AN EXPERIMENTAL PLASMA DISPLAY TERMINAL

J.W. BRAHAN, A.M. HLADY, J. HUMPHRIES

NATIONAL RESEARCH COUNCIL OF CANADA

ABSTRACT

A computer display terminal has been developed at the National Research Council as part of our research effort in Computer-Aided Learning. Some of the various display and special device control features of the terminal are a vector graphics capability, interchangeable symbol sets that can be loaded and invoked from the host computer, and an ability to superimpose continuous tone images on computer generated information.

The terminal currently uses a 512x512 plasma panel as the display medium and a TV compatible display is being developed as an alternative. The control logic has been implemented through a software program in a dedicated mini-computer rather than with a hard-wired controller in order to provide flexibility for design improvements.

Conformity to applicable standards has been given a high priority in the development of this terminal. Our attempt to select protocols and coding that agree with published standards and recommendations has provided considerable insight into the extent to which available standards apply to display terminals.

RÉSUMÉ

Un terminal d'affichage graphique a été développé au Conseil national de recherches dans le cadre d'un programme de recherches sur l'enseignement à l'aide d'ordinateurs. Ce terminal se caracterise par les moyens d'affichage et de contrôle des dispositifs spéciaux comme l'affichage des vectors, jeux de caractères qui peuvent être transmis et invoqués de l'ordinateur central, et la capacité de présenter les images de diapositives en les mêlant aux données provenant de l'ordinateur.

Le terminal utilise actuellement un écran à plasma, avec une résolution de 512 par 512 points, comme unité d'affichage, mais un écran qui va de pair avec la télévision est en train d'être dévelopé comme alternative. La logique de contrôle est comprise dans le logiciel d'un "mini-ordinateur" au lieu de circuits fixes. Ce dessin offre plus de flexibilité afin d'améliorer les caracteristiques du terminal.

On a accordé aux normes une priorité importante dans le développement de ce terminal. Dans les efforts de choisir les protocoles et les codes en accord avec les normes publiées, on a obtenu une connaissance de l'efficacité des normes déjà disponibles pour les terminaux d'affichage graphique.

AN EXPERIMENTAL PLASMA DISPLAY TERMINAL

Introduction

The development of an experimental plasma display terminal is being undertaken as part of the National Research Council's Computer-Aided Learning (CAL) project which involves the cooperation of a number of educational organizations throughout Canada(1). The role of NRC in this project is focussed on the development of system software, authoring language, and special purpose terminal equipment of which the plasma display terminal is one example.

Of particular importance in any CAL system is the terminal which provides the interface between the student and the course materials. While in some cases the conventional "typewriter-like" computer terminal may be satisfactory, in general it imposes undesirable restrictions on the presentation of information and on the form of response by the student. The characteristics of the "ideal" student terminal are determined by the subject matter and also by the capabilities of the student. Thus the features which have been incorporated in the plasma display terminal as requirements for a CAL student terminal will have more or less importance, according to the specific application being considered.

The basic feature of the terminal is a conventional alphanumeric display capability, enhanced by the capability of varying the size of the displayed characters and also the capability of redefining the symbols of the character set. To this is added a graphic display capability which permits the presentation of relatively high resolution line drawings on which photographic images can be superimposed. The presentation of audio messages under computer control is a requirement of many CAL systems and provision has been incorporated in the terminal for control of a speech synthesizer or audio disc in addition to other special devices.

Input facilities include an alphanumeric keyboard and a graphic input device. In addition, provision has been made for the connection of special input devices which may be required for particular applications of the terminal.

The development of the terminal was undertaken with the objective of providing a unit enriched with as many of the desired features of a student terminal as possible so that they might be evaluated in a variety of CAL applications. At the same time, it was recognized that the terminal design had to be sufficiently flexible to permit modifying features as required by experimental results. To provide the necessary flexibility, the design was implemented using a mini-computer to provide the basic logic control, with the characteristics of the terminal being defined by software. The configuration is illustrated in Figure 1.

