These keys had a fixed task associated with them, such as home the cursor or clear the display. This approach was satisfactory as long as the function set was small. Later terminals, with many more functions, required the user to type in escape sequences (the **ESC** key followed by one or more other keystrokes) to exercise many of the functions. These escape sequences were designed for use by a host computer system driving the terminal and so were designed for compactness rather than ease of use. The result was that many of the advanced features of these terminals were used only by sophisticated users willing to memorize or look up the escape sequences. To solve these problems, more recent terminals have introduced the concept of tree-structured, screen-labeled softkeys. The 2626A has extended this approach.

Softkeys are keys that do not have a dedicated function, but instead perform many different functions depending upon the state of the terminal. To indicate the current function of a 2626A softkey, a visual representation of the key is displayed on the screen with a label describing the key's current function (see Fig. 1). A softkey may either perform a terminal function or cause a new set of softkey labels to be displayed on the screen, thus assigning a new set of features

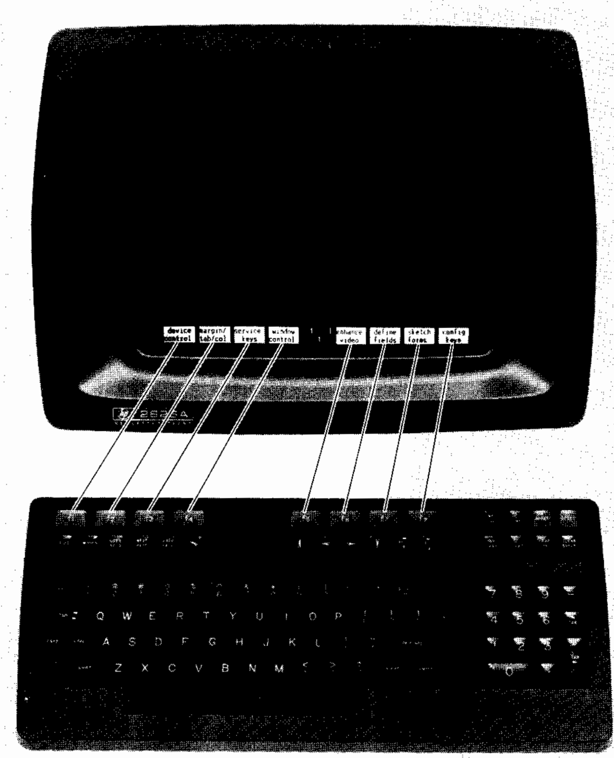


Fig. 1. *The relationship between the eight softkeys and their screen labels.*

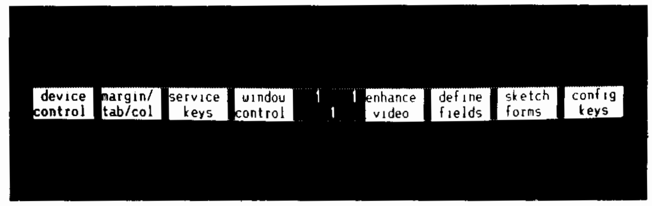


Fig. 2. *The top level of the softkey tree. Each key causes a branch to a separate functional area of the terminal.*

to all of the softkeys. Each set of labels can be thought of as a node of a softkey tree, hence the name tree-structured, screen-labeled softkeys.

Within the terminal family of which the 2626A is a member, a softkey labeling convention has been established. A key that performs a function is labeled in capital letters, while a key that performs a branch to a new set of softkeys is labeled in lower-case letters. Fig. 2 indicates the labels displayed at the top level of the softkey tree. These are all lower-case, so each branches to a separate functional area of the terminal. This is the functional choice level of the softkey tree, often referred to as the AIDS level. There is, in fact, a dedicated key labeled **AIDS** that always displays these labels when pressed, thus providing easy access to the top of the softkey tree.

An example will illustrate several points. Suppose you wish to enhance a certain section of text on the CRT screen by displaying it in blinking inverse video. Pressing the **AIDS** key returns the terminal to the top of the softkey tree. Pressing the fifth softkey, labeled enhance video, then causes a branch to the video enhancement level and displays the labels shown in Fig. 3. The keys labeled INVERSE VIDEO and BLINK VIDEO (**f5** and **f6**) can then be pressed to indicate the desired combination of enhancements. In response to this, asterisks appear in the lower right-hand corners of these key labels to indicate they have been selected. Next, the cursor is positioned on the screen at the starting position for the selected enhancements. When this is accomplished, the SET ENHNCMNT key is pressed to propagate the selected combination. To end the enhancement the cursor is placed at the ending position and SET ENHNCMNT is again pressed. That's all there is to it. The softkey labels direct the user through the entire operation. If a mistake is made when selecting enhancements, pressing the erroneous key a second time toggles the displayed asterisk off, thus reversing its selection.

Contrast this to the older method. First there was no indication to the occasional user that video enhancements were available in the terminal (the enhance video key serves this function in the 2626A). Once aware, the user would have to position the cursor to the desired starting position and type in

escape & d C

to propagate the enhancement, then reposition the cursor and type

escape & d @

to terminate it. These escape sequences had to be memorized or found by looking through the terminal reference manual, either of which requires a much higher level of sophistication than letting the softkey labels direct the user's actions to the desired result. It should be noted that all 2626A features are still accessible to the host system through escape sequences. The softkey tree is simply a user-oriented means of accomplishing the same thing.

The 2626A softkey tree is shown in Fig. 4. Each row of softkey labels represents one level (or node) of the softkey tree and the labels within that level identify the functions available there. The arrows between rows indicate the transitions that are possible between levels using the branching (lower-case) keys.

Relating 2626A softkey operation to the four areas of user interface design leads to the following conclusions. First, extensive use of tree-structured, screen-labeled softkeys significantly improves the user model by breaking the terminal feature set into smaller functionally related areas, each accessible through branching keys. Second, the use of softkeys makes for a particularly simple command language. All features are available with a few keystrokes (often one) and there is virtually no syntax to learn or

