CT-82

# CT·82

# TERMINAL SYSTEM

# USER'S GUIDE

IMPORTANT NOTE

Although every effort has been made to make the supplied software and its documentation as accurate and functional as possible, Southwest Technical Products Corporation will not assume responsibility for any damages incurred or generated by such material. Also, Southwest Technical Products Corporation reserves the right to make changes in such material at any time.

SOUTHWEST TECHNICAL PRODUCTS CORPORATION
219 W. RHAPSODY SAN ANTONIO, TEXAS 78216

## How to Use This Manual

The CT-82 Terminal System is a sophisticated and complex unit. This instruction manual has been written to gradually introduce the novice to the CT-82's features and operational details. Readers with more computer experience may wish to skip some of the material in the earlier chapters.

For those people who are either professional programmers or experienced with writing programs to drive terminals, Chapters 1, 2 and probably Chapter 3 should be skipped. Chapter 4, Appendix A and Appendix B should provide all of the information necessary.

Those people who have heard about the CT-82 and want to know, "What's so great about the CT-82?", should read Chapter 1 and if more information is required should read Chapter 3. Chapter 2 should be skipped.

Those people who have a CT-82 available and who prefer to learn by "tinkering", should follow through with the demonstration of the CT-82 provided in Chapter 2, then continue with Chapters 3 and 4.

The Appendices of this manual and specifically Appendices A and B, are intended as reference material and contain all of the detailed information about the terminal.

All of the descriptions of the hardware used in the CT-82 is included in the Appendices starting with Appendix J. Typical interconnection methods are shown in Appendix J.

## Notation

^—The "^" character is used throughout this manual in two contexts. If it precedes another character in a control character sequence, it means the same as CTRL. The other context is as the ASCII data character for a circumflex.

[ ]—When a character, word or phrase is enclosed in square brackets, it is a brief description of the byte that should appear in that position. For instance, [control character] is used in control character sequences that require any one of the control characters to appear at that location in the sequence. As another example, [count] is used where a single byte containing a binary count is required.

UPPER CASE LETTERS—When describing a key that appears on the keyboard of the CT-82, the word that appears on the key is capitalized. For instance, SHIFT refers to the shift key that appears on most keyboards including the CT-82's.

(X,Y)—When used to describe a position on the screen, X generally refers to the horizontal position and Y to the vertical position.

(m,n)—When two numbers are used to describe a position on the screen, the first number of the pair describes the horizontal position and the second describes the vertical position.

(hex)—When a number is followed by this, the number is a hexadecimal number.

(decimal)—When a number is followed by this, the number is a decimal number.

## Table of Contents

## Chapter 1. Introduction

Until recently all terminal functions were designed with hardware logic. A relatively simple terminal with limited functions could easily require as many as sixty or more integrated circuits. More sophisticated terminals with a moderate amount of intelligence could easily have over a hundred IC's. All this has now changed. With the introduction of MOS video controller circuits it has become possible to design a terminal using a controller and a microprocessor that will perform almost any imaginable function with firmware. The CT-82 has one hundred twenty-eight separate functions—all of which are firmware driven. It contains fewer parts than most "dumb" terminals.

The normal screen format is 16 lines (20 lines selectable) with 82 characters per line. This is an upper-lower case display with a 7 x 12 dot matrix. The high resolution characters are displayed on a "Motorola Data Products" M-2000 series monitor with a green P-31 phosphor. This monitor has a 12 Mhz video bandwidth and dynamic focus circuits to insure a crisp well focused display over the entire face of the tube. An alternate all capital letter format is available (optional) with 22 lines and 92 characters per line. The lower case portion of this character set has graphic symbols. In this mode the lines may be moved together to give a solid figure, or line. Direct cursor addressing combined with the plotting capability makes it possible to indicate the end points of a line and then to automatically draw a line between them.

Both the monitor and the character generator have sockets provided for alternate material in the form of an EPROM. This makes it possible to have special terminal functions, or character sets that can be switched in under computer control.

The CT-82 has its own internal editing function. This allows inserting and deleting lines and characters, erasing quadrants, or lines; doing rolls, scrolls, slides and other similar functions. The CT-82 can block transmit completed material to the computer, or output material to its own remote printer through the built-in parallel printer I/O port. The terminal can be programmed to operate at any system baud rate that is normally used from 50 to 38,400. The baud rate may be changed at any time within this range with a software command.

The cursor position, type of cursor, cursor ON-OFF and blinking are all provided. A command is provided to print control characters and also to turn on and off a tape punch, or tape reader. Protected fields, shift inversion, dual intensity and many other miscellaneous features make the CT-82 one of the most flexible terminals available.

A fifty-six key alphanumeric keyboard plus a twelve key cursor pad is standard. Connection to the terminal is through a standard DB-25 connector and RS-232-C signal levels. The CT-82 weighs 20 lbs, and is a compact 18 wide, 10 high and 18 inches deep.

When using this manual note that Chapter 2 contains a step-by-step guide to demonstrate how the CT-82 is used. It is not a complete description of all of the available functions, but is provided to give the user a learning tool. Chapter 3 contains a complete description of the CT-82's configurations, as well as a brief description of all of the control functions available. Some of the control functions cannot be used effectively without some background knowledge about their use. These principles of operation are provided in Chapter 4. Appendix A contains a detailed description of each control function. Appendix B describes the operation of the terminal when it receives a data character. These appendices are intended as a reference for users already familiar with the CT-82. Appendix C is a glossary containing definitions for many of the terms used throughout this manual. These terms are defined as they apply to the CT-82. Appendix D is a flow chart describing the operation of the terminal when it receives a character. Appendices E, F, G, H and I provide indices and tables useful as quick references. The remainder of the appendices, starting with Appendix J, describe the CT-82's hardware with Appendix J containing information on connecting the CT-82 to a computer system.

### 1.1 CRT Screen

The CT-82 uses the ASCII convention of dividing the characters received by the terminal into two sets, control characters and data characters. Data characters are simply displayed on the screen by the CT-82. All of the control characters have been associated with special functions used to

manipulate the data already displayed on the screen. These special functions may be used to do things like erase or shift the data on the screen. More detailed information on these and the other control functions is furnished in the following chapters.

**1.2 Keyboard**

The CT-82's keyboard provides the capability for entering all of the characters described by the table of ASCII characters in Appendix E. To enter a control character from the keyboard, it is necessary to hold the CTRL key depressed while typing the selected character.

The CT-82's keyboard has several special purpose keys which are listed below with a brief description of each.

SHIFT – The key that is used to access the alternate characters on the keyboard.

CTRL – The key that is used to generate an ASCII control character from the keyboard. This key is used in the same manner as the SHIFT key.

CAN–This key generates the same character as CTRL X. This character when echoed to the terminal executes the Cancel function.

ESC–This key generates the same character as CTRL [. When echoed to the terminal, it forces the execution of the Escape function.

SHIFT LOCK–When this key is latched down, the keyboard responds as if the SHIFT key were held depressed. This key can be unlatched by depressing it again.

RETURN–This key generates the same character as CTRL M. When echoed to the terminal, it forces the execution of the Carriage Return function.

DEL–This key generates the ASCII rubout character. The rubout character is generated by typing SHIFT _ (the underline character, shifted).

LF–This key generates the same character as CTRL J. When echoed to the terminal, it forces execution of the Line Feed function.

BACKSPACE–This key generates the same character as CTRL H. When echoed to the terminal, it forces execution of the Backspace function.

SCROL DOWN–This key generates the same character as CTRL O. When echoed to the terminal, it forces execution of the Scroll Down function.

SCROL UP–This key generates the same character as CTRL N. When echoed to the terminal, it forces execution of the Scroll Up function.

FORM–This key generates the same character as CTRL L. When echoed to the terminal, it forces execution of the Form Feed function .

HOME–This key generates the same character as CTRL P. When echoed to the terminal, it forces execution of the Home Up function.

XMIT–This key generates the same character as CTRL W. When echoed to the terminal, it forces execution of the Transmit function.

DELETE–This key generates the same character as CTRL Z. When echoed to the terminal, it forces execution of the Delete Line, Up function.

INSERT–This key generates the same character as CTRL Y. When echoed to the terminal, it forces execution of the Insert Line, Up function.

BREAK–This key causes the serial output line from the terminal to be held in the "mark" state until either another key is typed, or the terminal detects a "mark" state on the incoming line. This is useful in communication disciplines.

←—This key generates the same character as CTRL D. When echoed to the terminal, it forces execution of the Bump Cursor Left function.

→—This key generates the same character as CTRL I. When echoed to the terminal, it forces execution of the Bump Cursor Right function.

↑—This key generates the same character as CTRL A. When echoed to the terminal, it forces execution of the Bump Cursor Up function.

↓—This key generates the same character as CTRL B. When echoed to the terminal, it forces execution of the Bump Cursor Down function.

**NOTE:** The keyboard supplied with the CT-82 has an automatic repeat feature. Depressing a key and holding it down for more than 1 second will cause the keyboard to enter the automatic character repeat mode.

## Chapter 2. Demonstration of the CT-82

The following chapter provides a brief description of some of the simpler configurations and control functions provided with the CT-82. It is provided so that the user can start using the terminal immediately, without the necessity of learning all of the details involved in using the CT-82 at full capacity.

### 2.1 Powering on the CT-82

The CT-82 is set up so that when it is turned on it will reset the terminal to its initial configuration. This allows the terminal to recover from any situation in which it may have been put. It also means that any previous configuration will be lost. The initial or default configuration depends on the setting of the CONFIGURE switch on the back panel of the CT-82 and on an 8-bit DIP switch under the CT-82's cover.

To begin using the terminal it is recommended that the user set the CT-82 to half-duplex, conversational mode. This allows the user to use the terminal stand-alone (without attaching it to a computer). Begin by removing the cover of the CT-82. This is accomplished by lifting the back of the cover then sliding it towards the front. The 8-bit DIP switch will be clearly visible in the corner of the circuit board closest to both the keyboard and the CRT (picture tube). These switches should be set as described below. Use the numbers etched on the circuit board, NOT the numbers that are imprinted on the DIP switch itself.

| | | | | |
|---|---|---|---|---|
| → | Switch #0 | = | OFF | } |
| → | Switch #1 | = | ON | } |
| ← | Switch #2 | = | ON | } Select 9600 baud. |
| ← | Switch #3 | = | OFF | } |
| ← | Switch #4 | = | OFF | } |
| → | Switch #5 | = | ON | Select Conversational mode. |
| → | Switch #6 | = | OFF | Select Half Duplex mode. |
| ← | Switch #7 | = | ON | Select 16-line screen. |

Now replace the cover by reversing the process described above. Set the CONFIGURE switch on the back panel to the PROGRAMMABLE setting. The PROGRAMMABLE setting uses the DIP switch for initial configuration while the AUTO setting will select 9600 baud, conversational mode, full duplex mode and the 16-line screen. Press the POWER switch on the back panel to turn the CT-82 on. If the terminal is already on, turn it off and then back on.

As mentioned, configuring the terminal in the above manner will enable the user to become familiar with the various aspects of the terminal without having a computer attached. After the user is comfortable with the features of the terminal, the proper computer connection can be made. Typical connection methods are described in Appendix J of this manual. Appendices starting with Appendix J all contain information relating to the technical aspects of the terminal. Even the user who has little or no electronics expertise should read over these sections for some general information.