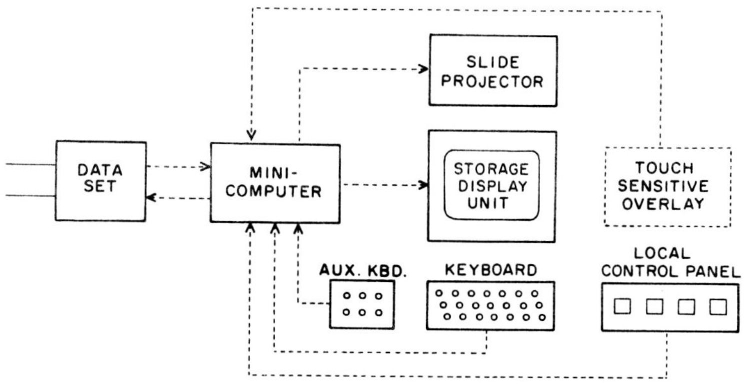


Figure 1 Terminal Configuration

Conformity to Standards

Several CAL projects have developed highly specialized terminals tailored for use with one computer system. This approach was rejected in the NRC-CAL project on the basis that if a CAL terminal is to be effective, it must be capable of being used with a variety of computer systems and, furthermore, significant economic savings are to be achieved if applicable features of conventional terminals can be exploited. For these reasons, adherence to standards published by the Canadian Standards Association (CSA), the International Organization for Standardization (ISO) and other standards-writing bodies has been given high priority in the terminal design.

The basic character set has been implemented in accordance with the CSA standard Z243.4(2), which provides for the presentation of both English and French text. Invocation of special character sets is in accord with the procedures recommended by ISO for code extension(3). Control codes for use with the terminal have been derived from these two standards.

It is to be noted that the published standards invariably lag behind the state-of-the-art and there are a number of instances where they do not provide adequately for the control procedures required in the terminal. In these cases, an attempt has been made to keep within the spirit of the basic recommended standard and proposals are being made to the appropriate CSA committees for extension of the standard to accommodate the new requirements.

Display Characteristics

The plasma panel being used is a Digivue unit produced by Owens-Illinois Inc. It contains 512x512 digitally addressable points on a screen 8½ inches square, and is characterized by an orange, fixed brightness, flicker-free display. Display storage with selective erase of individual dots is an inherent feature of the panel so that no refresh memory is required. The location of each dot is physically fixed on an orthogonal grid, which eliminates problems such as drift and edge distortion.

The display screen is transparent, flat and only ½ inch thick; thus it is possible to superimpose photographic images on the same area by rear projection. A GAF random access slide projector with 100 slide capacity is included in the terminal. The parallax caused by the

$\frac{1}{4}$ inch separation between the projection screen and the computer generated display is not objectionable. Balancing the relative intensities of the two displays, however, has been a problem. A 0.70 neutral density filter installed in front of the projection lens appears to be a good compromise but the computer display still tends to be washed out when superimposed on light-coloured portions of a slide image.

At present, a major disadvantage of the plasma panel is high cost. A TV compatible video display unit is being developed as an alternative. This unit includes internal memory and will be able to use the existing terminal control program with an addition for video image control. An earlier experiment using a storage tube was unsuccessful because of the poor selective erase capability.

A display unit based on a storage screen with selective erase possesses certain characteristics in common with normal volatile CRT displays and others in common with conventional storage tube displays. For example, features such as flashing symbols or roll mode which are relatively easy to implement on normal CRTs are impractical on a plasma panel. On the other hand, an overprinting capability is inherently available on all storage screens but requires extra circuitry to achieve on CRT displays. This overprinting feature is useful under certain circumstances such as underlining text or creating special symbols but is not desirable in most cases. Because of this, the normal procedure in the experimental terminal is to erase a symbol space before writing a symbol into it.

Another factor that must be considered is cursor generation. The plasma panel does not have a non-storing display mode. The cursor on this terminal is a flashing vertical line with the flashing being accomplished by alternately writing and erasing the cursor under control of a real-time clock in the local computer. When the cursor is moved, it must be erased from its previous position. Although this action does not interfere with standard alphanumeric symbols, it causes a problem when moving the cursor over any special symbols that use the full width of the symbol space. To overcome this problem, a cursor on/off control has been provided.

Character Display

The screen capacity of 64 columns by 28 rows was dictated by the choice of a symbol space grid of 8x18 display elements. Each alphanumeric symbol is formed on a 7x10 grid instead of the more usual 5x7 or 7x9 in order to produce a set of 94 alphanumeric characters, including accented letters, of sufficient quality for CAL applications. The 18 dot height of the symbol space permits super/sub-scripting without interference. The basic alphanumeric symbol set is locally switchable to be either of the two sets specified in CSA Z243.4, which provide either ASCII compatibility or bilingual capability. A full page of text appears in Figure 2.