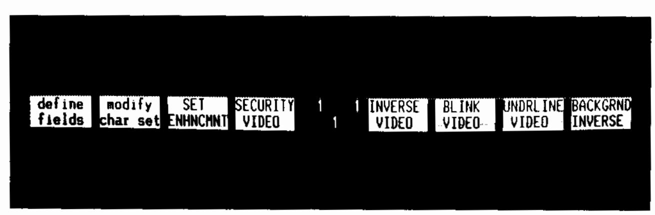


Fig. 3. *The video enhancement level of the softkey tree.*

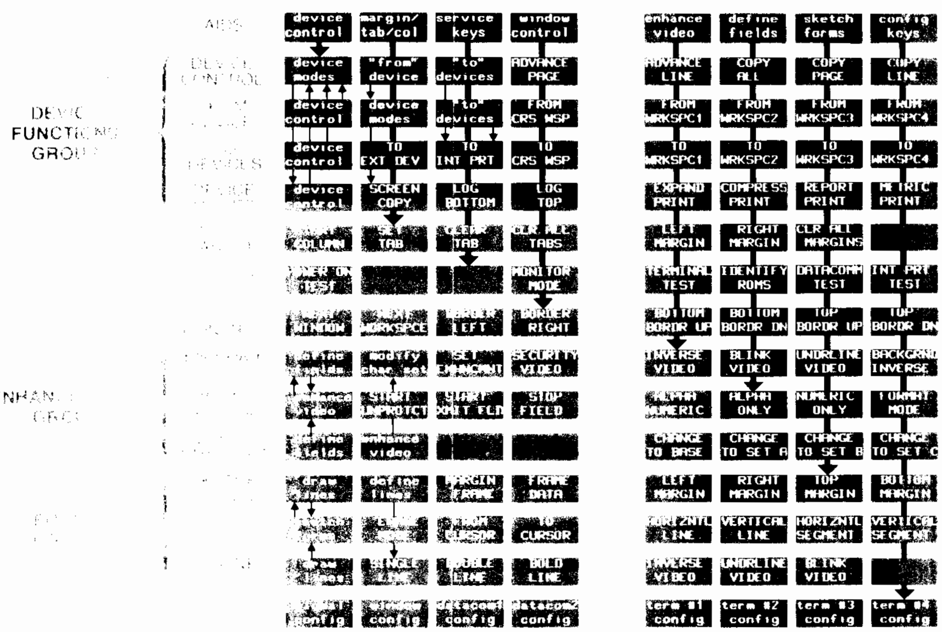


Fig. 4. *2626A softkey tree. Each row of labels represents one level of the tree and the labels within that level identify the functions available there. The arrows between rows indicate the transactions that are possible using the branching (lower-case) keys.*

memorize. Consistent labeling conventions further simplify the command language by visually differentiating between function softkeys and branching softkeys. Third, the structure of the softkey tree presents several opportunities for enhancing the display of information to the user. Since the softkeys are tree-structured, only relevant information is presented. The use of sixteen characters to label key functions exceeds what can be placed on a normal keytop and thus makes it easier to understand the function of the key.

Menu-Driven Configuration

Early terminals were designed for specific purposes and their feature sets did not have a great deal of breadth. Later designs offered features that covered larger portions of the total terminal market. Some of these features were added as optional extras, while some were available as strap options within the terminal. Straps usually took one of two forms, either a physical switch on a printed circuit board or some type of wire jumper. The terminal could be made to exhibit a different feature set simply by modifying the arrangement of internal straps. The user became aware of the straps and their related meanings by reading the reference manual for the terminal. To alter the straps the user had to unplug the terminal, open the case, locate the board containing the strap, remove it, locate the strap, make the adjustment, reassemble the terminal, turn it on, and verify the operation of the terminal with the modified strap.

Strapping the terminal in this way, and setting a few external switches, such as the baud rate and duplex controls for data communications, is called configuring the terminal.

To enhance the user interface in this area and make the full terminal feature set accessible to all users rather than just the skilled few, the 2626A uses a menu approach to accomplish the same result. There are no physical straps inside the 2626A, so a user will never have to open the case. All configuration items are selected simply by choosing menu items displayed on the CRT screen during configuration mode. This approach not only improves the user interface but also eliminates keyboard and internal switches. Thanks to the relative ease of this approach it is now feasible to reconfigure the terminal quickly for different applications.

The terminal's feature set is broken into four smaller areas, each with its own configuration menu. The four areas are: global configuration, window configuration, terminal configuration, and datacomm configuration. Since the 2626A can act like four virtual terminals, there are four terminal configuration menus. Similarly, there are two datacomm configuration menus to accommodate the two datacomm ports.

Global Configuration. Contains configuration items global to all four virtual terminals. Examples are the terminal language (USASCII or one of the six international languages) and frame rate (50 Hz or 60 Hz).

Window Configuration. Contains all items associated with the window/workspace relationships within the terminal. This configuration menu is used to partition the terminal's memory into workspaces and to partition the CRT screen into windows.

Terminal Configuration. Contains all items specific to a given virtual terminal, i.e., virtual terminal modes (remote, block, format, etc.), handshaking requirements, and choice of alternate character sets.

Datacomm Configuration. Contains all datacomm items.

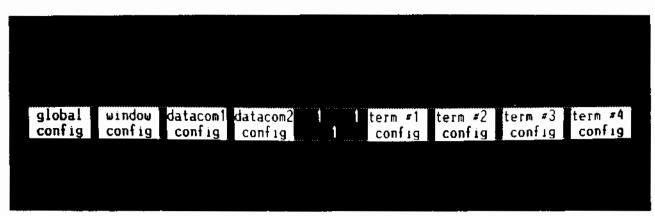


Fig. 5. *The configuration level of the softkey tree.*

One first chooses from a set of menus describing the various datacomm protocols supported by the terminal and then fills out the configuration items for that particular protocol.

An example will best illustrate the use of configuration menus. Suppose that the terminal language is to be changed from USASCII to Danish. Like all other features the configuration menus are accessed through the softkeys. Pressing the **AIDS** key returns the terminal to the top softkey level. Then, pressing config keys (**f8**) causes a branch that displays a set of softkey labels allowing access to all of the configuration menus in the terminal (Fig. 5). Since the terminal language is a global item, pressing **f1** (global config) brings up the appropriate menu (see Fig. 6). Note that the first configuration item on the second line of this menu is the language specification.

To change this the cursor is positioned in the field and the softkey labeled NEXT CHOICE (**f3**) is used to scroll through a list of all available languages. Each depression of this key displays the mnemonic associated with one of the languages supported by the terminal. For our example, two depressions are required to display the desired mnemonic, DANSK/NORSK (Danish/Norwegian). At this point any other changes required in the global configuration menu can be made using a similar approach. When all changes have been completed and the SAVE CONFIG key (**f1**) pressed, all changed items are activated. Typically, this is all that is required to change any configuration item in the terminal. Once again, the softkey labels combined with English-language field names in the configuration menu lead the user through the operation.

Within a configuration menu the cursor can be quickly advanced from field to field with a single depression of the **TAB** key. the **BACK TAB** key returns it to the previous field, allowing quick access to all menu items.

Values are selected for each configuration item in one of two ways. If the value field of the configuration item is underlined, it implies that the NEXT CHOICE/PREV CHOICE softkeys are to be used to scroll forward/backward through a predefined list of values for that field (as in our example). If not underlined, then the user types in the choice directly from the keyboard. Depressing **f4** (DEFAULT VALUES) or **f5** (POWER ON VALUES) causes the entire menu to be refreshed

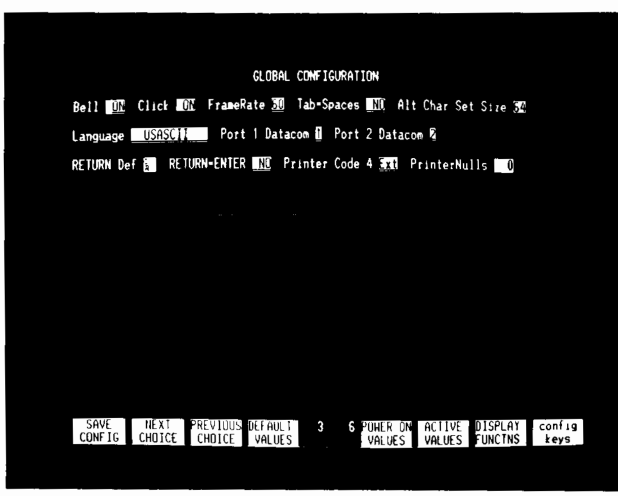


Fig. 6. *Global terminal configuration menu.*

with either the default values stored in ROM or the power-on values stored in nonvolatile RAM.