### 2.2 Configuring the Cursor

There are two different shapes in which the cursor may be displayed. The block cursor is the one blinking in the upper left corner of the screen when the CT-82 is powered on. To change to the other shape, type a CTRL ^ followed by a CTRL T. This shape is similar to an underline character. To change the shape back to the previous shape, type a CTRL ^ followed by a CTRL D.

To change from a blinking cursor to a non-blinking cursor, type a CTRL ^ followed by a CTRL S. To change back type a CTRL ^ followed by a CTRL C.

It is recommended that the user choose the cursor configuration which he prefers and write all programs so that they will configure the cursor as desired. If a blocked, blinking cursor is the one selected no configuration will be necessary as the terminal assumes that configuration on power-up.

## 2.3 Cursor Control

Since a great number of the control functions make use of the cursor position in determining where the functions are to be performed, it is necessary that the cursor be easily controlled. The following section describes some of the cursor movement and control functions.

It is often necessary to position the cursor to the upper left corner of the screen. To do this, type a CTRL P. Alternatively, if the CT-82 has the Function keypad and not the numeric keypad at the right side of the keyboard, then simply typing the HOME key will do a Home Up (position to upper left corner of screen). A Home Down (position to lower left corner of the screen) will be performed when a CTRL C is typed. CTRL \ followed by a CTRL P will cause a Home Up to Right and a CTRL \ followed by a CTRL C will cause a Home Down to Right. Note: The HOME key may be used in place of a CTRL P at any time.

There are control functions to move the cursor up, down, left or right by one position–CTRL D to bump it left, CTRL I to bump it right, CTRL A to bump it up and CTRL B to bump it down. If the Function keypad is available, the left arrow is equivalent to CTRL D, the right arrow to CTRL I, the up arrow to CTRL A and the down arrow is equivalent to CTRL B.

There are other cursor movement and positioning control functions, but these are more suitable to being produced by a computer, rather than from the keyboard. Probably the most useful of these is the 'Set Cursor Position to (X,Y)' where X is the horizontal position and Y is the vertical position.

The ability to turn the cursor off is sometimes very useful. To do this, type a CTRL E. To turn it back on, type CTRL U.

## 2.4 Erases

The CT-82 is capable of performing a large number of very useful erases, but describing a few of them here will show the utility of all of them. First type a lot of characters onto the screen so that they may be erased. The repeat feature on the keyboard will prove useful for this. Simply hold down a key until it starts repeating.

Position the cursor so that there is at least one line of characters above the cursor and one line below the cursor. It should also be positioned towards the middle of the line. Now type CTRL V. This will cause an 'Erase to End of Frame function to be performed.

Position the cursor to the middle of a line of characters and type a CTRL F. This will erase the line from the cursor to the end of the line.

Type in some characters on a line then type the BACK SPACE key. This will perform a function similar to typing the BACK SPACE key on a typewriter. Typing a BACK SPACE is equivalent to typing a CTRL H. Now type the CAN key. The cancel function will backspace to the beginning of the line. Typing CAN is equivalent to typing CTRL X.

There is another useful feature of BACK SPACE and CANcel. Turn off the cursor as described in the Cursor Control section, then type in a few characters. Now turn the cursor back on and type some more characters (on the same line) that can be distinguished from the preceding characters. Now typing CAN will only back up the cursor to the CURSOR ON position. This is because the CURSOR ON position is treated as a boundary by the Cancel control function. BACK SPACE will also treat the CURSOR ON position as a boundary.

## 2.5 Scrolls, Inserts and Deletes

To demonstrate the scrolls effectively, type characters on the top two lines of the screen as well as the bottom two. Type CTRL N to scroll the screen upwards and CTRL O to scroll the screen downwards. If the Function keypad is available, type either SCROL UP or SCROL DOWN to get the same effect.

To see the effect of the 'Insert Line, Up' function, type several lines of characters near the middle of the screen. Position the cursor so that there is at least one line of characters above the

cursor and at least one line below the cursor, then type CTRL Y. The INSERT key may be used, if if it is available. With the cursor positioned to the line which was inserted, type CTRL Z. This will delete the line which was just inserted. If it is available, the DELETE key may be used.

To insert a character into a line that is already on the screen, type a line onto the screen. Now position the cursor to the position in the line where you wish to insert the character and type a CTRL \ followed by a CTRL X followed by the character you want inserted. To delete a character of the line, position the cursor as previously described and type a CTRL \ followed by a CTRL H.

### 2.6 Configuring the Screen

The user has the choice of 3 screen sizes and possibly two character sets. To get the choice of two character sets, a second character generator ROM must be plugged into the terminal's optional character generator socket. To configure for the 16 by 82 screen, type CTRL \ followed by CTRL Q or CTRL S depending on which character set is desired. If the optional character set is not available, the CTRL \ , CTRL S sequence will yield unpredictable results. Typing CTRL \ followed by CTRL R will configure the screen for 20 lines by 82 characters using the normal character generator ROM. CTRL \ followed by CTRL T yields the same screen size, but uses the alternate character set.

The Pseudo-Graphics mode configuration uses a 22 line by 92 character screen format with the alternate character generator. To use any of the graphics control functions, the graphics character generator ROM must be used as the alternate character set. If the graphics character generator is not used, all of the graphics control functions will cause unexpected characters to appear on the screen.

### 2.7 Graphics

Before using any of the graphics control functions, the CT-82 must be configured for graphics mode. To do this from the keyboard, type CTRL ] followed by a CTRL V. The screen will be cleared and the terminal set up for graphics. Note: If the graphics character generator ROM is not used as the alternate character generator, the graphics control functions will not work as desired.

To demonstrate the use of the graphics control functions, only the 'Set' functions will be used. Besides the 'Set' functions, 'Clear' and 'Invert' functions are also implemented. The 'Set' functions insure that the specified pixel(s) are turned on, the 'Clear' functions insure that the specified pixel(s) are turned off and the 'Invert' functions change the specified pixel(s) to their opposite sense. This means that pixels that are already on get turned off and vice versa.

To set the pixel at location (Ø, Ø) from the keyboard requires that the user knows which key(s) transmit a zero. Location (Ø, Ø) is horizontal position Ø, vertical position Ø. By referring to the ASCII character set table in Appendix E, the user will find that to enter a zero from the keyboard he must type a CTRL @. Type a CTRL] followed by a CTRL S followed by a CTRL @ followed by another CTRL @. Note: the first two control characters specify that the CT-82 is to set a single pixel and the second two control characters specify the screen position, horizontal position first and vertical position second. To set the pixel at location (100, 50), type the following sequence of characters: CTRL ], CTRL S, d, 2 (100 = 'd' and 50 = '2').

Many times it is useful to be able to connect two points on the screen with a straight line. To connect the two pixels at locations (Ø, Ø) and (1ØØ, 5Ø), type the following control sequence: CTRL ], CTRL C, CTRL @, CTRL @, d, 2. To draw a diagonal, type the following control sequence: CTRL ], CTRL C, CTRL @, 2, d, CTRL @. To draw a box around the X on the screen, type the following control sequences.

CTRL ], CTRL C, CTRL @, CTRL @, d, CTRL @
CTRL ], CTRL C, d, CTRL @, d, 2
CTRL ], CTRL C, d, 2, CTRL @, 2
CTRL ], CTRL C, CTRL @, 2, CTRL @, CTRL @

## 2.8 Carriage Return and Line Feed

Many computer systems are implemented so that Carriage Return and Line Feed are handled one way and other systems use them in an entirely different manner. The CT-82 may be configured to be compatible with most of these systems. For instance, some systems expect the terminals to automatically generate a Scroll Up function when a Line Feed would put the next line below the screen. Other systems expect the Line Feed to be ignored. The CT-82 can be used in either environment by Enabling or Disabling the Scroll on Line Feed. To Enable Scroll on Line Feed, type CTRL ^ followed by CTRL H. To Disable Scroll on Line Feed, type CTRL ^ followed by CTRL X.

Some systems expect the terminal to automatically issue a Line Feed with every Carriage Return, others do not. To enable an Automatic Line Feed on Carriage Return, type CTRL ^, CTRL Y. To Disable an Automatic Line Feed on Carriage Return, type CTRL ^, CTRL I. If your system expects the terminal to automatically issue a Carriage Return when the rightmost column of the screen is reached, type CTRL ^, CTRL J, to configure the CT-82. To Disable this Automatic New Line on Overflow, type CTRL ^, CTRL Z.

Powering the terminal on will automatically Enable Scroll on Line Feed, Disable Automatic Line Feed on Carriage Return and Enable Automatic New Line on Overflow.

## 2.9 Summary

This chapter has led the user through a step by step demonstration of the CT-82 terminal by showing him how to issue commands to the terminal from the keyboard. This has only been a demonstration though. The CT-82 is capable of many more functions than were presented in this chapter. Also, at the beginning of this demonstration the terminal was configured for half duplex mode so that the keyboard could be used to issue commands to the terminal. The CT-82 can be used more effectively by using an attached computer to issue the commands to the terminal. For more detailed information about the control functions, refer to the next chapter of Appendix A. The user can gain a great deal of understanding about the functions if he experiments with them from the keyboard as in this chapter.

## Chapter 3. Using the CT-82

This chapter is provided to give the user a start in learning how to use the CT-82 to the fullest possible extent. All of the possible configuration options will be described as well as a brief description of all of the control functions that may be used.

### 3.1 Configuring the CT-82

The CT-82 gets a great deal of its flexibility from its many possible configurations. The things that can be configured are as follows: The screen size, the character set, the baud rate and sixteen option flags.

#### 3.1.1 Configuring the Screen Size

The user has a choice of three screen sizes: 16 lines by 82 columns, 20 by 82 and 22 by 92. The screen size is selected by issuing one of the Configure Screen commands. The 16 by 82 screen is considered the "normal" screen size. The 20 by 82 configuration gains additional lines by leaving less space between lines. This means that letters with descending tails, like y, j and g, will be run together with the line below. The 22 by 92 configuration is provided for the pseudo-graphics support.

#### 3.1.2 Configuring the Character Set

The user has a choice of two character sets: The standard character set and the alternate character set. The standard character set is provided by a Motorola MCM66750 character generator ROM. This ROM provides displayable characters for the entire ASCII character set, including control characters. The alternate character set may be furnished by the user and plugged into a socket provided for that purpose. The optional graphics character generator ROM described in this documentation may be obtained from Southwest Technical Products.

#### 3.1.3 Configuring the Baud Rate

The CT-82 can be configured for one of 32 different baud rates in the range of 50 baud to 38,400 baud. The baud rate is changed by issuing a Set Baud Rate command. A list of the available baud rates is given in Appendix M.

#### 3.1.4 Option Flags

There are 16 user settable option flags provided in the CT-82. They provide configuration capability ranging from control of the characters displayed, to control of the way characters are transmitted from the keyboard. The following is a list of the option flags and a brief description of each.

**Disable/Enable Escape Character—**

When the escape character is disabled, any escape character, 1B (hex), received by the CT-82 will be ignored. When enabled, escape characters received by the terminal will cause the Escape control function to be executed.

**Clear/Set Escape Data Mode—**

When escape data mode is set, all control characters received by the CT-82 will be displayed on the screen as if they were data characters. After a control character has been displayed the associated control function will be executed.

**Clear/Set Graphics Cursor Mode—**

This option bit is for internal use only. Setting or clearing this flag may cause the CT-82 to behave in an unexpected manner.

