SA#2260FP

CONF-810805--4

# TOUCH-SENSITIVE GRAPHICS TERMINAL APPLIED TO PROCESS CONTROL

S. I. Bennion and J. D. Creager
Westinghouse Hanford Company

R. D. VanHouten
Boeing Computer Services, Richland

Association for Computing Machinery
SIGGRAPH '81
August 3-7, 1981, Dallas, Texas

**HANFORD ENGINEERING DEVELOPMENT LABORATORY**
**Operated by Westinghouse Hanford Company, a subsidiary of Westinghouse Electric Corporation, under the Department of Energy Contract No. EY-76-C-14-2170**

# 1. Introduction

An automated fuel fabrication capability for fast breeder reactor fuel is being developed at the Hanford Engineering Development Laboratory (HEDL) in Richland, Washington on behalf of the U.S. Department of Energy. The Secure Automated Fabrication (SAF) line incorporates a centralized computer system linked to a distributed process control system as shown in Figure 1.

The Process Instumentation and Control System (PICS) is comprised of dual minicomputers with supporting mass storage and additional dual minicomputers acting as communications processors. These computers, as well as the unit operation dedicated controllers, are Hewlett Packard HP-1000 minicomputers.

The fabrication line is being developed such that it can be operated remotely by a small staff located in the control center. To support this goal, the control center makes extensive use of several integrated control stations consisting of color graphics terminals with touch sensitive screens (touchscreens) applied over the Cathode Ray Tube (CRT) face.

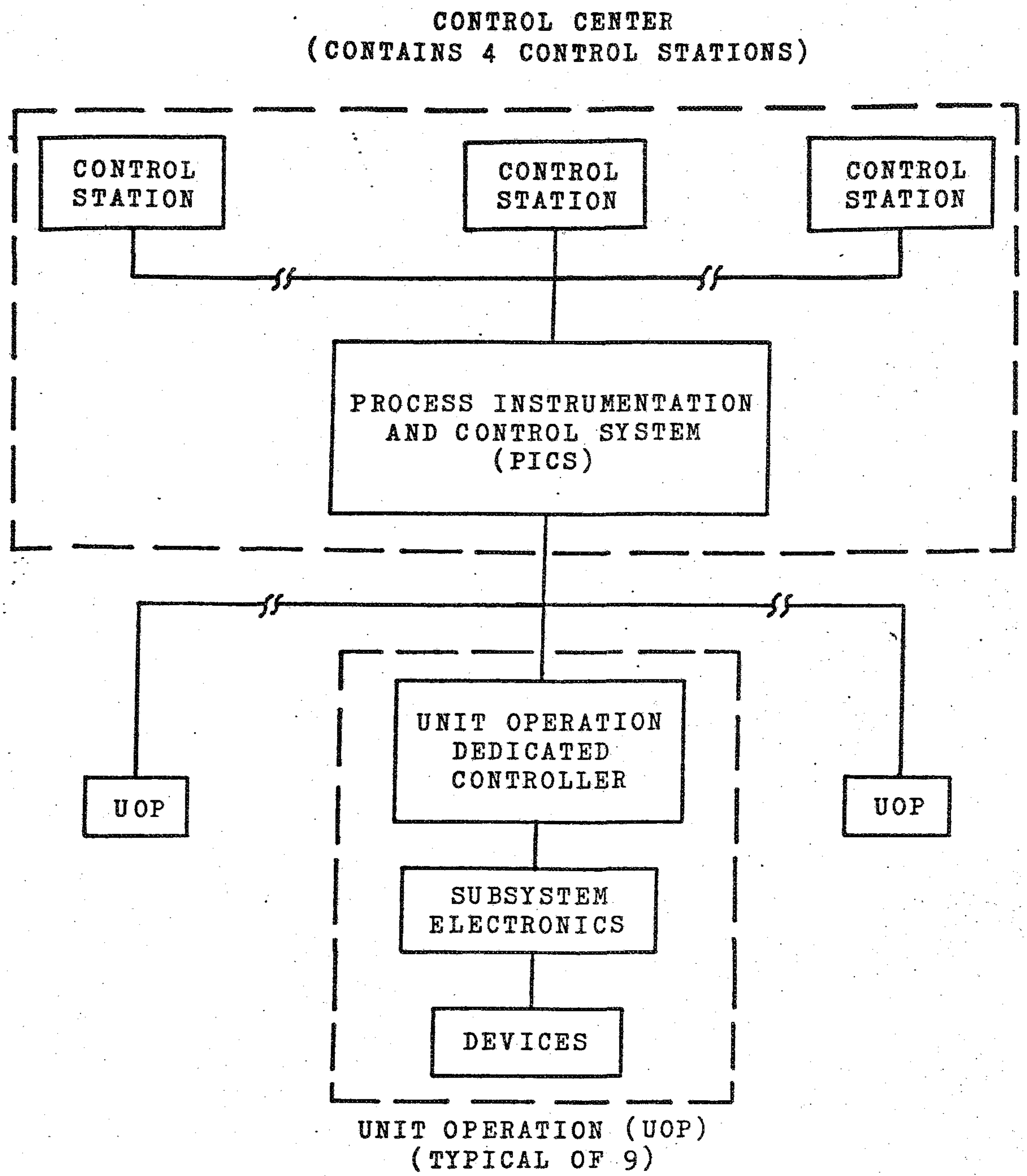


Fig. 1. SAF control system block diagram

## 2. Interactive Color Graphics in Process Control

A computer-based process control application, such as SAF, benefits greatly from a friendly interface for the operator who ultimately controls the activity of the system. Graphics displays allow the operator to selectively view all required process data in varying degrees of detail at a compact control station. Instead of having a control console consisting of many dedicated signal indicators laid out in a large area, he receives the information he needs in the form of graphic displays on color CRT's. The use of color graphics provides increased flexibility for the system engineer responsible for designing and implementing a data display and process control scheme. Complex data relationships are communicated efficiently and clearly with graphical displays, while the use of color significantly increases the density of information which can be displayed and comprehended by the operator.