Let's review the above features with respect to the four guidelines for designing user interfaces. As with the softkeys, the configuration menus enhance the user model by dividing the entire terminal function set into smaller related groups. All items associated with each functional area are presented on the same configuration menu, while unrelated items are always contained on other menus. This helps the user develop an understanding of the terminal by understanding its subfunctions.

The command language used to configure the 2626A is very simple. Once the appropriate menu is displayed, items to be modified are selected with the cursor and the user either types in the new value or selects from a predefined list of values. Again, there is no syntax to learn or remember.

Feedback to the user takes several forms during configuration. When using the NEXT CHOICE/PREV CHOICE keys the value of each choice is fed back via an English-language mnemonic in the selected field. When values are entered directly from the keyboard, the range of each value is checked and the consistency of all menu items is verified. When erroneous entries are detected the user is informed by error messages in the softkey label area, and the cursor is placed in the offending field to aid in correcting the error.

It is particularly in the area of presenting information to the user that the configuration menus make a large contribution. In the older schemes, each strap was given a letter identifier. This letter was then associated with its function in the reference manual for the terminal. With configuration menus, each "strap" is given an English-language mnemonic describing its function. The equivalent letter identifier is parenthetically included to maintain consistency with older products and with existing applications that may refer to straps by letter only. The fact that all choices available to the user are displayed at once greatly enhances the user's ability to make quick, accurate changes in the configuration without resorting to a manual. Since the NEXT CHOICE/PREV CHOICE keys present only valid data to the user, the possibility of error is greatly reduced.

Acknowledgments

The design of the user interface for the 2626A represents the contributions of a large number of people. In particular, Maxine Brown developed the framework for both the softkey tree and the soft configuration menus. Prem Kapoor, project manager for the 2626A, was instrumental in developing the softkey tree to its present state of refinement.

Gordon C. Graham

Gordon Graham did firmware programming for the 2626A Display Station. He joined HP in 1979 with several years of experience in programming and circuit design. A native of Covina, California, he received his BSEE and MSEE degrees from the University of California at Los Angeles in 1971 and 1975. He's married, has a daughter, lives in San Mateo, California, and enjoys photography and bicycling.

SPECIFICATIONS

HP Model 2626A Display Station

SCREEN SIZE: 150 mm × 215 mm (6 in × 8.5 in) diagonal measurement 262 mm (10.4 in).

SCREEN CAPACITY: 24 lines × 80 columns (1920 characters). 25th and 26th for labeling of function/softkeys or as message/status lines.

CHARACTER GENERATION: 7 × 11 enhanced dot matrix with interstitial dots; 9 × 15 dot character cell; non-interlaced raster scan.

CHARACTER SIZE: 2.4 mm × 3.5 mm (.094 in × .138 in).

CHARACTER SET: Upper/lower case, displayable control codes, extended line drawing. Optional math symbols, large characters, Finnish/Swedish, Danish/Norwegian, French, German, Spanish and U.K. characters.

CURSOR: Blinking-underline, blinking-square for insert mode.

DISPLAY ENHANCEMENTS: Inverse video, underline, blink, security.

REFRESH RATE: 60 Hz (50 Hz optional).

TUBE PHOSPHOR: P4.

IMPLOSION PROTECTION: Tension Band.

MEMORY: 80 characters by 119 lines including a default datacomm buffer. (Buffer expandable to 2K bytes—reduces memory to 80 characters by 107 lines.) 128 bytes non-volatile configuration memory (battery powered).

KEYBOARD: Full ASCII code keyboard; eight screen-labeled keys; cursor controls; 14-key numeric pad; auto-repeat; N-key rollover; detached with 1.2-m (4 ft) cable. Optional Finnish/Swedish, Danish/Norwegian, French, German, Spanish, and U.K. keycaps.

Data Communications

DATA RATE: 110, 134.5, 150, 200, 300, 600, 1200, 1800, 2000, 2400, 4800, 9600 baud and external. Operation with control codes, escape sequences, integral printer, or baud rates above 4800 may require CPU supplied delays or handshakes. Typical printer throughput is 60 cps. Full 24 × 80 line character screen copies in 18 seconds.

PORT 1 ASYNCHRONOUS/SYNCHRONOUS INTERFACE (50 PIN): EIA standard RS-232-C; fully compatible with Bell 103A, 202C/D/S/T modems. Choice of main channel and/or reverse channel in half-duplex. CCITT V.24 hardware handshaking available. Accessory pods provide current loop (13266A), asynchronous or synchronous multipoint (13267A first terminal, and 13268A daisy-chain terminal. 300 Baud Modem 13268A U.S. only).

TRANSMISSION MODES: Full or half-duplex, asynchronous, synchronous.

PORT 2 ASYNCHRONOUS/SYNCHRONOUS INTERFACE (25 PIN): EIA standard RS-232-C; fully compatible with Bell 103A modems, 202C/D/S/T modems. Choice of main channel and/or reverse channel in half-duplex. CCITT V.24, hardware handshaking for control of external printer.

TRANSMISSION MODES: Full or half-duplex, asynchronous, synchronous (does not support HP Multipoint polled protocol).

OPERATING MODES (BOTH PORTS): On-line; off-line; character, line modify, line and block.

PARITY: Selectable; even, odd, none, 0, 1.

General

POWER REQUIREMENTS

INPUT VOLTAGE:	100/120V (+5%, −10%) at 60 Hz (±5%)
	100/220/240V (+5%, −10%) at 50 Hz (±5%)
with Option 050	115V (+10% −25%)/230V (+10%, −15%) at 50/60 Hz (±5%)
POWER CONSUMPTION:	75W
with Option 050	120W

ENVIRONMENTAL CONDITIONS

TEMPERATURE, FREE SPACE AMBIENT: Non-Operating: −40 to +60°C (−40 to +140°F).
Operating: 0 to +55°C (+32 to +131°F).
WITH OPTION 050: Operating: +5 to +40°C (+41 to +104°F).

HUMIDITY: 5 to 95% (non-condensing).
WITH OPTION 050: 5 to 80% (non-condensing).

ALTITUDE: Non-Operating: Sea level to 15240 metres (50,000 ft).
Operating: Sea level to 4572 metres (15,000 ft).