**Set Blinking/Non-Blinking Cursor—**

Clearing this flag reconfigures the CT-82 to display the cursor so that it blinks. Setting it forces the terminal to display the cursor without blinking.

**Set Block/Underline Cursor–**

Setting this flag configures the CT-82 to display the cursor as an underline characater. Clearing the flag causes the full-sized cursor to be displayed.

**Display Cursor/Suppress Cursor Display–**

When this flag is clear, the cursor will be displayed on the screen at the current cursor position. When the flag is set the cursor position will be maintained without displaying the cursor on the screen.

**Write Unprotected/Protected Characters–**

All data characters displayed on the screen are written to the screen either as protected (low intensity) or as unprotected (high intensity) characters. Setting this flag causes all subsequent data characters to be written as protected (low intensity) characters. Clearing the flag causes all subsequent data characters to be written as unprotected (high intensity) characters.

**Ignore/Honor Character Protection (Retain Protected Fields Option)–**

When this flag is set, any character that is write protected will not be modified by control functions like the erases. When the flag is clear, write protected characters are treated no different than unprotected characters.

**Enable/Disable Scroll on Line Feed**

If this flag is clear, a line feed with the cursor on the bottom line of the screen causes the screen to be scrolled up one line. If the flag is set, a Line Feed with the cursor on the bottom line of the screen will be ignored.

**Disable/Enable Automatic Line Feed on Carriage Return–**

If this flag is set, the CT-82 will automatically execute a line feed function after executing a carriage return. Otherwise, no line feed function will be executed on execution of a carriage return function.

**Enable/Disable Automatic New Line on Overflow–**

When this flag is clear, a character entered into the rightmost column of the screen will automatically cause the execution of a carriage return, line feed sequence. Otherwise, entering a character into the rightmost column has no special significance.

**Enable/Disable Rubout as Data–**

When this flag is clear the ASCII rubout character, 7F (hex), is treated as a data character. Otherwise, it is ignored.

**Enable Upper and Lower Case/Set Upper Case Only–**

This option flag applies only to the alphabetic (A-Z) characters entered at the keyboard. If the flag is clear, both upper and lower case characters may be entered from the keyboard. If the flag is set, both upper and lower case characters are entered as upper case only. (Lower case characters entered by the user are internally converted to upper case.)

**Enable/Disable Shift Inversion–**

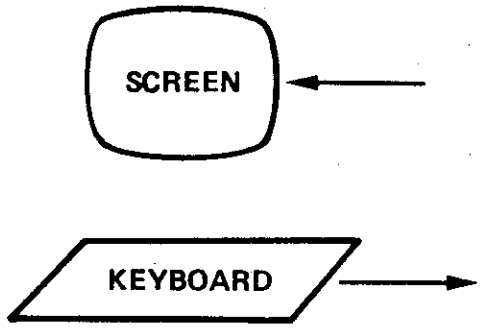
This option flag applies only to the alphabetic (A-Z) characters entered at the keyboard. When shift inversion is enabled, non-shifted alphabetic characters are transmitted as upper case characters, while shifted alphabetic characters are transmitted as lower case characters. This is inverted from the normal sense of the SHIFT key on typewriter keyboards. When shift inversion is disabled, non-shifted alphabetic characters are transmitted as lower case characters, while shifted alphabetic characters are transmitted as upper case characters. This is the same as the normal sense of the SHIFT key on typewriter keyboards.

**Set Full/Half Duplex—**

This option flag is ignored unless the "Set Conversational/Paged Edit mode" option flag is set for conversational mode. Half and full duplex have no meaning in paged edit mode.

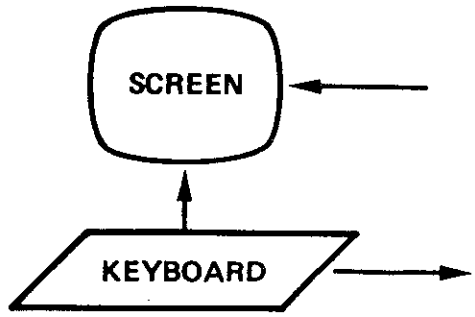
When the CT-82 is configured for full duplex while in conversational mode, two data paths are set up within the terminal as follows:

1.) Characters entered from the KEYBOARD are TRANSMITTED, and
2.) Characters RECEIVED are displayed on the SCREEN.




When the CT-82 is configured for half duplex while in conversational mode, three data paths are set up within the terminal:

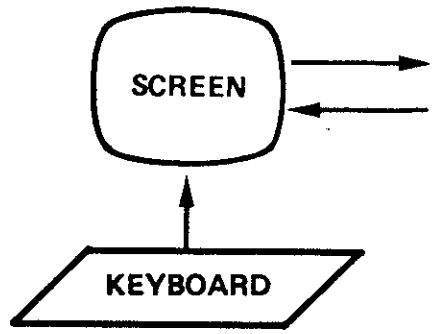
1.) Characters entered from the KEYBOARD are displayed on the SCREEN.
2.) Characters entered from the KEYBOARD are TRANSMITTED and
3.) Characters RECEIVED are displayed on the SCREEN.




**Set Conversational/Paged Edit Mode—**

When the CT-82 is configured for conversational mode, the data paths set up within the terminal are as described under the "Set Full/Half Duplex" option flag. For details, see the 'Set Full/Half Duplex' option flag documentation. When the CT-82 is configured for paged edit mode, the data paths within the terminal are set up as follows:

1.) Characters entered from the KEYBOARD are displayed on the SCREEN,
2.) Characters read from the SCREEN may be TRANSMITTED, and
3.) Characters RECEIVED are displayed on the SCREEN.



## 3.2 Control Characters

All terminals have the capability to display data either on a CRT (picture tube), paper or in some other manner. The things that makes a terminal useful is how it can manipulate that data once it has received it. Control functions are the method provided to force the CT-82 to manipulate the data as desired by the user. Processing of data by the control function is initiated by sending a control character(s) to the terminal. The control character(s) may be sent from the terminal's keyboard or from an interfaced computer system.

Some of the types of things that may be done with the CT-82 are the following: 1.) Position the cursor anywhere on the screen. The cursor position is used as a reference point for many of the terminal functions listed below. Manipulating the cursor is described under the headings CONFIGURE CURSOR and CURSOR CONTROL in the following section. 2.) Remove data that is no longer of interest or that is in error. This is accomplished on the CT-82 by using the control functions described under the heading ERASES. 3.) Move data on the screen upwards, downwards, left or right. The control functions described as ROLLS AND SLIDES may be used to perform this movement. 4.) Insert or Delete a small portion of data on the screen. To do this, use the functions under the heading INSERTS AND DELETES. 5.) Change the number of characters on the screen and change the character set used by the terminal. The functions under CONFIGURE SCREEN may be used for this purpose. 6.) Draw graphs or pictorials on the screen. Use the functions under the heading GRAPHICS. 7.) Use the terminal as a glass teletypewriter. Typical teletypewriter control functions are listed under TELETYPEWRITER FUNCTIONS. 8.) Change the way in which Carriage Return and Line Feed operate. Look under CONFIGURE CR/LF. 9.) Modify a single character or the way a character is handled. The control functions listed under CHARACTER CONTROL may be used to manipulate single characters or to configure the way in which a character is handled. 10.) Communicate with a device attached to the terminal. These control functions are listed under DEVICE CONTROL. 11.) Other useful terminal configuration abilities are listed under CONFIGURE INTERFACES and CONFIGURE MODES. 12.) Treat the character to follow as a special case. Several special cases are handled by the functions under LEADIN CHARACTERS. 13.) The control functions under MISCELLANEOUS may prove to be some of the more interesting control functions supplied on the CT-82.

## 3.3 Control Functions

The following is a list of the control functions available on the CT-82 with brief descriptions of each. A more detailed description of each control function is provided in Appendix A.

### 3.3.1 Configure Cursor

**DISPLAY CURSOR (^ ^ ^E)**—Cause the cursor to be displayed. This function differs from Cursor On in that the CURSOR ON position remains unchanged.

**SUPPRESS CURSOR DISPLAY (^ ^ ^U)**—Turn off the displayed cursor. This function is different from the Cursor Off function in that the CURSOR ON position is unchanged.

**SET BLINKING CURSOR FORMAT (^ ^ ^C)**—Configure the cursor to blink when it is displayed.

**SET NON-BLINKING CURSOR (^ ^ ^S)**—Configure the terminal to use a non-blinking cursor.

**SET BLOCK CURSOR FORMAT (^ ^ ^D)**—Configure the cursor to display as a block, as opposed to an underline.

**SET UNDERLINE CURSOR (^ ^ ^T)**—Configure the terminal to display the cursor as an underline character.

### 3.3.2 Cursor Control

**BUMP CURSOR DOWN (^B)**—Move the cursor one line downward on the screen. Bump cursor down is not the same function as Line Feed. See the documentation on Line Feed for details.

**BUMP CURSOR LEFT (^D)**—Move the cursor one position to the left on the screen. This function is not the same as the Backspace function. See the documentation on Backspace for details.

**BUMP CURSOR RIGHT (^I)**—Move the cursor one position to the right on the screen.

**BUMP CURSOR UP (^A)**—Move the cursor one line upward on the screen.

**CURSOR OFF (^E)**—Cause the cursor not to be displayed on the screen. The cursor position is defined whether or not the cursor is currently being displayed.

**CURSOR ON (^U)**—Cause the cursor to be displayed on the screen. The format of the displayed cursor depends on the current setting of the Cursor Select Option bits.

**HOME DOWN (^C)**— Position the cursor in the lower left corner of the screen.

**HOME DOWN TO RIGHT (^\ ^C)**—Postion the cursor in the lower right corner of the screen.

**HOME UP (^P)**—Position the cursor in the upper left corner of the screen.

**HOME UP TO RIGHT (^\ ^P)**—Position the cursor in the upper right corner of the screen.

**MOVE CURSOR DOWN (^\ ^B [count])**—Move the cursor downward on the screen by the number of lines specified in the [count].

**MOVE CURSOR LEFT (^\ ^D [count])**—move the cursor to the left by the number of positions specified in the [count].

**MOVE CURSOR RIGHT (^\ ^I [count])**—Move the cursor to the right the number of positions specified in the [count].

**MOVE CURSOR UP (^\ ^A [count])**—Move the cursor upward on the screen the number of lines specified in the [count].

**SET CURSOR POSITION (X, Y) (^K [X] [Y])**—Position the cursor to a specified horizontal and vertical position. [X] specifies the horizontal position and [Y] specifies the vertical position.

**SET CURSOR POSITION (Y, X) (^\ ^K [Y] [X])**— Position the cursor at a specified vertical and horizontal position. [Y] specifies the vertical and [X] specifies the horizontal position.

**SET HORIZONTAL CURSOR POSITION (^\^W [X])**—Set the horizontal cursor position to [X].

**SET VERTICAL CURSOR POSITION (^\ ^G [Y])**—Set the vertical cursor position to [Y].

### 3.3.3 Erases

**BACKSPACE (^H)**— Erase the last data character displayed. A "normal" backspace occurs as follows; 1) the cursor is moved left one position, and 2) the character at the new cursor position is replaced with a blank.

**CANCEL (^X)**—Erase to the left from the current cursor position to the Cursor On position. This function is equivalent to using the Backspace function repeatedly.

ERASE FIELD (^] ^F)—Erase the remainder of the field to the right of the cursor. Consective characters with the same protection are considered a field.