Most control schemes based on interactive graphics suffer from one serious drawback: the operator is required to communicate with the system through a keyboard entry scheme. During system transients or off-normal conditions, when their skills and judgement are most severely taxed, they are required to divert their attention from the process display and concentrate on manual command entry functions.

Most implementations make an effort to alleviate this problem by using special function keys for common command entries. While these efforts have had some success, they do not address the root of the problem. The approach to this problem being taken at HEDL is to minimize the need for a keyboard by integrating touchscreens with the display devices. This allows the operator to direct his attention at the process display instead of alternating between the display and the keyboard. When control actions are necessary, they are initiated by simply touching an appropriate location on the process display. This technique is more natural to an operator than the other common methods of interacting with graphic displays, such as trackballs or joysticks.

## 2.1 Touchscreens

Several different technologies exist for applying touchscreens to display terminals. The sensor arrangements fall into one of three broad classifications: (1) Sensors which detect the interruption of a reference signal; (2) Sensors which detect the reflection of a reference signal; or (3) Sensors which directly sense the presence of the object touching the screen.

A signal interruption device typically consists of a matrix of light-emitting diodes and photo-detectors mounted parallel with the CRT face. When an object touching the screen interrupts these beams, the coordinates are determined by the loss of signal on one or more of the photo-detectors. This type of device is generally of relatively low cost, mounts easily, and has low resolution. If the diodes and detectors operate in the infra-red region, the light beams do not degrade the display contrast. One serious drawback in a developmental application is that the resolution is not adjustable. Further, since the sensing elements are typically mounted some distance from the screen to allow for CRT curvature, parallax errors become a significant problem.

A signal reflection device typically consists of a pair of signal sources and detectors mounted on the edge of the CRT screen. When an object touches the screen, some of the energy in the reference signal is reflected back to the detectors. By measuring the time duration between the initial and reflected pulses, the location of the object can be determined quite accurately. These devices are generally expensive since they must be fabricated for a specific CRT geometry. They offer high resolution, are adjustable and do not degrade the display.

A direct detection device typically consists of a transparent overlay on the CRT face which senses the presence of an object. Such a device is used in this application. The overlay consists of two mylar sheets placed one atop the other. The sheet nearest the CRT face supports a resistive layer terminated at connectors on each of the four edges. The outer sheet supports a conductive layer mounted so that it is not normally in contact with the resistive layer. When the overlay is touched it causes contact to be made between the resistive and conductive layers, forming a voltage divider network. Voltage pulses, alternately applied to opposing sides of the overlay generate linear voltage gradients on the overlay. When the overlay is touched, voltages representative of the touch location are picked off by the conductive layer.

This type of touchscreen is generally of moderate cost, simple to mount and has adequate, adjustable resolution. It does interfere with the light transmission from the terminal to some extent, but our experience has been that this is not a significant problem.

## 2.2 Previous Work

An earlier effort at HEDL demonstrated the usefulness of touchscreens and color graphics terminals in support of process control. A simulation of a two-loop heat transport system was controlled from a graphics terminal with a touchscreen. The terminal was used to display graphical representations of the process status. In addition to a summary display of overall status, other displays detailed various aspects of the simulated process. The graphics terminal was periodically updated to maintain a display of the current process variable values. The operator was able to control process conditions, such as opening and closing valves or turning devices on and off, by touching the appropriate device and a corresponding action button on the display. By touching appropriate areas on the display, the operator was able to bring up new displays showing different aspects of the process status.

While no effort was made to make this demonstration either fast or efficient, it did perform as expected and provided a test-bed for various communications schemes. It proved the fundamental concepts while providing much useful data on the system behavior. The implementation required a large amount of memory in the host computer to handle the terminal display and the touchscreen. This resulted in contention with the simulation code for system resources as well as scheduling problems. As a result, the response time of the system was very poor: a delay of 3-7 seconds in responding to a touch and an additional 7-18 second delay to change the display.

In evaluating the results of this experiment, it was noted that much of the poor performance could be eliminated by making better use of the microprocessor in the display terminal. The microprocessor, in this case, an Intel 8080A, could provide the interface with the touchscreen as well as performing the bulk of the work associated with the processing of status displays. This method would eliminate much of the contention in the host computer while making better use of existing capacity in the graphics terminal. A beneficial by-product would be a significant reduction in the communications load between the terminal and the host computer. The following sections describe that implementation and the results which have been obtained.