VIBRATION AND SHOCK: Vibration: 0.38 mm (0.015 in) pp, 5 to 55 Hz, 3 axis
Shock: 20 g, 11 ms, ½ sine.
Type tested to qualify for normal shipping and handling in original shipping container.

PRODUCT SAFETY

Product meets the requirements of the following safety agencies for EDP equipment and/or office equipment in the following countries: Canada—CSA, Finland—FEI, Germany—VDE, Switzerland—SEV, U.K.—BSI, United States—U.L.

PHYSICAL SPECIFICATIONS

DISPLAY MONITOR WEIGHT: 16.8 kg (37 lb).
With Option 050: 19.0 kg (42 lb).

KEYBOARD WEIGHT: 2.0 kg (4.4 lb).

DISPLAY MONITOR DIMENSIONS: 380 mm W × 475 mm D × 440 mm H (15.0 × 18.7 × 17.3 in); 665 mm D (26.2 in) including keyboard.

KEYBOARD DIMENSIONS: 430 mm W × 190 mm D × 75 mm H (17 × 7.5 × 3.0 in).

ORDERING INFORMATION:

		Prices in U.S.A.
2626A	Display Station. ASCII keyboard with numeric pad, 80 character by 119 lines display memory, serial I/O port, 60 Hz, 110 volt operation.	$4150
Option		
001	Finnish/Swedish character set and keyboard†	265
002	Danish/Norwegian character set and keyboard†	265
003	French character set and keyboard†	265
004	German character set and keyboard†	265
005	United Kingdom character set and keyboard†	265
006	Spanish Language character set and keyboard†	265
013	240V, 50 Hz operation	---
014	100V, 60 Hz operation	---
015	220V, 50 Hz operation	---
016	100V, 50 Hz operation	---
050	Integral forms copy thermal printer, 120 characters-per-second using 8½-inch-wide paper	1210
201	Math and Large Character sets. (Standard with any language option)	265

†Deletes U.S. keyboard and includes math and large character sets.

MANUFACTURING DIVISION: DATA TERMINALS DIVISION
974 East Arques Avenue
Sunnyvale, California 94086 U.S.A.

Srinivas Sukumar performed the detailed design of the configuration menus. Special appreciation is given to Tom Anderson for his help in defining the user interface for the windows and workspaces in the 2626A. The untiring efforts of Gary Lum and Frank Santos in verifying proper operation of the 2626A deserve special mention.

Reference

1. W.M. Newman and R.F. Sproull, "User Interface Design," Principles of Interactive Computer Graphics, Second Edition, Chapter 28. McGraw-Hill Book company, New York, 1979.

Hardware and Firmware Support for Four Virtual Terminals in One Display Station

by Srinivas Sukumar and John D. Wiese

THE 2626A DISPLAY STATION is a sophisticated CRT terminal with features that form a superset of the 2645A Terminal's feature set. With its windowing capabilities and dual communications ports, the 2626A can function as up to four virtual terminals. It has a custom SOS video controller chip, 80K bytes of ROM, 32K bytes of display RAM and 2K bytes of program RAM for variables. A fast character ROM containing ASCII and extended line drawing characters is a standard feature.

The 2626A represents a significant contribution to CRT display terminal capabilities. The following objectives guided its design and development:

- 2645A-compatible feature set. Because of HP's large established customer base, it was considered mandatory that all host applications and device drivers developed for the 2645A function as well as with the 2626A.
- Improvement in price/performance ratio. To meet the needs of increasingly sophisticated users and take advantage of new technology, a significant enhancement of the feature set was required.
- Ease of use. There are many useful features in the 2645A that are tedious to use and require a detailed knowledge of the terminal's control sequences. A good example is the design of forms. The HP 2626A provides easier access to all terminal features. There are no hardware straps to be set by the user. Instead, menus provide a highly visible means of configuring the terminal and displaying its current state.
- Ease of manufacturing, reliability, and easy serviceability.

Firmware Design

The terminal's feature set is provided by a combination of hardware and firmware (microprograms stored in read-only memory). One of the main goals of the firmware design was modularity. This is achieved by a clear specification of the firmware interfaces. The firmware modules called intrinsics in the 2626A provide the interface between the hardware and the main code so that the main code is not overly burdened with the hardware aspects of the design.

The firmware in the 2626A is divided into six major parts: the main code, the operating system, the display intrinsics, the keyboard intrinsics, the datacomm intrinsics, and the printer intrinsics. The main code controls all the terminal's features, thereby providing the 2626A with its personality. All input processing beyond the interrupt service routines is done by the main code.

Traditionally, terminals do not have an operating system. The 2626A has a simple operating system that is adequate for its needs. In studying the various functions that the terminal has to perform, the following tasks can be identified:

- Input from datacomm port(s)
- Datacomm output(s)
- Keyboard input
- Block mode process. A block of data is transmitted when the **ENTER** key is hit.
- Device transfers. Data is transferred from the datacomm port to devices like workspaces and printers.
- User softkey processing. A user-defined string of characters is processed when a softkey is pressed.

The operating system on the HP 2626A is designed to control the execution of these tasks as the system (terminal) is responding to interrupts from its I/O ports.

A task in the 2626A may be in one of three states: waiting, scheduled or executing. If a task is currently executing, in the absence of a hardware interrupt it remains in control until it gives up control to the operating system by WAITing or YIELDing. That is, the operating system never specifically takes control away from an executing task. After the task WAITs or YIELDs, the operating system looks to see if another task has been scheduled and executes the first one that it encounters. It examines all tasks in a round-robin fashion such that the last task executed is the last task checked when the search is made to find another task to which to give control.

How long a task remains in control depends on the device or function associated with the task. For example, all tasks associated with I/O devices WAIT after all the characters in a burst from that device have been processed. But these tasks YIELD after processing a certain fixed number of characters in a burst. The datacomm tasks process 256 characters before YIELDing, but the keyboard YIELDs after each character processed. Tasks controlling BLOCK mode and device transfers WAIT after the complete data transfer is done. But they YIELD after processing one line of display memory.

Interwoven into this multitasking system is the concept of windows and workspaces. A workspace is a block of display memory that can be associated with a display screen window. Traditionally a workspace is all of display memory in a terminal and a window is the 24-line screen that is currently visible. The 2626A has four possible workspaces, with the number of columns in each configurable from 80 to 160 characters. The total in all workspaces is 119 lines of 80 characters. As the line width increases the number of lines available decreases proportionally.

Each task in the terminal is associated with a workspace. This association is established through the terminal workspace/window configuration, which can attach a datacomm port to a workspace and the keyboard to another workspace. Thus the 2626A can process inputs from both datacomm ports and the keyboard simultaneously.