ERASE NORTH EAST QUADRANT (^\ ^\)—Erase the screen quadrant above and to the right of the cursor position.

ERASE NORTH WEST QUADRANT (^\ ^L)—Erase the screen quadrant above and to the left of the cursor position.

ERASE SOUTH EAST QUADRANT (^\ ^]) —Erase the screen quadrant below and to the right of the cursor position.

ERASE SOUTH WEST QUADRANT (^\ ^M)—Erase the screen quadrant below and to the left of the cursor position.

ERASE TO BEGINNING OF FRAME (^\ ^V)—Erase from the cursor position to the top of the screen.

ERASE TO END OF FRAME (^V) —Erase from the cursor position to the bottom of the screen.

ERASE TO BEGINNING OF LINE (^\ ^F)—Erase the line to the left of the cursor.

ERASE TO END OF LINE (^F)—Erase the line to the right of the cursor.

### 3.3.4 Rolls and Slides

ROLL DOWN NORTH EAST QUADRANT (^] ^^)—Roll the screen quadrant above and to the right of the cursor position down one line.

ROLL DOWN NORTH WEST QUADRANT (^] ^N)—Roll the screen quadrant above and to the left of the cursor position down one line.

ROLL DOWN SOUTH EAST QUADRANT (^] ^_)—Roll the screen quadrant below and to the right of the cursor position down one line.

ROLL DOWN SOUTH WEST QUADRANT (^] ^O)—Roll the screen quadrant below and to the left of the cursor position down one line.

ROLL UP NORTH EAST QUADRANT (^\ ^^)—Roll the screen quadrant above and to the right of the cursor position upwards by one line.

ROLL UP NORTH WEST QUADRANT (^\ ^N)—Roll the screen quadrant above and to the left of the cursor position upwards by one line.

ROLL UP SOUTH EAST QUADRANT (^\ ^_)—Roll the screen quadrant below and to the right of the cursor position upwards by one line.

ROLL UP SOUTH WEST QUADRANT (^\ ^O)—Roll the screen quadrant below and to the left of the cursor position upwards by one line.

SCROLL DOWN (^O)—Scroll the entire screen downwards by one line.

SCROLL UP (^N)—Scroll the entire screen upwards by one line.

SLIDE SCREEN LEFT (^] ^L)—Shift the entire screen to the left by one character position.

SLIDE SCREEN RIGHT (^] ^M)—Shift the entire screen to the right by one character position.

### 3.3.5 Inserts and Deletes

DELETE CHARACTER, LEFT (^] ^H)—Delete the character under the cursor, shifting the left portion of the line to the right by one character position.

DELETE CHARACTER, RIGHT (^\ ^H)—Delete the character under the cursor, shifting the right portion of the line to the left by one character position.

INSERT CHARACTER, LEFT (^] ^X [character])—Insert [character] into a line at the cursor position. The left portion of the line is shifted left by one character position.

INSERT CHARACTER, RIGHT (^\ ^X [character])—Insert [character] into a line at the cursor position. The right portion of the line is shifted right by one character position.

INSERT LINE, DOWN (^\ ^Y)—Insert a blank line at the current cursor position by rolling the lines below the cursor down.

INSERT LINE, UP (^Y)—Insert a blank line at the current cursor position by rolling the lines above the cursor upward.

DELETE LINE, DOWN (^\ ^Z)—Delete a line on the screen by rolling the lines above the cursor down.

DELETE LINE, UP (^Z)—Delete a line on the screen by rolling the lines below the cursor upward.

### 3.3.6 Configure Screen

SET CRT DISPLAY FORMAT I (^\ ^Q)—Configure the terminal to use an 82 character by 16 line screen with the standard character generator ROM.

SET CRT DISPLAY FORMAT II (^\ ^R)—Configure the terminal to use an 82 character by 20 line screen with the standard character generator ROM.

SET CRT DISPLAY FORMAT III (^\ ^S)—Configure the terminal to use an 82 character by 16 line screen with the alternate ROM character generator.

SET CRT DISPLAY FORMAT IV (^\ ^T)—Configure the terminal to use an 82 character by 20 line screen with the alternate ROM character generator.

SET GRAPHICS CRT DISPLAY FORMAT (^] ^V)—Configure the terminal to use a 92 character by 22 line screen with the pseudo-graphics character generator ROM.

### 3.3.7 Graphics

CLEAR GRAPHICS DOT (^] ^T [GX] [GY])—Turn off a pseudo-graphics pixel at the specified graphics position. [GX] specifies the horizontal position and [GY] specifies the vertical position of the pixel.

INVERT GRAPHICS DOT (^] ^U [GX] [GY])—Invert a pseudo-graphics pixel at the specified graphics position.

SET GRAPHICS DOT (^] ^S [GX] [GY])—Turn on a pseudo-graphics pixel at the specified graphics position.

CLEAR GRAPHICS LINE (^] ^D [GU] [GV] [GX] [GY])—While in graphics mode, clear a straight line between two points. [GU] specifies the horizontal position and [GV] specifies the vertical position of the first end point. [GX] specifies the horizontal position and [GY] specifies the vertical position of the second end point.

INVERT GRAPHICS LINE (^] ^E [GU] [GV] [GX] [GY])—While in graphics mode, invert a straight line between two points.

SET GRAPHICS LINE (^] ^C [GU] [GV] [GX] [GY])—While in graphics mode, draw a straight line between two points.

PITCH DOWN GRAPHICS SCREEN (^] ^I)—Roll the pseudo-graphics pixels on the screen downward by one pixel position.

PITCH UP GRAPHICS SCREEN (^] ^J)—Roll the pseudo-graphics pixels on the screen upward by one pixel position.

YAW LEFT GRAPHICS SCREEN (^] ^Y)—Shift the pseudo-graphics pixels on the screen left by one pixel position.

YAW RIGHT GRAPHICS SCREEN (^] ^Z)—Shift the pseudo-graphics pixels on the screen right by one pixel position.

### 3.3.8 Teletypewriter Functions

BELL (^G)—Give the operator an audible alert.

CARRIAGE RETURN (^M)—Return the cursor to the left edge of the screen. If the Auto Line Feed option is set, a Line Feed function is executed.

FORM FEED (^L)—Clear the screen. This function causes the equivalent of a Home Up function followed by an Erase to End of Frame function. (For details see Home Up and Erase to End of Frame.)

LINE FEED (^J)—Move the cursor to the next line of the screen. If the cursor is not on the bottom line of the screen, this function is treated like a Bump Cursor Down function. If the cursor is on the bottom line of the screen and the Scroll on Line Feed Option is set, the screen will scroll upward by one line.

LINE UNFEED (^\ ^J)—Move the cursor to the previous line of the screen. This function is the opposite of Line Feed.

### 3.3.9 Configure CR/LF

DISABLE AUTOMATIC LINE FEED ON CARRIAGE RETURN (^^ ^I)—Configure the terminal to not execute an automatic Line Feed function whenever a Carriage Return is executed.

ENABLE AUTOMATIC LINE FEED ON CARRIAGE RETURN (^^ ^Y)—Configure the terminal so that a Carriage Return will automatically execute a Line Feed function.

DISABLE AUTOMATIC NEW LINE ON OVERFLOW (^^ ^Z)—Configure the terminal so that no new line function is executed if data characters overflow the line to the right.

ENABLE AUTOMATIC NEW LINE ON OVERFLOW (^^ ^J)—Configure the terminal to execute a Carriage Return, Line Feed sequence whenever the rightmost column of the screen is written.

DISABLE SCROLL ON LINE FEED (^^ ^X)—Configure the terminal so that a Line Feed occuring on the bottom line of the screen does not execute a Scroll Up function.

ENABLE SCROLL ON LINE FEED (^^ ^H)—Configure the terminal to execute an automatic Scroll Up function when a Line Feed is executed with the cursor on the bottom line of the screen.

### 3.3.10 Character Control

CHANGE CONTROL CHARACTER (^] ^W [funtion byte] [control character])—Change the control function of any of the 32 primary control characters.

DISPLAY CONTROL CHARACTER (^\ ^@)—Selectively disable Group A control characters. This function is most useful when translated into Group A using the Change Control Character function. For example: If the only control functions you want executed by the terminal are Carriage Return and Line Feed, all other characters must be changed to execute the Display Control Character function. When this is done, only the Carriage Return and Line Feed functions will be executed. All other control characters will be displayed as data characters.

NULL (^@)—No operation (NOP) function for teletypewriter compatibility.

CLEAR CHARACTER PROTECT BIT (^] ^R [X] [Y])—Force the character at the specified screen position to assume unprotected (high intensity) status. [X] specifies the horizontal position and [Y] specifies the vertical position.

SET CHARACTER PROTECT BIT (^] ^Q [X] [Y])—Force the character at the specified screen position to assume protected (low intensity) status.

SET BACKGROUND MODE (^\ ^E)—Configure the terminal for background mode; that is, all characters written to the screen will be written as protected (low intensity) characters.

SET FOREGROUND MODE (^\ ^U)—Configure the terminal for foreground mode; that is, all characters written to the screen will be written as unprotected (high intensity) characters. In addition, protected (low intensity) characters may not be erased or overwritten.

DISABLE RUBOUT AS DATA (^^ ^[)—Configure the terminal so that a rubout (hex 7F) presented as a data character is ignored.

ENABLE RUBOUT AS DATA (^^ ^K)—Configure the terminal to treat a rubout (hex 7F) as a data character.

DISABLE ESCAPE CHARACTER (^^ ^@)—Escape characters (hex 1B) will be ignored by the terminal.

ENABLE ESCAPE CHARACTER (^^ ^P)—Escape characters (hex 1B) will invoke a Data Link Escape Control function. For details, see Data Link Escape.

HONOR CHARACTER PROTECTION (^^ ^W)—Configure the terminal to honor character protection during data entry, erase, and transmit functions.

IGNORE CHARACTER PROTECTION (^^ ^G)—Configure the terminal to ignore the protection status of characters displayed on the screen.

WRITE PROTECTED CHARACTERS (^^ ^V)—Configure the terminal so that all characters displayed will be write protected (low intensity).

WRITE UNPROTECTED CHARACTERS (^^ ^F)—Configure the terminal to write characters on the screen in unprotected (high intensity) format.

### 3.3.11 Device Control

DISABLE TERMINAL PRINTER PASSTHROUGH (^] ^G)—Disable the optional terminal printer. For details about the Terminal Printer Passthrough capability, see the documentation of Enable Terminal Printer Passthrough.

ENABLE TERMINAL PRINTER PASSTHROUGH (^] ^K)—When the printer is enabled, any data characters received by the terminal will be passed on to the printer by the terminal.

PUNCH OFF (^T)—Produce a strobe suitable for deactivating an external device. This strobe is available on pin 25 of the serial I/O connector on the back panel. (This is compatible to the SWTPC AC-30 interface.)

PUNCH ON (^R)—Produce a strobe suitable for activating an external device. This strobe is available on pin 11 of the serial I/O connector.

READER OFF (^S)—Produce a strobe suitable for deactivating an external device. This strobe is available on pin 18 of the serial I/O connector.

READER ON (^Q)—Produce a strobe suitable for activating an external device. This strobe is available on pin 10 of the serial I/O connector.

### 3.3.12 Configure Interfaces

ENABLE UPPER AND LOWER CASE (^^ ^L)—Configure the terminal keyboard to transmit both upper and lower case alphabetic characters. This option does not affect non-alphabetic characters.