## 3. Integrated Control Station

The control station is constructed from an intelligent color graphics terminal and a touchscreen. Figure 2 illustrates the major components of the station. The display generator, communications interface, keyboard, Centrol Processing Unit (CPU), read only memory, random access memory and color CRT were already available in the graphics terminal. The screen overlay and touchscreen interface were commercially available and readily integrated with the terminal. The overlay was compatible with the

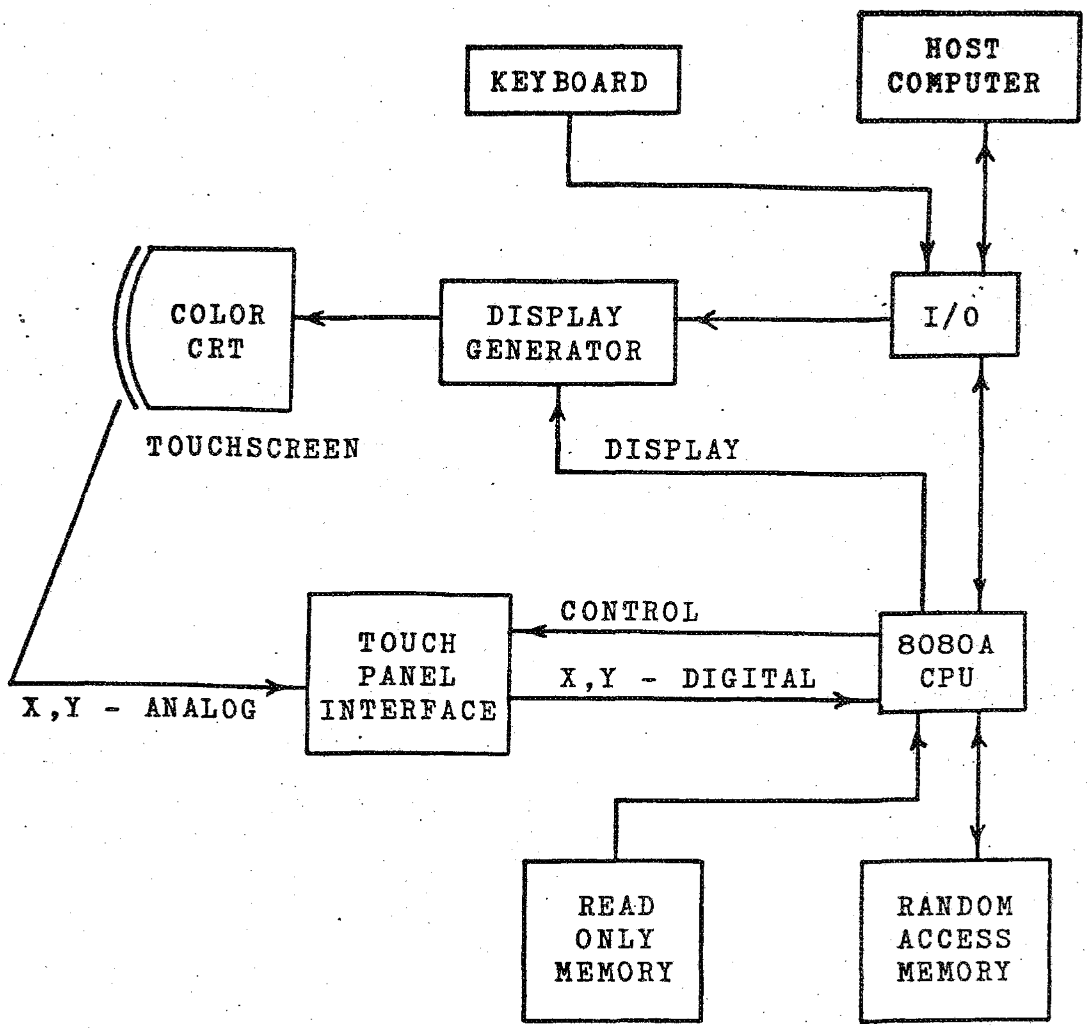


Fig. 2. Control station block diagram

specific CRT used and the interface card was designed to plug into the existing cardcage in the terminal. As a result, physical integration was quickly completed.

The normal mode of operation for the touchscreen is to provide a single coordinate report for each touch. While this is the preferred mode of operation for most activities, it was anticipated that some operator functions would best be implemented if the capability existed to provide a continuous stream of coordinate reports if contact with the screen was maintained. A possible use for such a capability would be the implementation of parameter ramping functions in which a parameter would increase in value while the screen was touched and stop changing when the touch was removed. A minor change in the interface electronics was necessary to implement that capability and to enable mode switching under software control. No other physical changes to the touchscreen were necessary during the development of the integrated control station.

The control software in the graphics terminal was supplemented to provide additional graphics capabilities. Several internal tables were defined which are used to interpret commands generated from the touchscreen or from the host computer and to update the display.

## 3.1 Graphics Capabilities

The color graphics terminal used in the control station is a very inexpensive device which provides only basic graphics capabilities. The terminal is capable of displaying eight foreground and background colors: black, white, red, green, yellow, blue, magenta and cyan. Cursor and screen control include horizontal and vertical cursor positioning, reading cursor position, cursor homing and screen clear. The device is capable of plotting with a vertical resolution of 192 points and a horizontal resolution of 160 points.

The terminal is normally used in a semigraphics mode. In this mode, the terminal is capable of presenting 48 lines of 80 characters. Each character consists of a dot matrix which can be configured to represent any combination of the available colors. In this way, graphics symbols can be defined and used in combination to compose appropriate displays. Each character can also be assigned a blink or non-blink attribute. Since this semigraphics capability is adequate for most status displays, the normal graphics mode is rarely used for our application.

### 3.2 Logical Buttons

Logical buttons are the primary mechanism for most operator action requests. The host computer defines logical buttons during display initialization and assigns them to specific locations on the screen. Attributes which may be specified include color, size of sensitive area, blink and status (either touch enabled or disabled). A maximum of 64 logical button assignments is allowed.

The display area is divided into 960 physical locations referred to as "Physical Button Numbers" (PBN) beginning at the upper left corner of the screen with PBN 1 and ending at the lower right corner with PBN 960. Since the display area contains 3840 normal size characters, each PBN occupies 4 character positions as shown in Figure 3.