One of the most powerful features of this terminal is ability to display user-defined symbol sets. In a CAL environment, an author will be able to select from a "symbol set library" or create a set to suit his own requirements. When a particular set is to be used, it can be loaded into the terminal from the host computer under program control.

Examples of symbol sets are alphabets for other languages, mathematical symbols, electronic component symbols, or pictorial symbols. Each alternate set can contain from 1 to 94 symbols using octal codes 041 to 176. Any of the 94 codes not re-assigned in a given set revert to the basic alphanumeric symbols by default.

A size control feature in the terminal permits any of the symbols, either alphanumeric or user-defined, to be displayed at 1, 2, 4, or 8 times their basic size under program control. When the size is changed, the same scale factor is automatically applied to all cursor movement commands.

All symbols are represented within the terminal as a sequence of vectors rather than as a dot matrix. Each vector or point in a symbol occupies an 8 bit byte and is described in terms of its X and Y components with respect to the lower left corner of an 8x18 or 16x18 symbol space. This representation occupies about the same amount of space as an equivalent dot matrix description and the entire 94 character basic set requires less than 500 words of local storage. The vector representation has the advantage of improved appearance of the symbols when they are enlarged. Figure 3 illustrates the result on some special symbols.

The basic alphanumeric set is the only symbol set that is prestored in the terminal. Any other sets must be loaded into the terminal from the host computer.

The procedure being used to transmit symbol sets is based on ISO Standard 2111(4), which covers code independent information transfer within a normal code environment. The Data Link Escape control character (octal code 020) is used for this purpose. Any block of code independent information can be transmitted as a sequence of 8 bit bytes provided it is preceded by a DLE.STX control sequence and terminated with a DLE.ETB. If any 8 bit group contains a bit combination equal to the DLE character, an additional DLE must be inserted. ISO 2111 also permits an identifying header and the following format has been chosen:

DLE STX	$C_1C_2C_3$	XX XX	DLE ETB

where C_1 is a data classifier, and C_2C_3 is a data identifier. For symbol set transmission, C_1 equals 1 and C_2C_3contain a symbol set number which is subsequently used to invoke that set. The number of symbol sets that can reside in the terminal simultaneously depends upon the memory space available. The terminal can presently hold 3 sets in addition to the basic alphanumeric set.

After a symbol set has been loaded into the terminal, it must be invoked by the appropriate control code sequence before it can be used. Here again, standard procedures have been followed. ISO Standard 2022 (3) describes a method of symbol set extension that is represented pictorially in Figure 4. The symbol set repertoire consists of all the sets currently stored in the terminal.

The Shift Out (octal 016) and Shift In (017) control characters are used to invoke a G1 set and G0 set respectively. Initially, the G0 set is in effect. In the experimental terminal, the initial or default G0 set is the basic alphanumeric set, identified as set 0. As will be explained later, the initial G1 set is actually graphics mode, which is identified internally as set 1.

In order to use one of the other sets in the repertoire, it must first be designated as a new G1 or G0 set by means of a 3 character Escape sequence. The last character of this sequence corresponds to the data identifier that is transmitted with a symbol set at load time. After a symbol set has been invoked, any of the printing character codes (octal 041 to 176) results in the appearance of the appropriate symbol from that set.

Graphics Mode

A graphics facility is available in the terminal to plot points and draw straight line segments on an XY coordinate grid. X-Y positions are specified and stored as 10 bit coordinates, although only the lower 9 bits are used on the plasma panel.

Graphics mode is accessed in the same way as a special character set. Internally identified as character set 1, it is considered as the default G1 character set and, therefore, is normally accessed by a single Shift Out control code. Upon entering graphics mode, the cursor changes to a single flashing dot. Treating graphics mode in this way goes beyond the scope of the code extension technique described in ISO 2022 because there no longer is a one-to-one correspondence between incoming characters and the displayed results. Unfortunately, there is no obvious alternative because the whole area of graphics displays is not covered by existing standards.

The representation of XY positions for transmission is another area which is not standardized. The experimental terminal uses a method similar to that found in Tektronix terminals. An XY position is represented by a 4 character sequence: $X_H X_L Y_H Y_L$, where X_H, Y_H contain the 5 most significant bits of each coordinate, and X_L, Y_L contain the 5 least significant bits. X_H, Y_H are transmitted as octal codes 060 to 117 and X_L, Y_L as 120 to 157. This encoding specifically avoids the use of codes 040 and 177 because, according to ISO 2022, they have fixed meanings regardless of the symbol set in effect.