SET UPPER CASE ONLY (^^ ^\ )—Configure the terminal keyboard so that alphabetic characters are forced to upper case.

SET CONVERSATIONAL MODE (^^ ^O)—Configure the terminal to conversational mode.

SET PAGED EDIT MODE (^^ ^_)—Configure the terminal to paged mode.

SET FULL DUPLEX (^^ ^N)—Configure the terminal to full duplex mode.

SET HALF DUPLEX (^^ ^^)—Configure the terminal to operate in the half duplex mode.

SET BAUD RATE (^] ^] ) [control character]—Reconfigure the terminal's baud rate.

ENABLE SHIFT INVERSION (^^ ^M)—Configure the terminal keyboard to invert the sense of the shift key for all alphabetic characters.

DISABLE SHIFT INVERSION (^^ ^] )—Disable shift key inversion on alphabetic characters.

### 3.3.13 Configure Modes

CLEAR ESCAPE DATA MODE (^^ ^A)—Turn the escape mode (control character display) off.

SET ESCAPE DATA MODE (^^ ^Q)—Control characters will be displayed on the screen before they are processed as functions by the terminal.

**CLEAR GRAPHICS CURSOR MODE (^^ ^B)**—Disable special cursor and scrolling functions used by CT-82 graphics control functions.

**SET GRAPHICS CURSOR MODE (^^ ^R)**—Configure the terminal for graphics mode scrolling and cursor function. This function is normally limited to internal use by the graphics functions.

### 3.3.14 Leadin Characters

**DATA LINK ESCAPE (^] ^P [character])**—[character] is treated as a seven-bit ASCII character and is displayed on the screen. If [character] is a control character, it is simply displayed without executing the associated control function.

**ESCAPE (^[ [character])**—[character] may be any 7-bit value, and will be treated as data and displayed on the screen, if the Enable Escape Character mode has been set.

**GROUP B CONTROL SEQUENCE (^\ followed by [control character])**—Provide a second group of control characters (Group B) for less commonly used control functions. A Group B control sequence consists of the Group B entry character followed by one of the Group B control characters. See the documentation of the Group B control functions for details.

**GROUP C CONTROL SEQUENCE (^] followed by [control character])**—Provide a third group of control characters (Group C) for less commonly used control functions. A Group C control sequence consists of the Group C entry character followed by one of the Group C control characters. See the documentation of the Group C control functions for details.

**OPTION BIT CONTROL (^^ followed by [control character])**—Provide a method for setting and clearing the 16 process option bits recognized by the CT-82 control ROM. For details, see the documentation of the Group D control functions.

**OPTION ROM CONTROL SEQUENCE (^_ followed by [control character])**—Provide a method for executing control functions provided in a possible Optional Control ROM. Individual functions are documented for the specific optional ROM installed.

**SET LEADIN CHARACTER (^\ ^[ [control character])**—Configure the terminal to require a control function Leadin Character. All control characters, unless preceded by the Leadin Character, will be treated as data characters.

### 3.3.15 Miscellaneous

**DISPLAY BYTE IN HEX (^] ^[ [byte])**—Display an 8-bit binary value in hexadecimal format.

**DISPLAY NUMBER IN DECIMAL (^] ^\ [MSP] [LSP])**—Display a 16-bit binary value in left-normalized, leading zero suppressed format.

**READ CURSOR POSITION (^] ^A)**—This function transmits the current cursor position as two characters, the first is the horizontal position and the second is the vertical position.

**READ LIGHT PEN POSITION (^] ^B)**—Transmit the screen position of the character touched by the Light Pen. (A light pen is not available from SWTPC.) The first character transmitted is the horizontal position and the second is the vertical position.

**TRANSMIT (SCREEN READ) (^W)**—Transmit the contents of the screen, from the upper left corner up to, but not including the current position of the cursor.

## Chapter 4. Principles of Operation

The following chapter is provided to give the user some background information that will prove useful when using some of the more flexible control functions.

### 4.1 Data Removed from the Screen

The CT-82 is not capable of transmitting any data from the screen except as described under the Screen Read control function. This means that any data that is not currently displayed on the screen is lost. Scrolling or sliding part or all a line off the screen forces all characters beyond the display boundaries to be lost.

### 4.2 Displaying Control Characters

There are three methods provided with the CT-82 to cause control characters to be displayed as data characters. The first method causes the control character to be displayed followed by the execution of the associated control function. The second method requires a leadin character followed by the control character to be displayed. The third method is used to disable the control function associated with the control character and cause that control character to be treated as a displayable data character.

The first method of displaying control characters described above involves changing one of the option flags. If the "Clear/Set Escape Data Mode" option flag is set, then a control character sent to the terminal will first be displayed exactly as if it were a data character and second the control function associated with that control character will be performed by the terminal. This is a useful debugging tool for analyzing the characters actually received by the terminal.

The second method of displaying control characters described above makes use of the Data Link Escape function. The control characters used to execute the Data Link Escape function are sent to the terminal followed by the control character that needs to be displayed as a data character. The control function associated with the control character is not executed.

The third method of displaying control characters described above provides a method for informing the terminal that specified control character(s) need to be treated as data character(s) and not control character(s). By using the Change Control Character function, the Display Control Character function can be associated with those control character(s) that need to be displayed.

### 4.3 High versus Low Intensity Characters

The CT-82's capability to display characters on the screen at two brightness levels can be used in two different ways. First it is possible to use the different intensities to simply highlight information on the screen. The second method is to treat the low intensity characters as protected characters.

Using the different intensities to highlight important information on the screen requires that attention be given to two of the option flags; the "Write Unprotected/Protected Characters" option flag and the "Ignore/Honor Character Protection" option flag. Whenever the user wishes to write characters at low intensity, he must set the "Write Unprotected /Protected Characters" option to Write Protected Characters. Similarly, if high intensity characters are desired this option must be set to Write Unprotected Characters. Since the user is not interested in whether the character is protected or unprotected, he must also set the Ignore/Honor Character Protection" option flag to Ignore Character Protection.

Writing programs for data entry is simplified a great deal by treating the low intensity characters as protected characters. For example, the "form" or information put on the screen requesting the data to be entered, can be put onto the screen as protected characters and all information typed in from the keyboard can be put there as unprotected characters. This allows the program to initialize the screen for the next entry by simply erasing the screen instead of re-displaying the form. Protected characters are not erased by the CT-82. Simply put the information on the screen by using the techniques described for highlighting above, then set the "Ignore/Honor Character Protection" option flag to Honor Character Protection.

### 4.4 Using the Change Control Character Function

Before this function can be utilized by the user, he must understand the difference between a control function and a control character. A control character is the character received by the terminal which is used to invoke a control function. For example, issuing a ØD to the terminal will invoke the Carriage Return control function.

With the Change Control Character function this association between control character and control function can be changed. It is possible to cause a ØD to force execution of some control function other than Carriage Return. The Change Control Character function is used to replace the existing association between control character and control function with a new association between that control character and a new control function.

### 4.5 Substituting the CT-82 for Another Terminal (Character Translation)

If the user wishes to subsititute the CT-82 in place of another terminal that is curently being used with his computer system, it is likely that the CT-82's Group A control functions are not the same as the control functions available on the terminal being replaced. This incompatibility can be handled by using the CT-82's Change Control Character function.

Example: A particular user wishes to use the CT-82 in place of his older terminal and he finds that the only incompatibility between the two terminals is the way in which their Set Cursor Position functions work. The computer system issues a CTRL K, followed by the vertical position, followed by the horizontal position when the cursor position needs to be set. When the old terminal receives this control character sequence, it does the cursor position properly. The problem is that when the CT-82 receives this control character sequence, it treats the vertical position as the horizontal position and vice versa.

To solve this problem the CT-82 needs to be reconfigured so that it will handle the horizontal and vertical positions properly. To do this, type the following sequence at the CT-82's keyboard: CTRL ], CTRL W, +, CTRL K. What this acccomplishes is to reconfigure the CT-82 so that the Set Cursor Position (Y,X) function is used instead of the Set Cursor Position (X,Y) function. After configuring the CT-82 in this manner, the CT-82 will work in place of the older terminal.

Any control function that operates differently on the terminal being replaced can be handled in a similar manner. Simply find the CT-82's equivalent control function and translate the CT-82's control character to execute that function.

Note: The translate table used by the CT-82 which allows this capability is reset every time the terminal is powered on. This means that every time the terminal is turned off, then turned back on again, the previous control character configuration is lost. All control character translates must be done again. This is best handled by changing initialization software executing on the computer to issue the translation commands to the CT-82, instead of typing in the control character sequences from the keyboard.

It may also be necessary to disable one or more of the CT-82's control characters. This may be required because the computer is issuing control characters that it expects the terminal to ignore. This can be accomplished by using the CT-82's Change Control Character function to translate to the Null function on those control characters which need to be ignored.

Another possiblity is that the terminal being replaced treats some of the control characters as data characters without an associated control function. This can be handled by using the Change Control Character function to associate the Display Control Character function with the control character that needs to be displayed.

### 4.6 Graphics Mode

A pseudo-graphics terminal, like the CT-82, remembers what is displayed on the screen as ASCII characters and uses a character generator to translate those characters into pixels. This is different from a true graphics terminal which does not have a character generator, but simply remembers whether each pixel is on or off, individually. The advantage of a pseudo-graphics terminal is that it uses the same hardware as a "normal" terminal. A true graphics terminal has better resolution, but pays for it with more expensive, special purpose hardware.

The pseudo-graphics capability implemented on the CT-82 makes use of a special graphics character generator ROM used as the alternate character set. This character set includes all of the special characters needed to generate pixels in each character position. In addition, it includes all of the ASCII character between 2Ø (hex) and 5F (hex). These characters include all of the upper case letters, the digits and most of the useful punctuation.

Note: To use the CT-82's graphics capability you must obtain the special graphics character generator ROM from SWTPC and be sure that it is plugged into the socket provided for the alternate character set.

Each of the graphics characters is divided into 6 pixels. A graphics character is 2 pixels wide and 3 pixels high. Since the graphics screen is 92 characters wide by 22 lines high, this yields a resolution of 184 pixels wide by 66 pixels high. Note that ASCII characters written to the screen take up an area equivalent to a 2 by 3 grid of pixels. These ASCII characters are written to character positions, not pixel positions. For example, if the following pixels on the screen are set: (80, 30), (81, 30) (80, 31), (81, 31), (80, 32) and (81, 32) then writing an ASCII character at position (40, 10) will overwrite the pixels.

When the CT-82 receives a graphics control command to modify a pixel position, the pixel position is converted into a character position and a character selector. The character already at that character position and the character selector are then used to select the graphics character appropriate to performing the graphics control function. This selected character is then written to the character position.

### 4.7 Timing Considerations

Each control function and data character that must be processed by the CT-82 requires a finite amount of time for the processing. Data characters and most of the control functions take very little time to execute while others, particularly those that must modify a large portion of the screen, require considerably more time to execute. Since the minimum amount of time between characters is fixed by the baud rate and since this time is conceivably much less than the amount of time required for the terminal to complete processing of some of the control functions, data may be lost.

The mechanism provided in the CT-82 to handle this problem is a buffer which holds all incoming characters until they can be processed. If the CT-82 must process a large number of functions that take excessive time, it is possible for this buffer to overflow. If the buffer gets filled, any characters received by the terminal will be lost until such time as more room becomes available in the buffer due to the processing of characters by the terminal.