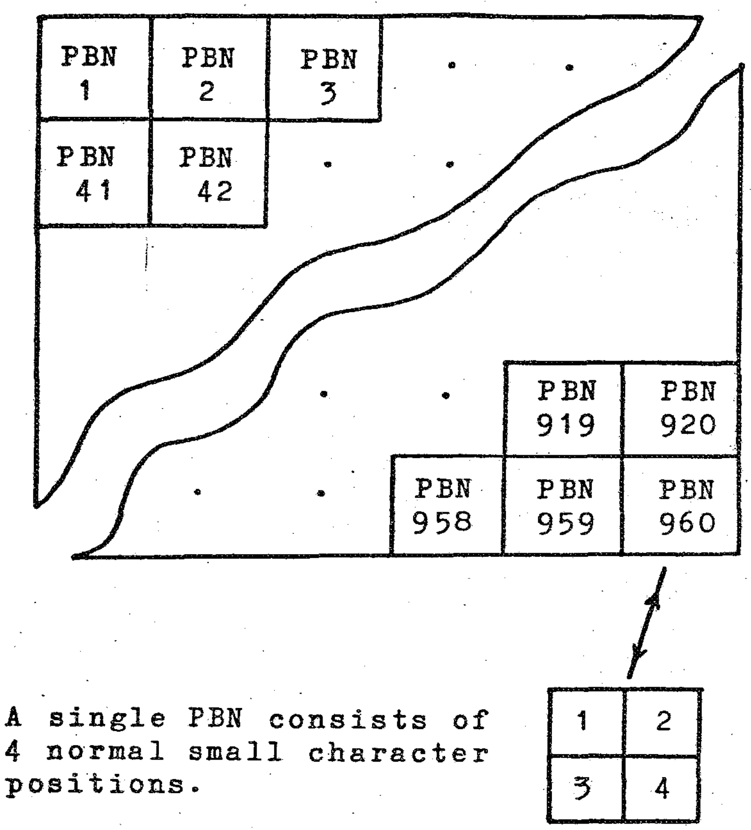


Fig. 3. Physical button coordinate system

## 3.3 Logical Meters

Logical meters provide a method of displaying alphanumeric information in real-time at a specific screen location. These "meters" are referred to in terms of logical units, similar to logical buttons, with the variable value displayed at the specified screen location. Graphics characters may also be transmitted as the update string in order to change symbols. (Changing the color of a valve to indicate a change in status, for example.)

## 4. Command Language

The command language which was implemented in the control station consists of a very simple syntax. Each command starts with a two letter mnemonic followed by required and optional parameters. The separation character is the comma. The commands fall into several broad categories: initialization/maintenance commands, touch commands, logical button and meter commands, and communications commands

### 4.1 Initialization/Maintenance Commands

These commands are used to perform display station calibrations, assist with routine maintenance, and initialize the control station characteristics.

When power is turned on, the graphics terminal comes up in native mode. The terminal will respond to keyboard and host computer input as if its firmware had not been modified. The enhanced (command) mode is entered upon receipt of a control-D. Command mode accepts only the new commands described here and rejects all escape sequences recognized in the native mode. The Native Mode (NM) command disables the command interpreter and returns the graphics terminal to the unenhanced, or native, mode of operation. It is possible to switch back and forth between the two operating modes in this manner and selectively alter the display or modify the internal tables. this capability is especially useful for developing new displays in an on-line manner.

The Calibrate Screen (CS) command initiates an interactive dialogue with the user to calibrate the touchscreen interface when system power is applied. The operator is first prompted by a small target in the lower left corner of the screen with the message "touch here." When he touches the target, the origin of the screen coordinate system is established. A target is then placed in the upper right corner of the screen and the operation repeated. This sequence sets the scale factor needed to finish the calibration sequence.

The Visible Numbers (VN) command enables the display of touch coordinates on the lower right corner of the screen. The data displayed includes the raw coordinate data received from the touchscreen, the normalized x,y location coordinates and the corresponding physical button number. This capability is particularly useful when calibrating the touchscreen interface electronics. The Disable Visible Numbers (VX) command removes this display of touch coordinates from the screen.

The Initialize Screen (IS) command is used to clear current table assignments and remove all active touchscreen areas. This command is normally issued as the first command when a new display is to be generated. It is followed by an Erase Page (EP) command which erases the first 44 lines of the display, but does not erase the bottom four lines which are reserved as a command/message area.

The Enable Message Display (EM) command causes the transmitted and received messages to be displayed on the message area of the screen. This capability is particularly useful while debugging system operations. This display is disabled with the Disable Message Display (DM) command. The Enable Alarm (EA) and Disable Alarm (DA) commands are used to turn the audible tone (beep) of the terminal on and off.

## 4.2 Touch Commands

The touch commands are used to specify the operating characteristics of the touchscreen. Only the general characteristics of the screen are affected; behavior of local areas is defined when the active areas are established. The Set Stream Mode (SS) command places the touchscreen in "stream" mode. In this mode, the location of the touch is continuously reported to the microprocessor as long as contact is maintained. A visible echo will follow the operator touch as his finger is moved over the screen. This mode is normally used to provide control of ramping functions. The Set Point Mode (SP) command is used to place the touchscreen in "point" mode. In this mode, the location of a touch is reported only when the contact is first made. In order to get a second report, the contact must be broken and reestablished. If an inactive area of the screen is touched, a visible echo appears at the touched location for a fraction of a second. If the location is active, the echo remains on the screen as a reminder until the completion of a valid command sequence. In the event of a touch error, this echo can be cleared by touching an inactive area or moved by touching the correct active area.