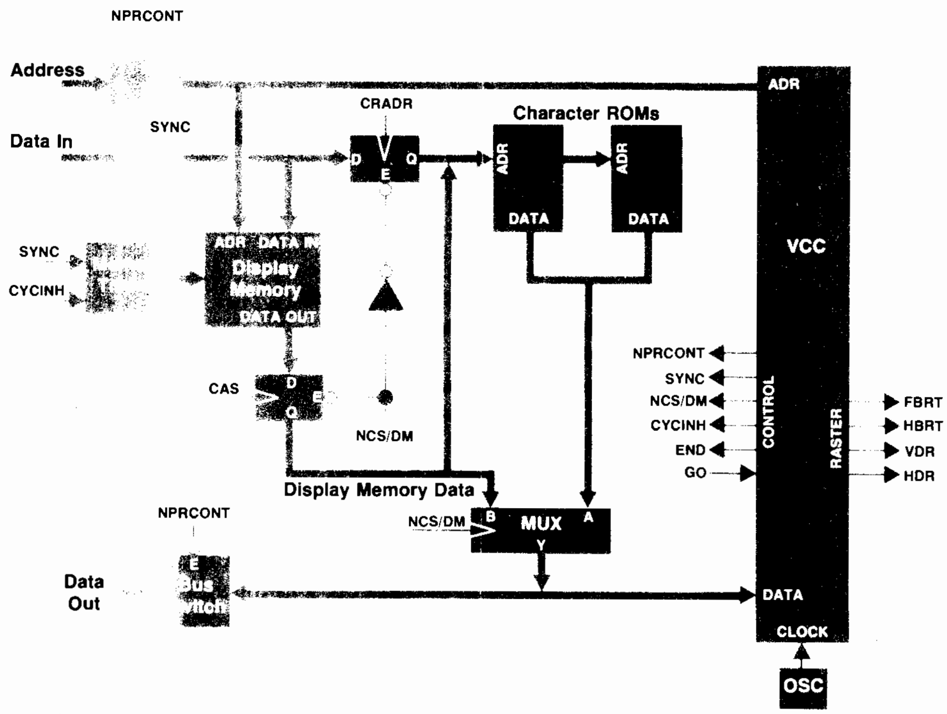


Fig. 1. *The 2626A display section is controlled by the video control chip (VCC). This large-scale integrated circuit makes the display section very simple.*

Terminal Intrinsics

The display intrinsics are designed to interface with the video controller (see article, page 16) and the display memory. These also have the workspace/window concepts built into them. At any point in time all the functional intrinsics operate on the currently active workspace. Whenever a device becomes active, the workspace associated with that device gets activated and the function to be performed by the character is executed in that workspace, whether it involves displaying the character or executing the control function associated with it. The display intrinsics can be classified into six major groups:

- Configuration of workspaces
- Configuration of windows
- Screen related functions like scrolling
- Display of characters
- Control of cursor movement
- Display memory partitioning, allocation and deallocation.

The keyboard intrinsics are completely table-driven, so that the various keyboard options like the Danish, German, Swedish, French, Spanish and Katakana keyboards can be supported in addition to the standard typewriter-style U.S. keyboard. The keyboard intrinsics isolate the hardware-dependent aspects of the keyboard from the logical operations and data needed by the main code.

The datacomm intrinsics provide the interface between the main code and the datacomm hardware. This is a character interface independent of the current protocol. That is, the main code does not have to know at any point whether the current datacomm configuration is point-to-point or multipoint. The datacomm code operates using a context area that is set up during terminal configuration. The point-to-point driver is completely reentrant so that the same code can be used for both ports even when one of the ports is configured as a printer port.

The print mechanism used in the 2626A has a dot interface, which requires that all character dot information be sent to the printer from the main code. The dots sent to the printer are the same character dots that are used by the video controller to display the data. Thus everything on the screen can be printed. The printer intrinsics control the printing of the characters as well as the position of the printhead. Thus some optimization is done to skip over blank characters and to print in both directions.

Hardware Design

To meet the goals of reliability, serviceability and ease of manufacture, the terminal electronics are on only three printed circuit boards: the power supply, the sweep, and terminal logic boards. This structure minimizes the number of interconnections, increasing reliability and making the terminal easy to build and service. Any problem in a terminal can be easily isolated and the faulty module quickly replaced.

The terminal logic board can be divided into four major sections: the microprocessor and its memory, the two datacomm interfaces, the keyboard scanner, and the display section. The microprocessor is a 16-bit HP CMOS SOS device chosen for its speed, low power dissipation, and input/output oriented instruction set. It is a modified MCC[1] called the MC5, which has TTL compatible inputs and outputs.

The program memory consists of ten 64K-bit ROMs (40K words) and 1K words of static RAM, which is used for the stack and for frequently used variables. To preserve the terminal configuration, a 256×4-bit CMOS RAM is included. It is powered by a battery when the terminal power is off.

The datacomm section consists of two nearly identical ports. Port 1 supports synchronous or asynchronous, full- or half-duplex, point-to-point or multipoint communications protocols. It also provides power to support a family of external datacomm pods (current loop, multipoint, and a 300 baud modem). Port 2 supports only asynchronous, full-duplex, point-to-point datacomm. Although Port 1 requires more control lines to support its additional protocols, both ports appear identical to the microprocessor so that a single firmware driver can be used to service both ports. Both ports use a universal synchronous asynchronous receiver/transmitter (USART) and appropriate interface chips to provide an RS-232-C interface.

To relieve the MC5 of the time-consuming task of scanning the keyboard, the keyboard is scanned by a single-chip microcomputer. It has 1K bytes of ROM, 64 bytes of RAM, 16 input/output lines and an eight-bit interface to the MC5. It is programmed to scan the keyboard, detecting and debouncing changes in the state of all keys. When it finds that a key has been pressed, it interrupts the MC5 and reports both the keycode and the state of the qualifier keys (shift and control). Each key can be programmed to be repeating, either slow or fast, or nonrepeating. Individual keys can also be locked out so they will not be reported to the main processor when pressed. The scanner provides N-key rollover, so overlapping keystrokes of a fast typist will not be lost. Besides scanning the keyboard, the scanner also drives a small speaker in the keyboard. The frequency and duration of the tone are programmable from the main processor.

Display Section

The display section is responsible for storing the data to be displayed, sending the necessary video, horizontal and vertical drive signals to the sweep to refresh the display, and supplying character dot information to the microprocessor for use by the optional thermal printer. A block diagram of the display section is shown in Fig. 1.

The display section is based on the CMOS SOS video control chip (VCC) described in the article on page 16. The VCC controls and refreshes the 16K×16-bit dynamic RAM, which is used to store data and enhancements. The character ROMs are 32K-bit CMOS SOS devices with an access time of 150 ns. Each pair of ROMs contains two character sets. The 2626A has the full 128-character ASCII set (upper- and lower-case Roman, numerals and control characters) and an extended line drawing set as its standard set. Optional sets include math and large characters.

The VCC reads its configuration information from the display memory and also reads a pointer list stored in the memory by the MC5. The pointer list points to rows of characters to be displayed. The VCC picks up a pointer and begins fetching characters and their enhancements at that address. As each word is read from display memory, the character and the character set select bits are used as an address for the character ROMs. The enhancements are read by the VCC, then the dots for the character are read from the character ROMs. The VCC modifies the dots with the enhancements and shifts them out to the sweep.