A warning to those people already familiar with terminals and their timing problems is appropriate at this time. A common method for handling timing considerations is by sending an appropriate number of Null characters to allow time for the control function processing to complete. A perfect example of this method is the execution of the Carriage Return function by teletypewriter type terminals. The Carriage Return function typically takes 4 character times to complete, these 4 character times are usually filled by issuing a Carriage Return, Line Feed, Null and another Null. THIS METHOD DOES NOT WORK WITH BUFFERED DEVICES, including the CT-82. What happens is that the Null character takes up one additional character position in the buffer. Each additional Null character takes up one more position in the buffer. This implies that there is less room in the buffer to accept data characters, which means that for each Null character put into the buffer one additional data character will be lost. This accomplishes the opposite of what was desired.

If data is being lost due to the buffer overflowing, there are two methods that can be used to prevent the CT-82 from losing data. The first method is to reconfigure the terminal to use a lower baud rate. This increases the amount of time between characters, giving the terminal more time to process the control functions causing the problem. This method is undesirable in that it imposes a limit on the baud rate that can be used. The second method is to connect a terminal handshake line (DTR) to the interfaced computer. This line provides status information to the computer which force it to pause until the terminal is ready to accept more data. This mechanism

allows the terminal to be configured up to its maximum baud rate (38,400 baud) without loss of any data. Further details are given in Appendix L.

If the user is unable or unwilling to connect the handshake lines, 9600 baud is an upper limit on the baud rate. Unless the terminal is configured for graphics mode, it is highly unlikely that any data will be lost when the terminal is operating at 9600 baud. In a similar manner it is equally likely that data will be lost when the terminal is operating at any of the baud rates above 9600 baud without the use of the handshake lines. Since the buffer is much smaller when the terminal is configured for graphics mode, either a lower baud rate or the handshake lines must be used.

### 4.8 Paged Edit Mode—Screen Read—Block Transmit

Paged Edit Mode is a mode of operation provided with the CT-82 which allows the user to type an entire page of information onto the screen before it is transmitted to the computer. This mode of operation goes under many names, among them Paged Edit Mode, Screen Read and Block Transmit.

Paged Edit Mode works as follows. First the CT-82 must be configured for Paged Edit Mode by setting the "Conversation/Paged Edit Mode" option flag for Paged Edit Mode. This sets up the terminal so that the only way in which information will be transmitted to the computer is when the XMIT key is typed or when the computer issues a Screen Read control character to the terminal.

Next the user types any information onto the screen that needs to be transmitted to the computer. This information can be put onto the screen in any format and by any means available to the user. This means that cursor positioning, erases and any of the other control functions may be used to get the information onto the screen. Control characters may be issued to the terminal by using the CTRL key in conjunction with the other keys at the keyboard. The format of the information on the screen depends only on the format expected by the program executing in the interfaced computer.

When all of the information is on the screen in the desired format, the user should position the cursor to the lower right corner of the screen and type the XMIT key. CTRL W can be used in place of the XMIT key, if the XMIT key is not available. The Screen Read function transmits the information on the screen line by line starting with the upper left corner of the screen and continuing until encountering the cursor. Once transmission starts there is no way to interrupt the transmission process. It must continue until the cursor is encountered. The cursor position remains unchanged throughout the transmission process.

If the line contains any trailing blanks, a single Carriage Return character is transmitted instead of the trailing blanks. If there are no trailing blanks in the line, a Carriage Return character is used to separate the lines. The cursor is considered the end of the last line and a Carriage Return character is transmitted to indicate the end of the line. There are no Line Feed characters transmitted following the Carriage Return characters transmitted at the end of each line.

If there are any control characters displayed on the screen at the time that the XMIT key is typed, they will not be transmitted. Also, the user can decide whether or not protected (low intensity) characters are transmitted. Unprotected (high intenisty) characters are always transmitted. See Appendix A for additional information about the Transmit function.

### 4.9 Printer Passthrough

Printer passthrough is a feature provided in the CT-82 which allows a parallel interfaced printer to be connected to and controlled by the terminal. This ability can be used to produce a hard copy of the information received at the terminal. Also, if the terminal is at a physically different location than the interfaced computer, printer passthrough can be used to print information at the remote site.

Just as the video screen is treated as a display device, the printer becomes another display device when it is enabled. This means that any character received by the terminal, is not only displayed on the screen, but is printed as well.

Since many of the control characters used by the terminal are undefined for most printers, the CT-82 passes only a limited set of control characters to the printer. All data characters are passed to the printer. The control characters passed to the printer are: Carriage Return, Line Feed, Form Feed and Bell.

When the printer connected to the CT-82 is unable to accept characters as fast as the terminal, one of two procedures must be used to keep the printer from losing data. The first and recommended procedure is to connect the terminal's handshake line (DTR) to the interfaced computer. The second procedure is to lower the terminal's baud rate to the point where the printer can keep up with the terminal.

Note that the information displayed on the screen cannot be passed to the printer in the same manner as Block Transmit can transmit the information displayed on the screen to the interfaced computer.

The following is an example of a sequence of characters transmitted to the CT-82 and the actions associated with the characters. Printer passthrough is turned on by issuing a 1D (hex) followed by a ØB (hex) to the terminal. Printer passthrough is turned off by issuing a 1D (hex) followed by a Ø7 (hex) to the terminal.

| Character Sequence: | 1D ØB | ØC | 1D Ø7 | THIS WILL ONLY BE DISPLAYED | ØD | ØA |
|---|---|---|---|---|---|---|
| Action at Terminal: | Printer On | Form Feed | Printer Off | (Displayed at Terminal) THIS WILL ONLY BE DISPLAYED | Carriage Return | Line Feed |
| Action at Printer: | None | Form Feed | None | None | None | None |

| Character Sequence: | 1D ØB | THIS WILL BE BOTH PRINTED AND DISPLAYED | ØD | ØA | 1D Ø7 |
|---|---|---|---|---|---|
| Action at Terminal: | Printer On | (Displayed at Terminal) THIS WILL BE BOTH PRINTED AND DISPLAYED | Carriage Return | Line Feed | Printer Off |
| Action at Printer: | None | (Printed on Printer) THIS WILL BE BOTH PRINTED AND DISPLAYED | Carriage Return | Line Feed | None |

### 4.10 CURSOR ON Position (Cancel and Backspace)

The CURSOR ON position is intended for use in conjunction with the Backspace control function, the Cancel control function and the data character function described in Appendix B. All of these functions are most useful when displaying data entered from the keyboard.

When a character is entered from the keyboard and echoed to the terminal, it is displayed on the screen using the data character function. Since the data character function automatically bumps the cursor to the next character position, there is no need for cursor positioning to occur between data characters. The Backspace function's method of positioning the cursor is the inverse of the method used by the data character function to bump to the next character position.

In addition to repositioning the cursor to the previous character position, the Backspace function erases the character that was already in that position. This provides a simple method for the operator to correct a single character typing error. Unfortunately, this gives the operator the capability to erase more information than he has typed. Backspacing beyond the first character he typed would erase any information displayed to the left of the space reserved for data entry. If this occurs, he may lose data he did not intend to lose.

The mechanism provided in the CT-82 to prevent backspacing too far is to provide a boundary called the CURSOR ON position. If the cursor position is the same as the CURSOR ON position, the Backspace function will be ignored by the terminal. This means using Backspace will work until the most recent CURSOR ON position is encountered.

It is recommended that when implementing data entry programs which make use of the CT-82, that the following sequence be used. After the cursor has been moved to the position where data is to be entered, the cursor should be turned on. This establishes the CURSOR ON position. Characters should then be obtained from the keyboard and echoed to the screen. When a Carriage Return

is received from the keyboard, the cursor should be turned off and data entry terminated. Using this method allows Backspace to be used during data entry.

The Cancel control function is provided to simplify erasing all the way to the CURSOR ON position. If Backspace is typed repeatedly, the cursor will be backed until the CURSOR ON position is encountered. To save the operator the effort of typing Backspace so many times, the Cancel key can be typed to accomplish the same thing.

### 4.11 Default Configuration

The configuration of the CT-82 after power up depends on the CONFIGURE switch on the back panel and the DIP switch under the terminal's cover. If the CONFIGURE switch is in the PROGRAMMABLE position , the 8-bit DIP switch is used to select the baud rate, the screen format, etc. Setting the DIP switch is described elsewhere in this manual.

If the CONFIGURE switch is set to AUTO, the terminal is configured as if the DIP switches were set as follows:

Switch #0 = OFF
Switch #1 = ON
Switch #2 = ON
Switch #3 = OFF
Switch #4 = OFF
} Select 9600 baud.

Switch #5 = ON Select Conversational mode.
Switch #6 = ON Select Full Duplex mode.
Switch #7 = ON Select CRT Format I.

The option flags are all cleared as shown below (the cleared position is in bold face type).

**Disable**/Enable Escape Character
**Clear**/Set Escape Data Mode
**Clear**/Set Graphics Cursor Mode
Set **Blinking**/Non-Blinking Cursor
Set **Block**/Underline Cursor
**Display Cursor**/Suppress Cursor Display
Write **Unprotected**/Protected Characters
**Ignore**/Honor Character Protection
**Enable**/Disable Scroll on Line Feed
**Disable**/Enable Automatic Line Feed on Carriage Return
**Enable**/Disable Automatic New Line on Overflow
**Enable**/Disable Rubout as Data
**Enable Upper and Lower Case**/Set Upper Case Only
**Enable**/Disable Shift Inversion
Set **Full**/Half Duplex
Set **Conversational**/Paged Edit Mode

# APPENDIX A

## BACKSPACE–

Control Character sequence: Ø8 (hex)
8 (decimal)
^H (keyboard)

Function: Erase the last data character displayed.

SPECIAL CONSIDERATIONS:

– The four actions that may occur are:

1. A "normal" Backspace occurs as follows:
   A. The cursor is moved left one position, and
   B. The character at the new cursor position is replaced with a blank.
2. The Backspace backs past a protected field as follows. The cursor is moved left until:
   A. An unprotected character is encountered (this character is replaced with a blank), or
   B. The CURSOR ON position is encountered (the cursor position will remain at the CURSOR ON position and no further action is taken).
3. The character under the cursor is replaced with a blank.
4. The Backspace function is ignored.

– The following algorithm is used by Backspace:

STEP 1. Is the cursor currently positioned to the right margin? If so, go to Step 2. If not, go to Step 3.

STEP 2 Is the character under the cursor a blank? If so, go to Step 3. If not, go to Step 5.

STEP 3 Is the current cursor postion the same or to the left of the CURSOR ON position? If so, terminate Backspace processing. If not, go to step 4.

STEP 4 Move the cursor left one position (this does not change the CURSOR ON position).

STEP 5 Is the Retain Protected Fields option set? If so, go to Step 6. If not go to Step 7.

STEP 6 Is the character under the new cursor protected(low intensity)? If so, go to Step 3. If not, go to Step 7.

STEP 7 Replace the character under the cursor with a blank and terminate Backspace processing.

## BELL–

Control Character sequence: Ø7 (hex)
7 (decimal)
^G (keyboard)

Function: Give the operator an audible alert.

**BUMP CURSOR DOWN–**

Control Character sequence: Ø2 (hex)
2 (decimal)
^B (keyboard)

Function: Move the cursor one line downward on the screen.