The Enable Screen (ES) command is used to enable the touchscreen electronics and start an interval timer which interrupts the microprocessor at a 60 Hertz rate. This interrupt initiates a routine to check for an operator touch. This command will not be executed unless the Calibrate Screen (CS) command has been previously executed. A corresponding Disable Screen (DS) command is used to disable the touchscreen electronics and stop the interval timer.

## 4.3 Button Commands

A logical button number (LBN) can be assigned to any physical button number (PBN) with the Assign Button (AB) command. This command defines a logical button, its associated physical location on the screen and its attributes. The attributes which can be assigned are the color of the button, whether or not it is blinking, whether or not it is sensitive to a touch, whether or not it is buffered, whether it is a "transmit-immediate" or a "transmit-on-enter" button and the button mode. The character codes specifying button attributes may be entered in any order since the code for each attribute is unique.

As expected, the color attribute specifies the color assigned to the button. One especially useful color provides for "invisible" buttons. When black is assigned as the button color, an "invisible" button is established.

This feature allows assignment of buttons to areas of the screen which correspond to symbols such as a valve, as shown in Figure 4, without disturbing the symbol.

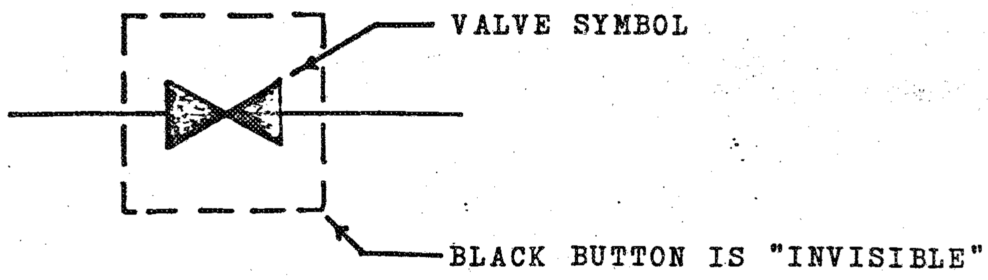


Fig. 4. "Invisible" buttons

The blink attribute specifies whether or not the button is to be blinking. This characteristic should be used sparingly in order to direct the operator's attention to an abnormal item on the display which may require his immediate attention. Active areas are indicated by solid buttons on the display while inactive buttons are hollow. When a touch is accepted by the control station, an audible beep sounds to provide feedback to the operator and the solid button becomes hollow.

The buffer attribute is used to enlarge the sensitive area around a LBN by assigning the 8 PBN's immediately surrounding the visible button indicator to the same LBN as shown in Figure 5. A touch on any of the PBN's within a relatively large area will be associated with touching a

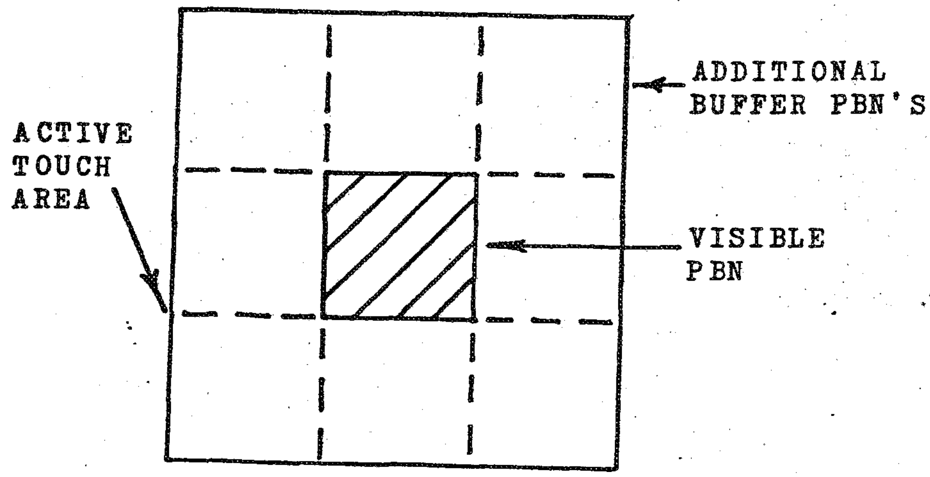


Fig. 5. Buffering of logical buttons

specific LBN. This buffer zone is invisible to the operator, but allows less critical touch positioning.

Most buttons are defined as "transmit-on-enter" buttons. When the operator presses an enabled button, a visible echo indicates that he has pressed an active area and that the station is waiting for further action. He then presses an "enter" button at which time the action associated with the previous button is initiated. If, during this sequence, the operator decides that an incorrect button was selected, he can either press the correct button, which moves the visible echo to the new button, or he can touch any inactive area of the screen to clear the previous touch. The visible echo provides positive operator feedback and the two-step entry mode reduces the probability of accidental touches resulting in unintended actions. "Transmit-immediate" buttons do not require that the operator press the "enter" button to initiate the corresponding action; the action is initiated as soon as the button is pressed. This type of button is used infrequently.

The mode attribute is simply a shorthand method of specifying button color and blink combinations. The currently defined button modes are:

| | |
|---|---|
| normal | green, no blink |
| warning | yellow, blinking |
| warning acknowledged | yellow, no blink |
| alarm | red, blinking |
| alarm acknowledged | red, no blink |

The Assign Space (AS) command is used to establish additional touch sensitive areas around a logical button beyond those assigned by buffering. These additional physical buttons can be assigned to a logical button in any pattern and quantity. The Update Button (UB) command is normally sent from the host computer to change the attributes of any currently defined logical button. Any or all of the attributes may be modified without affecting the remaining attributes.