When the MC5 needs to access the display memory to update the pointer lists or to read or write data, it simply addresses the desired location as though it were normal RAM. The VCC holds off the MC5 until the next time the VCC does not need to use the display memory (usually during horizontal or vertical retrace). The VCC then disables its address bus and enables the MC5 to address the memory. The VCC then cycles the memory and returns a signal to the MC5 that the operation is complete.

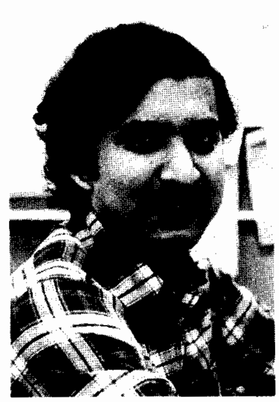
The terminal processor board is designed to be easily tested and repaired. An extensive self-test is run whenever the terminal is turned on. Any detected errors are displayed on the CRT. The error messages generally point the repair person to a particular socketed component that has failed. If replacing this component does not fix the problem, then the board is replaced with a new one.

Acknowledgments

Prem Kapoor was the project manager for the 2626A and provided a lot of guidance to the lab team. The datacomm intrinsics were written by Chris Vandever, Sara Hilbert and John Hill. Grant Head was responsible for the operating system. Brodie Keast and Bill Rytand helped him in designing it and wrote the printer intrinsics. We would also like to thank Janelle Bedke and Ed Tang for their contributions.

Reference

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



Srinivas Sukumar

Srinivas Sukumar received his BE degree in electrical engineering from Victoria Jubilee Technical Institute in Bombay in 1973 and his MS degree in electrical engineering from Washington State University in 1975. After two years of software design for medical information systems, he joined HP in 1977 and contributed to the firmware design of the 2626A Display Station. He's now a project manager with HP's Data Terminals Division. Born in Bombay, Srinivas is married and now lives in Sunnyvale, California. He enjoys racquetball, gardening, and table tennis.

John D. Wiese

Born in Norman, Oklahoma, John Wiese received his BSEE degree from Stanford University in 1969 and joined HP the same year. He's done hardware design for the 2570A Coupler/Controller and the 2313A Analog/Digital Subsystem, hardware and firmware design for the 2240A Measurement and Control Processor, and design of the processor board and self-test code for the 2626A Display Station. He's now a project manager with HP's Data Terminals Division. John enjoys volleyball, backpacking, and the study of wine. He's married, has two children, and lives in Palo Alto, California.

A Silicon-on-Sapphire Integrated Video Controller

by Jean-Claude Roy

THE 2626A DISPLAY STATION'S multifaceted personality is best seen through its display. The important display design objectives are very high character legibility and quality, hardware-supported windowing and blank filling, and horizontal and vertical scrolling. Other requirements are ease of manufacture, reliability, low cost, and low power consumption.

In support of these objectives a raster-scan display[1] operating at a 24.90-kHz line rate and a 25.7715-MHz dot rate was chosen. The character size is 7 dots by 11 scan lines inscribed within a 9-dot-by-15-line cell. Interstitial dots are fully encoded so that individual dots can be shifted one-half dot space for increased character resolution.

The screen format consists of 26 rows of 80 characters. The display may be configured so that any number of contiguous screen rows starting from the top constitute the upper screen, and the remainder the lower screen. This is easily done by having the 2626A firmware set a value for the demarcation row that indicates where the split is to occur. This allows flexible and efficient vertical screen partitioning into logical blocks. In normal operation the 2626A firmware reserves the upper 24 rows for the user and the bottom two rows for the softkey labels.

Horizontal screen splitting within either the upper screen or the lower screen can be done by configuring the display's upper and lower window screen count parameters. Using these and the demarcation row it is possible for the firmware to divide the screen into four independent windows.

To ease the task of discriminating between windows for the user, a programmable demarcation line can be turned on by firmware. This line coincides with the demarcation row, and its position, thickness, and intensity are controlled by means of appropriate configuration constants. Similarly, a right border may be turned on at the interface between the left and right windows. Finally, a bottom border may be activated on a row basis for each window to allow for further screen splitting in the vertical direction.

The two types of cursor supported by the display system are a blinking underscore and a full character cell rectangle. Cursor type, blinking, and underscore position within the cell are all selected by the firmware through cursor control words. Moving the cursor inhibits its blinking to make it visually easier to track for the terminal user.[2] This is done by internally comparing the old and new cursor positions for each frame and firing a blink-inhibiting digital one-shot multivibrator if they differ.

The display system supports ten combinations of the blink, underline, inverse video, half-bright, and security enhancements. In addition, a full-screen background setting capability allows the user to change the entire display from white-on-black to black-on-white or vice versa. Four character sets of 128 characters each can be accommodated. Finally, an internal dot and crosshatch pattern generator can be invoked to simplify terminal setup and alignment during manufacture. All of these features are available through configuration parameters and entries in the display memory.

To make this display system practical and reliable, it was felt that integration was mandatory. Since no commercially available CRT controller exists with this flexibility and large repertoire of features, a new video control chip (VCC) was developed. All the circuits required to support the 2626A's display system are included in the chip with the exception of a crystal oscillator, character ROMs, display

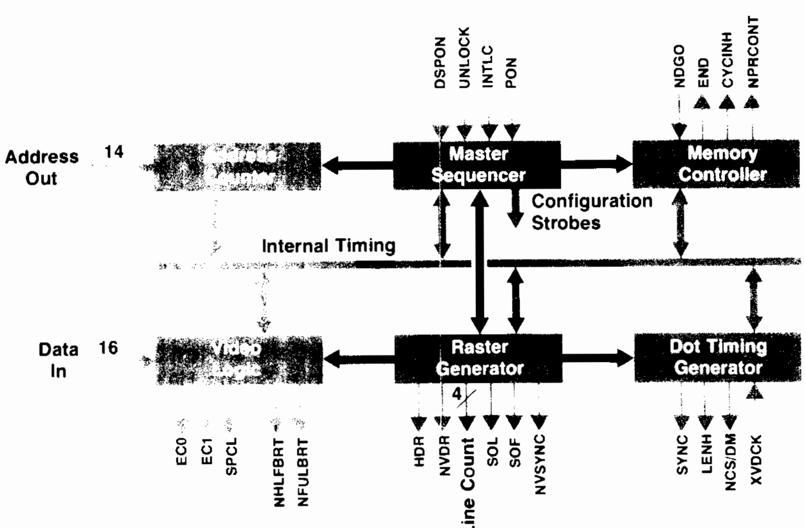


Fig. 1. *Block diagram of the video control chip (VCC). Several interlocked state machines and counters are driven by parameters read from the display memory (black lines are control paths).*

memory, sweep circuits, a delay line, and a bus switch.

The reasons for integration are compelling. Given the complexity of this chip, the equivalent in TTL logic would have required approximately 220 integrated circuit packages and used 775 cm² of six-layer printed circuit board. This logic would have dissipated approximately 15 watts. Custom integration resulted in one component that requires about 29 cm² of board area and dissipates only 900 mW.