The characters representing X_L and Y_L are sent for every position but X_H and Y_H do not have to be sent unless they have changed from the previous position or they are part of an initial position. Therefore, an XY position is transmitted as 2, 3, or 4 characters.

The principal graphics mode is a vector drawing mode in which a straight line is drawn to each received position from the previous position. There is provision within this mode for generating starting points easily. A point plotting mode is also available in which every position results in a single point being plotted.

In addition to the normal solid line segments, it is possible to display up to nine different line types made of dots or dashes, as illustrated in Figure 5. There is also a capability in the terminal for doing a limited amount of graphics composing using the alphanumeric keyboard. One can position the flashing dot cursor by using the regular cursor positioning keys and/or by typing in the XY coordinates of the desired position. In graphics mode, use of the Delete key cancels the most recent line or point.

Other Terminal Features

Symbol highlighting is accomplished by reverse contrast, producing dark symbols on a bright background. The overwrite mode makes it also possible to emphasize information by underlining. Blinking or intensity variation are not practical methods of highlighting on the plasma panel.

Another terminal feature is a mode that permits all incoming control characters to be displayed as symbols instead of being executed. This has proven useful in debugging programs that generate special terminal control codes. For example, it could be used to examine the output of a program in the host computer for generating symbol sets. Any control code (octal 0 to 037) is displayed as an alphabetic with a double underscore.

The display cursor can be positioned anywhere on the screen, either in terms of columns/rows or as an XY position. There is also a control code that causes the terminal to report the current cursor position as an XY position.

Terminal Control Program

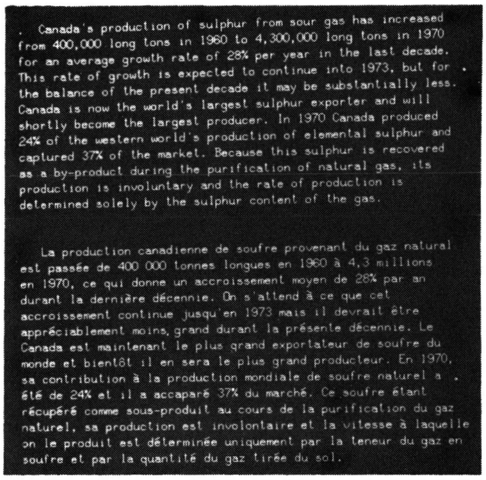
The terminal control program, written in assembly language for the Alpha 16 mini-computer, consists of just over 2000 lines of code. After including the dual version basic symbol set, a 300 character input buffer, and a small editing buffer, the space remaining in a 4K memory is sufficient for 3 extra symbol sets.

The basic display module is the vector generator which provides the sequence of dots that best fit a straight line between the two end points. In addition to its application in graphics mode, it is used by the symbol generator to produce each line segment in a symbol. For this purpose, the vector generator contains a routine that makes line width proportional to the current size setting.

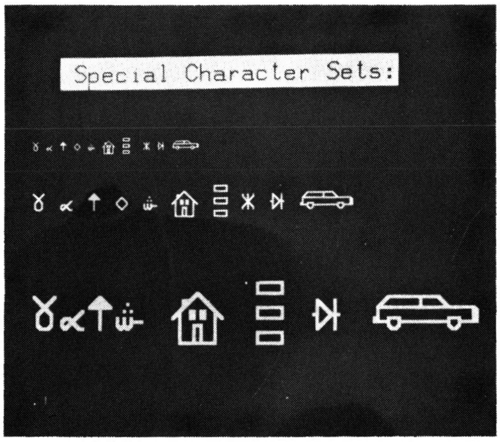
The symbol generator module is used to produce all symbols, including the control character symbols and the cursor symbol. Any size change therefore, will affect all symbols in the same way. The symbol generator is designed to handle double width symbol patterns occupying a 16x18 grid space. When this type of symbol is encountered in the display, the cursor is automatically incremented by the correct amount, so that no special user action is required.