## 4.4 Meter Commands

The meter commands, Assign Meter (AM), Update Meter (UM) and Release Meter (RM), are similar in function to the button commands. Meters are not assigned additional active areas since they are sufficiently large without them. The attributes associated with a logical meter are its starting location, length, foreground color, background color, the size of the characters displayed, whether or not blink and touch are enabled, mode and the string to be displayed. All meters are "transmit-on-enter" types.

The coordinate system used in assigning logical meters to specific screen locations is the terminal's standard x,y (80X48) coordinate system where (0,0) is the upper left corner of the screen and (79,47) is the lower right corner. The mode attribute is simply a shorthand method of specifying common foreground and background color and blink combinations. The meter modes currently defined are:

| | |
|---|---|
| normal | white on black, no blink |
| warning | yellow on black, blinking |
| warning acknowledged | yellow on black, no blink |
| alarm | red on black, blinking |
| alarm acknowledged | red on black, no blink |

### 4.5 Communications Commands

These commands are used to control communications characteristics with the host computer. The Enable Checksum (EC) command initiates a checksum routine which verifies the integrity of messages sent between the control station and the host computer. The Disable Checksum (DC) command disables this checksum routine.

The Transmit Status to Host (ST) command is issued by the host computer periodically to check for any operator action at the control station. When the request is made, one of four responses is expected. If no action has transpired since the last status check, the message "CL" (Clear) is returned to the host. If a logical button or meter has been pressed, the message "TB,n" or "TM,n" is returned, where n is the number of the logical button or meter. The other expected response is a host computer command entered through the control station keyboard. Any function which may be initiated by touching the screen may alternatively be entered as a keyboard command. This capability provides a measure of redundancy to the command entry system and enables operation to continue in a degraded mode in the event of a touchscreen failure.

## 5. Control Station Command Interpreter

The graphics terminal was modified to provide two new basic capabilities: to accept and process the commands previously described and to detect touchscreen contacts and correlate them with appropriate actions. The interpretation of the command set is accomplished simply with a command buffer, parse routine, jump table and a set of command handlers. The incoming command is placed in the command buffer and then parsed into the command mnemonic and associated parameters. The command mnemonic is the entry into a jump table which transfers control to the appropriate routine for each command. Each such routine accepts the required and optional parameters from the parse routine and performs the necessary modifications to the internal tables.

### 5.1 Control Station Internal Tables

Two tables are used to specify the location and attributes of each of the 64 LBN's and their associated PBN's. PBNTBL is a table of single byte elements mapping directly to associated PBN's on the display. Each element (byte) of this table contains two items: bit 7 is set on if this element maps to a "visible" PBN associated with a LBN and bits 0-6 contain the binary equivalent of the LBN associated with the PBN. Each LBN assignment causes at least one entry in PBNTBL and, if buffered, causes 9 such entries. There can be more than one visible PBN for a given LBN through multiple button assignments using the same LBN. ATRTBL is a 64 byte table which maps directly to a given LBN. As shown in Figure 6, an element (byte) in this table contains all information necessary to define the attributes of the LBN.

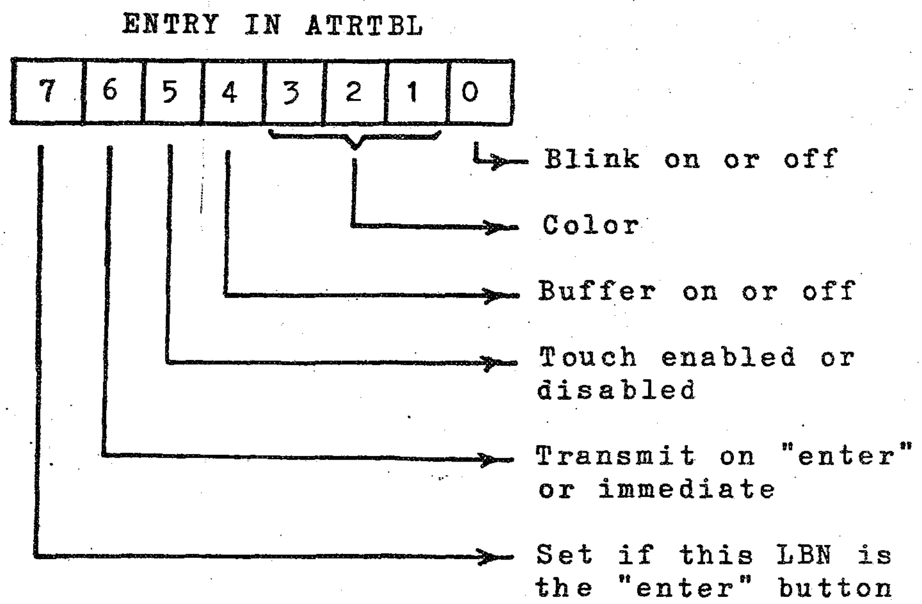


Fig. 6. Information coding of ATRTBL table

Four tables are used to define the location and attributes of the 128 logical meters. MXTBL and MYTBL are each 128 byte long tables containing the x- and y-coordinates of the logical meters. The nth element of MXTBL contains the x screen coordinate for LMN n. Note that logical meter positions are in terms of the screen coordinate system rather than in terms of PBN's. LENTBL is also 128 bytes long and contains two items of information: bits 0-6 contain the length of the logical meter and bit 7 is used as a flag to indicate whether or not the meter is touch enabled. As shown in Figure 7, MTRTBL is similar to ATRTBL for buttons in that it contains flags for all attributes (except touch) of a particular meter. Again, this table is 128 bytes long and maps directly to a specific LMN.