The decision to implement the VCC in HP's silicon-on-sapphire (SOS) process[3] was based on SOS's high speed, low power, static operation, and reasonable density.

Video Control Chip

The VCC has a flexible structure configurable by firmware to allow a variety of screen formats. Thus it may be used not only in the 2626A but also in other terminal products without any redesign. It supports character enhancement and all dot processing, and has its own built-in DMA capability.

The VCC is implemented as a hierarchy of interlocked state machines and counters driven by configuration and control parameters read from the display memory. A simplified block diagram of the VCC is shown in Fig. 1.

The master sequencer, a complex state machine, controls the overall operation of the VCC. The algorithm of the master sequencer is driven both by qualifiers from the raster generator representing the current state of the screen and by configuration information from the firmware. The master sequencer controls the address counter, the vertical window counter, and the memory controller. At chip turn-on it reads the hard configuration area of the display memory and loads the values into the chip's registers. These represent raster-defining parameters such as the number of screen rows and columns (see Table 1). During on-screen activity the master sequencer coordinates the fetching of pointers and characters from the display memory and during horizontal retrace it controls memory refreshing. Once per frame it goes through a soft configuration housekeeping cycle and reads such dynamic information as the current cursor position and the vertical window width.

The address counter is 14 bits wide, giving it an addressing range of 16K words. Under master sequencer control it performs the operations of row pointer and character fetching, display memory refreshing, and the addressing of the hard and soft configuration memory areas.

The memory controller arbitrates processor access of the shared display memory. In normal operation the VCC locks out the processor during on-screen columns and allows access only during horizontal and vertical retrace. Under processor command the VCC may allow itself to be preempted by the processor, which causes screen blanking. One use of this mechanism is the interleaving of VCC and processor accesses. Another use, helpful for whole-screen updates, involves blanking the entire frame from the time the processor desires the memory until the start of the frame following the release of the memory.

The video logic consists of separate dot and enhancement processors. These accept the character enhancement information read from the display memory and the character dot information read from the character ROMs. Together these processors generate the two video signals representing full-bright and half-bright bit streams, which go directly to the sweep assembly.

The dot timing generator is driven directly by the 25.7715-MHz dot rate clock. It provides all of the VCC's internal timing including the high-speed clocking required by the video logic's parallel-to-serial converters. It also

Table I

Hexadecimal Address		
		Hard Configuration Constants
0	VHRC	Variable Horizontal Retrace Columns
1	AC	Active Columns
2	CHDR	Clear Horizontal Drive Signal
3	SHDR	Set Horizontal Drive Signal
4	(0)	(not used)
5	SR	Number of Visible Screen Rows
6	(0)	(not used)
7	(0)	(not used)
8	(0)	(not used)
9	SSL	Starter Scan Lines
A	(0)	(reversed for testing)
B	(0)	(reserved for testing)
C	RL	Scan Lines per Row
		Soft Configuration Constants
10	CROW	Cursor Row
11	CCOL	Cursor Column
12	CC	Cursor Control
13	DC	Display Control
14	DLC	Demarkation Line Control
15	SLD	Demarkation Line Starting Scan Line
16	SLDR	Demarkation Row Scan Lines
17	DR	Demarkation Row
18	ESL	End Scan Lines
19	VR	Vertical Retrace Lines
1A	LSEA	Lower Screen Enhancement Assignment
1B	USEA	Upper Screen Enhancement Assignment
1C	LSHM	Lower Screen Hardware Mask
1D	USHM	Upper Screen Hardware Mask
1E	LSVW	Lower Screen Vertical Window
1F	USVW	Upper Screen Vertical Window
20-22	(0)	(not used)
23	(0)	(reserved for testing)
24	(0)	(reserved for testing)

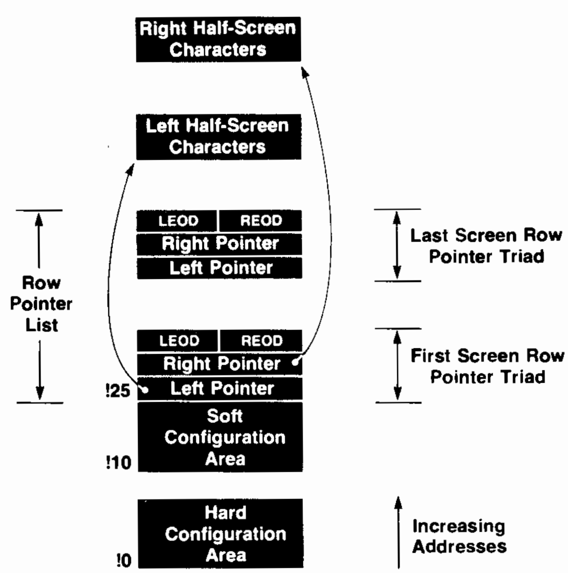


Fig. 2. *The display memory contains two configuration areas, a pointer list, and the rows of characters to be displayed.*

supplies the external timing needed for the display memory and the character ROMs. The 2626A uses an external delay line driven by a synchronization signal to generate the memory RAS and CAS signals (row and column address strobe).

The raster generator consists of the column, line, row, and vertical window counters. These configurable counters represent much of the VCC's flexiblity, allowing the chip to be firmware-configured for a variety of applications. The specific configuration required is loaded into the raster generator during the hard and soft configuration housekeeping cycles. By way of example, two screen configurations are used in the 2626A to select between the 60-Hz United States frame rate and the 50-Hz European rate. By judiciously selecting the appropriate line counts for vertical retrace and by padding the frame with dummy scan lines it is possible to select either rate and still maintain both constant vertical centering and a constant horizontal scan rate. Frame rate selection is seen by the user as a choice between 50 Hz and 60 Hz. When a change is made to the terminal configuration, the firmware modifies two constants in the soft configuration table. Beginning on the next frame, the raster generator responds to the new values by operating at the new frame rate. This type of flexibility totally eliminates the need for the usual plethora of switches and jumpers.

Display Memory Organization

The display memory may be either 4K or 16K words by 16 bits. It is implemented in the 2626A by means of industry standard dynamic RAMs. It could also be implemented in static RAM, PROM, or ROM, provided that the system timing requirements are met. Dynamic memories represent the lowest system cost for full-screen displays. However, these memories must be periodically refreshed to preserve their contents. For this reason the VCC performs explicit refresh cycles as part of its horizontal retrace activities. Once per scan line four successive columns are cycled. An internal seven-bit refresh register is kept as part of the address counter for this purpose.

Access to the display memory for refreshing the CRT is done on a real-time basis with no line buffering. The processor is allocated access during both horizontal and vertical retrace and may elect to lock out the video system for large block transfers. This represents a reasonable compromise between memory cost, bandwidth, and memory availability.