The plasma panel can be driven at a rate of 20 usec. per point and the Alpha 16 is able to perform most memory reference instructions in 3.2 usec. These speeds have turned out to be well matched to each other and also to the 1200 baud asynchronous communication line that is currently being used. The maximum display rate of the terminal has been measured as 160, 50 and 15 characters/sec. for alphanumerics of size 1, 2, and 4 respectively. In overwrite mode, which eliminates the erasing step, the rates are increased to 210, 95, and 45 characters/sec. These measurements indicate that, although the communication line limits the rate for basic size symbols, the terminal is the limiting factor for larger symbols.

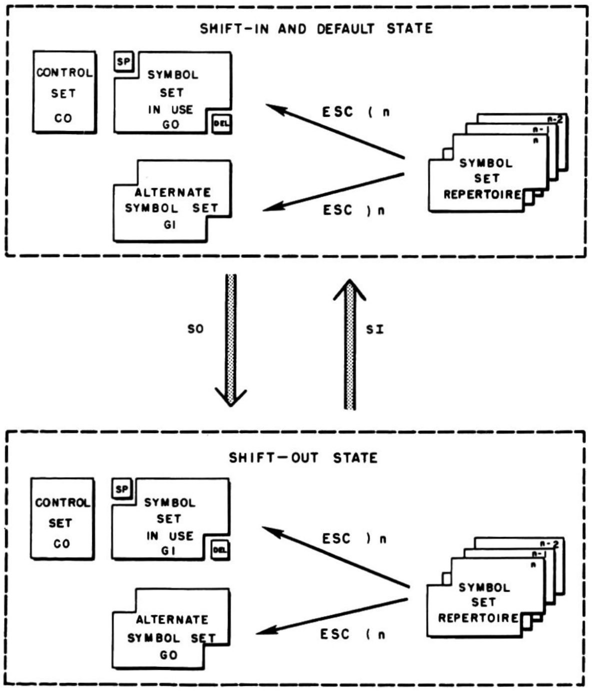
In graphics mode, the terminal has a maximum vector drawing rate of about 225 inches/sec. Although the 1200 baud line could sustain a drawing rate of 360 inches/sec. for corner to corner vectors, graphics mode is actually limited by the line speed in most applications.



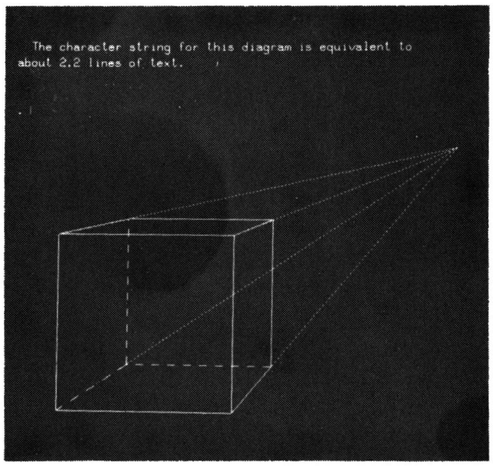
Bilingual Text
Figure 2



Variable Size Symbols
Figure 3



Symbol Set Extension
Figure 4



Graphics Mode
Figure 5

Conclusions

It is difficult to determine when an experimental facility of this type is completed. Other features can be added for the cost of the programming time to implement them plus the memory space to store the routines. In many cases, such extensions can be built upon existing routines. Examples of possible additions include more extensive error checking, a circular arc generator, an incremental graphics mode, a bit mapping mode, variable symbol orientation, and offset registers that would permit "windowing" on a larger plotting space.

Supporting software in the host computer is still inadequate for procedures such as designing special symbol sets or manipulating graphics files. A new CAL authoring language, NATAL-74, now being implemented, used in conjunction with a planned author support package, will permit course writers to specify and invoke symbol sets, diagrams or graphics, and handle other terminal devices such as the slide projector or touch sensitive overlay, all in a high level language. It is hoped that this facility will permit non-technical users to make effective use of sophisticated terminal features.

The choice of a general purpose mini-computer for terminal implementation is based on a requirement for flexibility and is not necessarily optimum for a production model. The approach has proven to be a useful experimental technique for terminal development.

References

(1) Brahan, J.W., Brown, W.C., "The Network Approach to Cooperative Development of Computer-Aided Learning Systems", Programmed Learning, Sept. 1973. NRCC 13481

(2) 7-Bit Coded Character Sets for Information Processing Interchange. CSA Standard Z243.4 - 1973.

(3) Code Extension Techniques for use with the ISO 7-bit coded character set. International Standard ISO 2022 - 1973.

(4) Data communication--Basic mode control procedures--Code independent information transfer. International Standard ISO 2111 - 1972.