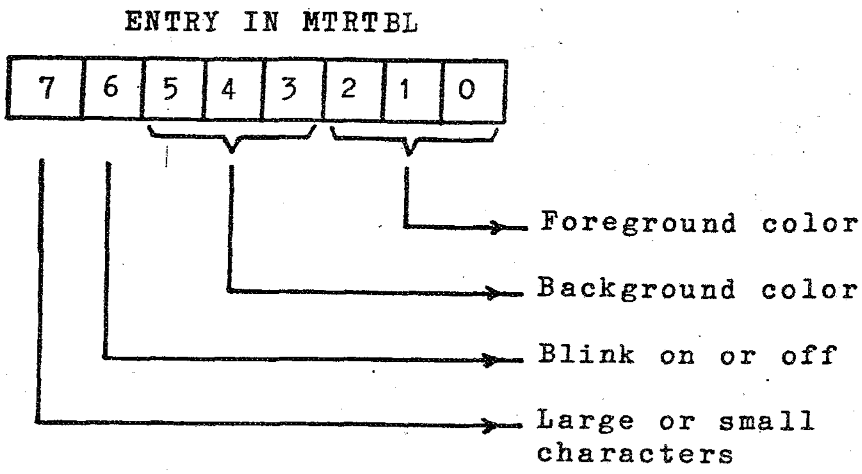


Fig. 7. Information coding of MTRTBL table

5.2 Command Generation from Touchscreen

In order to illustrate how a "TB,n" or "TM,n" message is generated when a LBN or LMN is touched, assume that the operator has just touched an active LBN. The touchscreen interface card captures x and y analog values and converts them to 12-bit digital data. A screen handler routine which is scheduled every 16 milliseconds reads the digital data and saves it in a temporary buffer for conversion. This digital data is converted first into standard x and y screen coordinates and then correlated to the PBN touched. The cursor is then moved to the x and y coordinate position calculated to establish a visible echo of the touch. PBNTBL is accessed, using the PBN as an index, to determine if the PBN is currently assigned to an LBN. If the PBN is not assigned, a short delay routine is called to allow a reasonable time period for echo retention and then the cursor is removed, resulting in the appearance of a momentary echo when unassigned locations are touched.

If, on the other hand, the PBN was assigned, then the LBN data is retreived from the PBNTBL entry and LBNTBL is accessed to determine if the button is touch enabled. If the button is disabled, the echo is again removed after a short delay. If the button is enabled, bit 6 is checked to determine the button transmission mode. If the button is a "transmit-immediate" button, then the host message buffer is immediately adjusted to include an appropriate "TB,n" message. If the button is a "transmit-on-enter" button, the LBN data is stored in a temporary memory location for retreival if the operator next touches the "enter" button.

Note that in this discussion, many details have been omitted for simplicity. Checks would also be made for logical meter touches, checks for the "enter" button, etc.

## 6. System Operation

The utilization of the control station in the SAF control scheme is illustrated in Figure 8. The host computer task ISCUP provides a periodic update of the control station display based on the current process data recorded in file DATA. ISCUP examines the Update Descriptor Table (UDT) to obtain format information needed to construct and send the proper update button and meter commands. Task ISC1 performs the initialization of the control station display, builds UDT and handles communications with the control station. Task DEVICE handles communications with

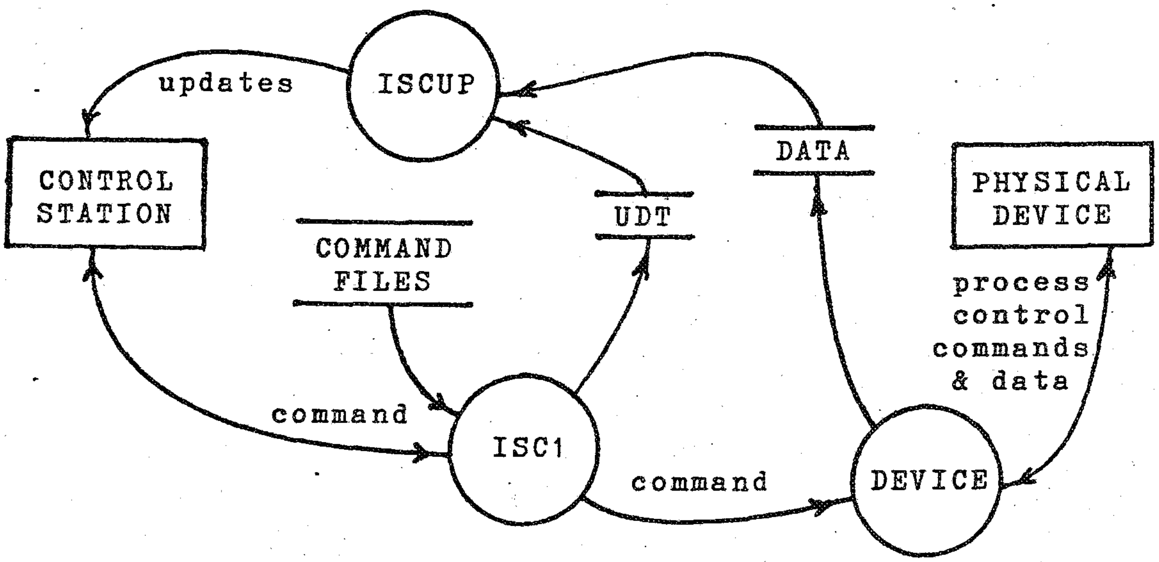


Fig. 8. System level data flow

the physical device controller. DEVICE periodically issues status commands and receives the current values of all process variables for updating file DATA. GAGE also receives process commands from ISC1 and transmits them to the device controller for execution.