The display memory contains the two configuration areas, the pointer list, and the rows of characters to be displayed. Fig. 2 shows the data structure used. At the bottom starting at address 0 is the hard configuration area. This is read once at chip turn-on and is never accessed again. The soft configuration area is located from address 10_{16} to address 24_{16} and is read once per frame to update the dynamic screen parameters.

The pointer list starts at location 25_{16} and contains three words per configured screen row. These are a pointer to the left window characters, a pointer to the right window characters, and two concatenated end-of-data bytes. These bytes represent the number of characters to display in the row's left and right window areas, respectively, before starting hardware blank filling.

The pointers may be manipulated by the firmware to implement scrolling, row insert, and other functions. Each pointer can address the entire 16K range of the display memory. It is the responsibility of the firmware to build a valid and consistent data structure since the VCC cannot write to the display memory nor does it know how big it really is.

The character entries in display memory each consist of seven bits of ASCII representing the character itself, four bits of character enhancement, and two bits for character set selection. The remaining three bits are reserved by the firmware for 2626A features. Mask registers in the VCC may be set to disable the enhancement and character set select bits on a half-screen basis. These are used if a chip's application requires the bits to be reassigned to a different purpose.

The logical length of a row of characters may be different from its physical length on the screen. Portions of longer rows will be off-screen until they are scrolled into view. This is accomplished in firmware by altering the row's pointer to redefine what is to be displayed in the window. Shorter rows may be blank-filled by the firmware at the expense of display memory. Alternatively, the firmware could use the row's end-of-data byte, which indicates how much of the row to display, with the remainder to be hardware blanked.

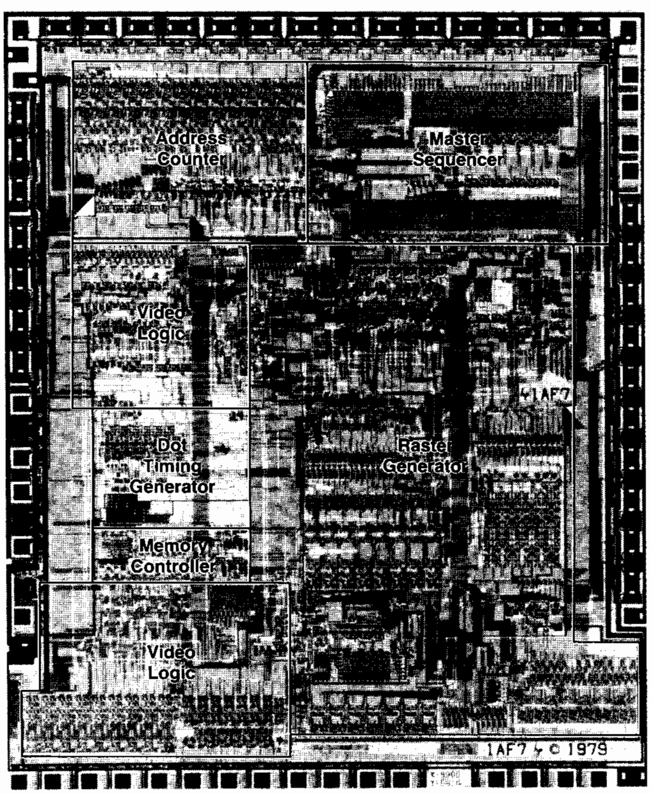


Fig. 3. *Photomicrograph of the video control chip.*

VCC Implementation

Fig. 3 shows a micrograph of the VCC with the principal blocks labeled. The chip's logic is designed for maximum speed in the critical circuits while minimizing area (the die is 4.9 mm by 5.8 mm). The configuration constants are stored in latches connected to an internal 16-bit data bus and clocked by the master sequencer. The same data bus is used as a stimulus bus when the chip is tested at wafer sort and package sort.

The internal state machines are implemented using D-type flip-flops and PLAs (programmable logic arrays). The chip is entirely static, a feature of CMOS/SOS that simultaneously allows low power dissipation and high speed. Static operation makes testing of this chip relatively easy. A circuit may be set to a state and left there for an arbitrary amount of time without loss of information.

Physical Description

The VCC is packaged on a 64-pin leadless square ceramic substrate measuring 4.4 cm by 4.7 cm (Fig. 4). This package fits into a special zero-insertion-force socket and is held in place with spring clips. Heat dissipation is through the substrate and a temperature rise of 10°C is typical at the 900 mW power level at which it operates.

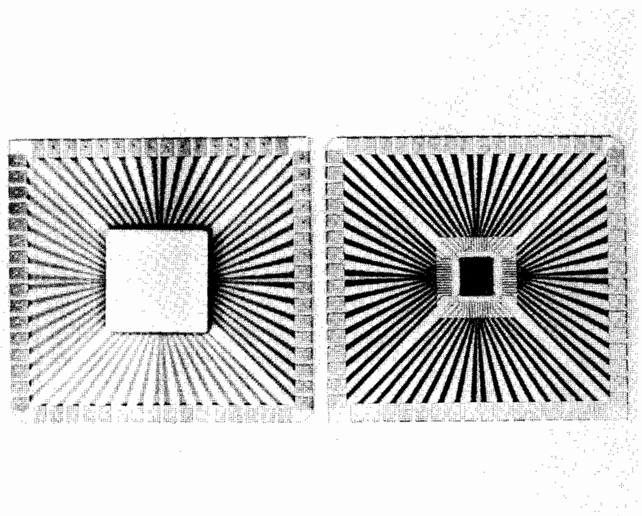


Fig. 4. *The VCC is packaged on a 64-pin leadless ceramic substrate measuring 44 mm by 47 mm.*

Acknowledgments

No LSI chip design can be done without the help of large numbers of people and organizational entities. My thanks to all of them. Special thanks go to David Parks (now at HP's Vancouver Division) for his design contributions and to Stan Moriya for developing the test and characterization programs. Mention must also be made of Al Desroches and Frank Wilson at the Cupertino Integrated Circuits Operation for all their help in getting the VCC into production.

References

1. J.C. Roy, "A High-Resolution Raster Scan Display," Hewlett-Packard Journal, June 1975.
2. J.C. Roy, "Display Apparatus Having Improved Cursor Enhancement," United States Patent No. 3,967,266, June 29, 1976.
3. B. Forbes, "Silicon-on-Sapphire Technology Produces High-Speed Single-Chip Processor," Hewlett-Packard Journal, April 1977.

Jean-Claude Roy

A native of Montreal, Canada, Jean Roy did his undergraduate work at the University of California at Davis, graduating in 1970 with a BSEE degree. He received his MSEE degree from Stanford University in 1974 and his MSCS degree from the University of Santa Clara in 1979. With HP since 1970, he's been responsible for software design for the 5407A Scintigraphic Data Analyzer, for the organization and logic design of the 2640A Terminal's display, and for the definition, design, and project leadership of the video control chip for the 2626A Display Station. Now a project manager with HP's Data Terminals Division, he's a member of the IEEE and a registered professional engineer in California. Jean is married and lives in San Jose, California. He's an avid reader and is interested in art, gourmet cooking, real estate, and raising dogs.