SPECIAL CONSIDERATIONS:

- If the cursor is on the bottom line of the screen, this function is ignored.
- The cursor's horizontal position is unchanged.
- The CURSOR ON position is cleared.
- Bump cursor down is not the same function as line feed. For details, see the documentation of Line Feed.

**BUMP CURSOR LEFT–**

Control Character sequence: Ø4 (hex)
4 (decimal)
^D (keyboard)

Function: Move the cursor one position to the left on the screen.

SPECIAL CONSIDERATIONS:

- The CURSOR ON position is cleared.
- The cursor's vertical position is unchanged.
- If the cursor is at the left edge of the screen, this function is ignored.
- This function is not the same as the backspace function. For details see the documentation on Backspace.

**BUMP CURSOR RIGHT–**

Control Character sequence: Ø9 (hex)
9 (decimal)
^I (keyboard)

Function: Move the cursor one position to the right on the screen.

SPECIAL CONSIDERATIONS:

- If the cursor is at the right edge of the screen, this function is ignored.
- The cursor's vertical position is unchanged.
- The CURSOR ON position is cleared.

**BUMP CURSOR UP–**

Control Character sequence: Ø1 (hex)
1 (decimal)
^A (keyboard)

Function: Move the cursor one line upward on the screen.

SPECIAL CONSIDERATIONS:

- If the cursor is on the top line of the screen, this function is ignored.
- The cursor's horizontal position is unchanged.
- The CURSOR ON position is cleared.

**CANCEL–**

Control Character sequence: 18 (hex)
24 (decimal)
^X (keyboard)

Function: Erase to the left from the current cursor position to the CURSOR ON position.

SPECIAL CONSIDERATIONS:

- If the cursor is currently off, the CURSOR ON position is assumed to be at the left edge of the screen.
- This function uses the Backspace function to erase characters.
- This function terminates when Backspace forces the cursor position to equal the CURSOR ON position.
- The CURSOR ON position is unchanged.

**CARRIAGE RETURN–**

Control Character sequence: ØD (hex)
13 (decimal)
^M (keyboard)

Function: Return the cursor to the left edge of the screen.

SPECIAL CONSIDERATIONS:

- The cursor is moved to the left edge of the screen.
- If the Auto Line Feed option is set, a Line Feed function is executed.
- The CURSOR ON position is cleared.

**CHANGE CONTROL CHARACTER–**

LL. 1D 17 2B ØB, sets term to use 2B (set y,x) for OB set x,y)

Control Character sequence: 1D 17 [function byte] [control character] (hex)
29 23 [function byte] [control character] (decimal)
^] ^W [function byte] [control character] (keyboard)

Function: Change the control function of any of the 32 control characters.

SPECIAL CONSIDERATIONS:

- Only the 32 control characters may have their control function changed.
- Only the control functions in Group A, Group B and Group C may be used to replace the control functions already associated with a control character.
- Group D control functions may not be used to replace the control functions already associated with a control character.
- If the function byte has a value between ØØ and 1F(hex), the specified control character is associated with a Group A control function.
- If the function byte has a value between 2Ø and 3F (hex), the specified control character is associated with a Group B control function. 2Ø (hex) refers to the function associated with the ØØ (hex) control character in Group B, 21 refers to Ø1, 22 refers to Ø2 and so on.
- If the function byte has a value between 4Ø and 5F (hex), the specified control character is associated with a Group C control function. 30 (hex) refers to the function associated with the ØØ (hex) control character in Group C, 31 refers to Ø1, 32 refers to Ø2 and so on.

**CLEAR CHARACTER PROTECT BIT–**

Control Character sequence:

| | | | | |
|---|---|---|---|---|
| 1D | 12 | [X] | [Y] | (hex) |
| 29 | 18 | [X] | [Y] | (decimal) |
| ^] | ^R | [X] | [Y] | (keyboard) |

Function: Force the character under the cursor to assume unprotected (high intensity) status.

SPECIAL CONSIDERATIONS:

- If the Graphics Cursor Mode option is set, this function is ignored.
- The state of the Retain Protected Fields and Write Protected options are unchanged.
- X specifies the horizontal position and Y specifies the vertical position.
- X must be between zero and one less than the maximum number of characters per line on the screen (inclusive).
- Y must be between zero and one less than the maximum number of lines on the screen (inclusive).
- If X or Y exceeds the allowable range, it is set to the maximum permissible value. (For maximum permissible values, reference the documentation on the CRT display formats.)
- Line Ø specifies the top line of the screen.
- Column Ø specifies the leftmost character position on a line.
- The cursor position is unchanged.

**CLEAR ESCAPE DATA MODE–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | Ø1 | (hex) |
| 30 | 1 | (decimal) |
| ^^ | ^A | (keyboard) |

Function: Turn the escape mode (control character display) off.

**CLEAR GRAPHICS CURSOR MODE–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | Ø2 | (hex) |
| 30 | 2 | (decimal) |
| ^^ | ^B | (keyboard) |

Function: Disable special cursor and scrolling functions used by CT-82 graphics control functions.

**CLEAR GRAPHICS DOT –**

Control Character sequence:

| | | | | |
|---|---|---|---|---|
| 1D | 14 | [GX] | [GY] | (hex) |
| 29 | 20 | [GX] | [GY] | (decimal) |
| ^] | ^T | [GX] | [GY] | (keyboard) |

Function: Turn off a pseudo-graphics pixel at the specified graphics position.

SPECIAL CONSIDERATIONS:

- GX specifies the horizontal position and GY specifies the vertical position of the pixel.
- GX must be between zero and one less than the maximum number of pixels per row (inclusive). (A row refers to a row of pixels.)

- GY must be between zero and one less than the maximum number of rows (inclusive).
- If GX or GY exceeds the allowable range, it is set to the maximum permissible value. (For maximum permissible values, reference the documentation on setting the graphics CRT format.)
- Row Ø specifies the top row of pixels on the screen.
- Column Ø specifies the leftmost pixel position in a row.
- The cursor position is unchanged.

**CLEAR GRAPHICS LINE–**

Control Character sequence:

| | | | | | | |
|---|---|---|---|---|---|---|
| 1D | Ø4 | [GU] | [GV] | [GX] | [GY] | (hex) |
| 29 | 4 | [GU] | [GV] | [GX] | [GY] | (decimal) |
| ^] | ^D | [GU] | [GV] | [GX] | [GY] | (keyboard) |

Function: Turn off the pseudo-graphics pixels in a straight line between two points.

SPECIAL CONSIDERATIONS:

- GU specifies the horizontal position and GV specifies the vertical position of the pixel at the first end point.
- GX specifies the horizontal position and GY specifies the vertical position of the pixel at the second end point.
- GU and GX must be between zero and one less than the maximum number of pixels per row (inclusive). (A row refers to a row of pixels.)
- GV and GY must be between zero and one less than the maximum number of rows (inclusive).
- If GU, GV, GX or GY exceeds the allowable range, it is set to the maximum permissible value. (For maximum permissible values, reference the documentation on setting the graphics CRT format.)
- Row Ø specifies the top row of pixels on the screen.
- Column Ø specifies the leftmost pixel position in a row.
- The cursor position is unchanged.

**CURSOR OFF–**

Control Character sequence:

| | |
|---|---|
| Ø5 | (hex) |
| 5 | (decimal) |
| ^E | (keyboard) |

Function: Cause the cursor not to be displayed on the screen.

SPECIAL CONSIDERATIONS:

- The CURSOR ON position is cleared
- The cursor position is defined whether or not the cursor is currently being displayed.

**CURSOR ON–**

Control Character sequence–

| | |
|---|---|
| 15 | (hex) |
| 21 | (decimal) |
| ^U | (keyboard) |

Function: Cause the cursor to be displayed on the screen.

SPECIAL CONSIDERATIONS:

- The CURSOR ON position is set equal to the cursor position.
- The format of the displayed cursor depends on the current setting of the cursor select option bits. For a complete description of available formats, see option bits documentation.

## DATA LINK ESCAPE–

Control Character sequence:

| | | | |
|---|---|---|---|
| 1D | 1Ø | [character] | (hex) |
| 26 | 16 | [character] | (decimal) |
| ^] | ^P | [character] | (keyboard) |

Function: Force the [character] to be treated as a data character.

SPECIAL CONSIDERATION:

– [character] is treated as a seven-bit ASCII character and is displayed on the screen.

## DELETE CHARACTER, LEFT–

Control Character sequence:

| | | |
|---|---|---|
| 1D | Ø8 | (hex) |
| 29 | 8 | (decimal) |
| ^] | ^H | (keyboard) |

Function: Delete the character under the cursor, shifting all characters to the left of the cursor right one position.

SPECIAL CONSIDERATIONS:

– The shifted field contains the following characters:
  A. The character under the cursor.
  B. If the Retain Protected Fields option is clear, then all characters to the left of the cursor, or
  C. If the Retain Protected Fields option is set, then characters to the left of the cursor up to but not including the first protected (low intensity) character, or up to the end of the line, which ever occurs first.
– This field is shifted one character position to the left.
– The first character position in the shifted field is blanked.
– If the cursor is positioned over a protected (low intensity) character, and the Retain Protected Fields option is set, this function is ignored.
– The cursor position is unchanged.

## DELETE CHARACTER, RIGHT–

Control Character sequence:

| | | |
|---|---|---|
| 1C | Ø8 | (hex) |
| 28 | 8 | (decimal) |
| ^\ | ^H | (keyboard) |

Function: Delete the character under the cursor and shift the right portion of the line left.

SPECIAL CONSIDERATIONS:

– The shifted field contains the following characters:
  A. The character under the cursor.
  B. If the Retain Protected Fields option is clear, then all characters to the right of the cursor.
  C. If the Retain Protected Fields option is set, then characters to the right of the cursor up to but not including the first protected (low intensity) character, or up to the end of line, which ever occurs first.
– This field is shifted one character position to the left.
– The last character position in the shifted field is blanked.
– If the cursor is positioned over a protected (low intensity) character, and the Retain Protected Fields option is set, this function is ignored.
– The cursor position is unchanged.

## DELETE LINE, DOWN –

Control Character sequence:

| | | |
|---|---|---|
| 1C | 1A | (hex) |
| 28 | 26 | (decimal) |
| ^\ | ^Z | (keyboard) |

Function: Delete a line on the screen.

SPECIAL CONSIDERATIONS:

- The line containing the cursor is deleted.
- The lines above the line containing the cursor are moved down one line.
- A blank line is brought down as the new top line.
- The cursor position is unchanged.

## DELETE LINE, UP –

Control Character sequence:

| | |
|---|---|
| 1A | (hex) |
| 26 | (decimal) |
| ^Z | (keyboard) |

Function: Delete a line on the screen.

SPECIAL CONSIDERATIONS:

- The line containing the cursor is deleted.
- The lines beneath the line containing the cursor are moved up one line.
- A blank line is brought up as the new bottom line.
- The cursor position is unchanged.

## DISABLE AUTOMATIC LINE FEED ON CARRIAGE RETURN–

Control Character sequence:

| | | |
|---|---|---|
| 1E | Ø9 | (hex) |
| 30 | 9 | (decimal) |
| ^^ | ^I | (keyboard) |

Function: Configure the terminal to not execute an automatic line feed function whenever a carriage return is executed.