There is a host program similar to DEVICE for each UOP in the SAF line to send commands to and receive data from the process. There is a host program copy of ISC1 for each control station in the control center. Only ISC1 and DEVICE are shown in Figure 8 to simplify the diagram. A single copy of ISCUP handles the updates of all control stations.

## 6.1 Host Generation of Control Station Display

When a new display is first called up, routine ISC1 reads a command file which contains commands specifying the background file to display and definitions of logical buttons and meters as well as the commands to be issued when active buttons or meters are touched.

In order to display the requested background, the control station screen is first erased with an EP command and placed into native mode with a NM command. The background display file is a frame image file consisting of 3840 character/status byte pairs (7680 bytes total). A series of escape sequences is sent to the graphics terminal to position the cursor in the upper left corner of the display and set the status for the first character. The first character is sent and the cursor moved to the next screen position. If the status for the next character has changed, an escape sequence is sent to set the proper status and the next character is sent. This sequence of changing the status, if required, and sending a character is repeated until the entire screen has been updated. A control-D character is then sent to return the control station to "COMMAND" mode. An IS (initialize screen) command is then sent to initialize all internal tables, clearing any previously defined buttons or meters.

ISC1 then continues reading button and meter assignment commands from the command file. AB (assign button) and AM (assign meter) commands are sent to the control station to set up the internal tables with the locations and attributes of logical buttons and meters. XB and XM commands in the command file generate entries into file UDT which is used to associate process variables with logical buttons and meters. SB and SM commands generate entries into a file named SOFT. This file is accessed by logical button or meter number and contains a command string to be executed when that button or meter is touched. Upon completing execution of the command file, the file is closed.

## 6.2 Host Update of Control Station Display

ISCUP periodically scans file DATA as directed by UDT and checks for changes in process variable values. When a change is detected, the current UDT entries described are accessed using the variable number to get the associated logical button or meter number and the type of data transformation required. Several routines exist to transform various types of raw data into the format necessary for transmission to the control station. When the required transformations have been accomplished, a routine is called which generates either an Update Button (UB) command or an Update Meter (UM) command which is sent to the control station.

## 6.3 Host Response to Control Station Input

In order to monitor the status of the control station touchscreen, ISC1 requests a status report, with an ST (status) command, at one second intervals. The control station response indicates whether or not a valid touch sequence has been executed or, alternatively, if a command has been entered via the control station keyboard. The response will be a "CL", "TB,n", "TM,n", or a keyboard command. (Reference section 4.5) A "CL" message indicates that no action has been requested from the control station.

If a "TB,n" or "TM,n" message is received from the control station, the corresponding soft command is retrieved from file SOFT and replaces the TB or TM command. If the command specifies that a new display be processed, acknowledges a warning or alarm condition or changes the unit operation that the control station is assigned to, then ISC1 performs the necessary steps to complete that command. Any command which is not recognized is assumed to be a unit operation command and is sent to the appropriate UOP task, such as DEVICE, to be transmitted to the unit operation for processing. If the unit operation notes a command error, it returns an error message and the operator is notified.

## 7. Performance

The initial application of the system described in this report has proven satisfactory and, in most cases, resulted in solutions to the problems encountered in the earlier approach. The host computer has been relieved of most of the "bookkeeping" functions associated with the touchscreen and, as a result, is better able to handle the tasks of operating and monitoring the process equipment. Programming of the host computer has been simplified by providing display functions (buttons and meters) which, once established, are maintained to a large degree by the control station with minimal input from the host computer. By integrating the touchscreen with the terminal, typical response times have been improved. Response delays currently experienced are in the range of 1 to 3 seconds with an average response time of 2 seconds. Changing displays requires between 5 and 11 seconds.

8. Planned Enhancements

As noted previously, the host computer currently transmits a character to each of 3840 screen locations in addition to a large number of status change characters when generating a new display. Even with no status change characters, the minimum picture transmission time at 9600 baud is 4 seconds. The majority of the screen area, even for a complex display, consists of blank areas. Currently available within the terminal, but not utilized, is the means to skip over these blank areas and move directly to a specified non-blank character position. It is estimated that with a few, not particularly complex, picture transmission optimization routines, the time required to generate a new display can be reduced to less than one second.

Current status requests by the host computer take place at one second intervals with update routines running at similar rates. Poor response times to operator touches are directly attributable to this low polling frequency. With the low demand on the host computer by the control station, it is possible to increase these rates by a factor of 2 to 4 in order to improve response times.

Logical meters and buttons must be considered a minimum graphics package. Even though these functions have proven to be effective, additional functions will be developed to provide additional process display types. Currently being developed are several types of analog meters and floating pointer indicators which may provide a display which is more easily interpreted than an all digital display.

Finally, the current method of set point entry incorporates the keyboard as an input device. Several methods of entering setpoints for a process via the touchscreen are being investigated.

9. Conclusions

Limited initial demonstrations of the system described took palace during September, 1980. A single CRT was used as an input device in the control center while operating a furnace and a pellet inspection gage. These two process line devices were completely controlled, despite the longer than desired response times noted, using a single control station located in the control center. The operator could conveniently execute any function from this remote location which could be performed locally at the hard-wired control panels.

With the installation of the enhancements described, it is anticipated that, especially in a remote operation configuration, the integrated touchscreen/graphics terminal will provide a preferable alternative to normal keyboard command input devices.