## DISABLE AUTOMATIC NEW LINE ON OVERFLOW–

Control Character sequence:

| | | |
|---|---|---|
| 1E | 1A | (hex) |
| 30 | 26 | (decimal) |
| ^^ | ^Z | (keyboard) |

Function: Configure the terminal so that no new line function is executed if data characters overflow the line to the right.

SPECIAL CONSIDERATION:

- When the line is filled, all new data characters will be displayed in the right-most position.

## DISABLE ESCAPE CHARACTER–

Control Character sequence:

| | | |
|---|---|---|
| 1E | ØØ | (hex) |
| 30 | 00 | (decimal) |
| ^^ | ^@ | (keyboard) |

Function: Disable the escape character.

SPECIAL CONSIDERATION:

- Escape characters, 1B (hex), will be ignored by the terminal.

**DISABLE RUBOUT AS DATA–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | 1B | (hex) |
| 30 | 27 | (decimal) |
| ^^ | ^[ | (keyboard) |

Function: Configure the terminal so that a rubout (hex 7F) presented as a data character is ignored.

**DISABLE SCROLL ON LINE FEED–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | 18 | (hex) |
| 30 | 24 | (decimal) |
| ^^ | ^X | (keyboard) |

Function: Configure the terminal so that a line feed occuring on the bottom line of the screen does not execute a Scroll Up function.

SPECIAL CONSIDERATION:

– A line feed when the cursor is on the bottom line of the screen is ignored.

**DISABLE SHIFT INVERSION–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | 1D | (hex) |
| 30 | 29 | (decimal) |
| ^^ | ^] | (keyboard) |

Function: Disable shift key inversion on alphabetic characters.

**DISABLE TERMINAL PRINTER PASSTHROUGH–**

Control Character sequence:

| | | |
|---|---|---|
| 1D | Ø7 | (hex) |
| 29 | 7 | (decimal) |
| ^] | ^G | (keyboard) |

Function: Disable the optional terminal printer.

SPECIAL CONSIDERATION:

– For details about the terminal printer passthrough capability, see the documentation of enable terminal printer passthrough.

**DISPLAY BYTE IN HEX–**

Control Character sequence:

| | | | |
|---|---|---|---|
| 1D | 1B | [byte] | (hex) |
| 29 | 27 | [byte] | (decimal) |
| ^] | ^[ | [byte] | (keyboard) |

Function: Display an 8-bit binary value in hexidecimal format.

SPECIAL CONSIDERATIONS:

– The byte is treated as an un-signed 8-bit binary value.
– The displayed byte will occupy two character positions on the screen.

**DISPLAY CONTROL CHARACTER–**

Control Character sequence:

| | | |
|---|---|---|
| 1C | ØØ | (hex) |
| 28 | 0 | (decimal) |
| ^\ | ^@ | (keyboard) |

Function: Selectively Disable Group A control characters.

SPECIAL CONSIDERATIONS:

- This function displays the last control character received by the terminal as a data character.
- This function is most useful when translated into Group A using the Change Control Character function. For example: If the only control functions you want executed by the terminal are Carriage Return and Line Feed, all other characters must be changed to execute the Display Control Character function. When this is done, only the Carriage Return and Line Feed functions will be executed. All other control characters will be displayed.
- When executed by a sequence of Control Characters, this function displays only the last character in the sequence. For example: the sequence, 1C ØØ, will display only the ØØ as the data character.

**DISPLAY CURSOR–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | Ø5 | (hex) |
| 30 | 5 | (decimal) |
| ^^ | ^E | (keyboard) |

Function: Cause the cursor to be displayed.

SPECIAL CONSIDERATION:

- This function differs from Cursor On in that the CURSOR ON position remains unchanged.

**DISPLAY NUMBER IN DECIMAL–**

Control Character sequence:

| | | | | |
|---|---|---|---|---|
| 1D | 1C | [MSP] | [LSP] | (hex) |
| 29 | 28 | [MSP] | [LSP] | (decimal) |
| ^] | ^\ | [MSP] | [LSP] | (keyboard) |

Function: Display a 16-bit binary value in left-normalized, leading zero supressed format.

SPECIAL CONSIDERATIONS:

- [MSP] and [LSP] specify an unsigned 16-bit binary number.
- The number is converted to decimal and displayed at the current cursor position.
- The number may occupy from one to six character positions on the screen.
- Output formats are determined according to the following rules:
  A. Number = 0, displayed as "0".
  B. 0 <number <10, displayed as "N".
  C. 9 <number <100, displayed as "NN".
  D. 99 <number <1000, displayed as "NNN".
  E. 999 <number <10000, displayed as "NNNN".
  F. 9999 <number <65536, displayed as "NN,NNN".

**ENABLE AUTOMATIC LINE FEED ON CARRIAGE RETURN–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | 19 | (hex) |
| 30 | 25 | (decimal) |
| ^^ | ^Y | (keyboard) |

Function: Configure the terminal so that a carriage return will automatically execute a Line Feed function.

**ENABLE AUTOMATIC NEW LINE ON OVERFLOW–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | ØA | (hex) |
| 30 | 10 | (decimal) |
| ^^ | ^J | (keyboard) |

Function: Configure the terminal to execute a carriage return line feed sequence whenever the rightmost column of the screen is written.

**ENABLE ESCAPE CHARACTER–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | 1Ø | (hex) |
| 30 | 16 | (decimal) |
| ^^ | ^P | (keyboard) |

Function: Enable the escape character.

SPECIAL CONSIDERATION:

– Escape characters will invoke a Data Link Escape control function. For details, see Data Link Escape.

**ENABLE RUBOUT AS DATA–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | ØB | (hex) |
| 30 | 11 | (decimal) |
| ^^ | ^K | (keyboard) |

Function: Configure the terminal to treat a rubout (hex 7F) as a data character.

**ENABLE SCROLL ON LINE FEED–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | Ø8 | (hex) |
| 30 | 8 | (decimal) |
| ^^ | ^H | (keyboard) |

Function: Configure the terminal to execute an automatic scroll up function if a line feed is executed when the cursor is on the bottom line of the screen.

SPECIAL CONSIDERATION:

– The Rollup on Linefeed Option is set.

**ENABLE SHIFT INVERSION–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | ØD | (hex) |
| 30 | 13 | (decimal) |
| ^^ | ^M | (keyboard) |

Function: Configure the terminal keyboard to invert the sense of the shift key for all alphabetic characters.

SPECIAL CONSIDERATIONS:

- Non-shifted alphabetic characters are transmitted as upper case.
- Shifted alphabetic characters are transmitted as lower case.
- Non-alphabetic characters are not affected.

## ENABLE TERMINAL PRINTER PASSTHROUGH–

Control Character sequence:

| | | |
|---|---|---|
| 1D | ØB | (hex) |
| 29 | 11 | (decimal) |
| ^] | ^K | (keyboard) |

Function: Enable the optional terminal printer.

SPECIAL CONSIDERATIONS:

- When the printer is enabled, any data characters received by the terminal will be passed on to the printer.
- Only the following control characters will be passed on to the printer.
  A. Carriage return.
  B. Line feed.
  C. Form feed.
  D. Bell.
- The printer interface consists of the following:
  A. An 8-bit data path.
  B. A data ready line and data accepted line for handkshake control between the terminal and the printer.
- It is recommended, when using the terminal printer passthrough feature, that the terminal's handshake lines (CTS & DTR) be connected. If these lines are not utilized, the terminal may lose characters since many printers operating speeds will be below that of the CT-82 terminal.

## ENABLE UPPER AND LOWER CASE–

Control Character sequence:

| | | |
|---|---|---|
| 1E | ØC | (hex) |
| 30 | 12 | (decimal) |
| ^^ | ^L | (keyboard) |

Function: Configure the terminal keyboard to transmit both upper and lower case alphabetic characters.

SPECIAL CONSIDERATION:

- This option does not affect non-alphabetic characters.

## ERASE TO BEGINNING OF FRAME–

Control Character sequence:

| | | |
|---|---|---|
| 1C | 16 | (hex) |
| 28 | 22 | (decimal) |
| ^\ | ^V | (keyboard) |

Function: Erase from the cursor position to the top of the screen.

SPECIAL CONSIDERATIONS:

- This function acts upon the following characters:
  A. The character under the cursor.
  B. All characters in the line to the left of the cursor, and
  C. All characters in all of the lines above the cursor.
- If the Retain Protected Fields Option is set, protected characters (low intensity) are bypassed.
- The cursor position is unchanged.

**ERASE TO BEGINNING OF LINE–**

Control Character sequence: 1C Ø6 (hex)
28 6 (decimal)
^\ ^F (keyboard)

Function: Erase the line to the left of the cursor.

SPECIAL CONSIDERATIONS:

- This function acts upon the following characters:
  A. The character under the cursor
  B. All characters in the line to the left of the cursor.
- If the Retain Protected Fields option is set, protected (low intensity) characters are bypassed.
- The cursor position is unchanged.

**ERASE TO END OF FRAME–**

Control Character sequence: 16 (hex)
22 (decimal)
^V (keyboard)

Function: Erase from the cursor postion to the bottom of the screen.

SPECIAL CONSIDERATIONS:

- This function acts upon the following characters:
  A. The character under the cursor.
  B. All characters in the line to the right of the cursor, and
  C. All characters in all of the lines below the cursor.
- If the Retain Protected Fields option is set, protected characters (low intensity) are bypassed.
- The cursor position is unchanged.

**ERASE TO END OF LINE–**

Control Character sequence: Ø6 (hex)
6 (decimal)
^F (keyboard)

Function: Erase the line to the right of the cursor.

SPECIAL CONSIDERATIONS:

- This function acts upon the following characters:
  A The character under the cursor, and
  B All characters in the line to the right of the cursor.
- If the Retain Protected Fields option is set, protected characters (low intensity) are bypassed.
- The cursor position is unchanged.

**ERASE FIELD–**

Control Character sequence: 1D Ø6 (hex)
29 6 (decimal)
^] ^F (keyboard)

Function: Erase the field to the right of the cursor.

SPECIAL CONSIDERATIONS:

- If the character under the cursor is protected (low intensity), the field contains the following characters:

A. The character under the cursor.
B. All characters to the right of the cursor up to but not including the first unprotected (high intensity) character, or up to the end of the line, which ever occurs first.

- If the character under the cursor is unprotected (high intensity), the field contains the following characters:
  A. The character under the cursor.
  B. All characters to the right of the cursor up to but not including the first protected (low intensity) character, or up to the end of the line, which ever occurs first.
- If the field contains protected characters and the Retain Protected Fields option is set, this function is ignored.

**ERASE NORTH EAST QUADRANT–**

Control Character sequence:

| | | |
|---|---|---|
| 1C | 1C | (hex) |
| 28 | 28 | (decimal) |
| ^\ | ^\ | (keyboard) |

Function: Erase the screen quadrant above and to the right of the cursor.

SPECIAL CONSIDERATIONS:

- This function acts upon all characters whose vertical position is less than or equal to that of the cursor, and whose horizontal position is greater than or equal to that of the cursor.
- If the Retain Protected Fields Option is set, protected (low intensity) characters are bypassed.
- The cursor position is unchanged.

**ERASE NORTH WEST QUADRANT–**

Control Character sequence:

| | | |
|---|---|---|
| 1C | ØC | (hex) |
| 28 | 12 | (decimal) |
| ^\ | ^L | (keyboard) |

Function: To erase the screen quadrant above and to the left of the cursor.

SPECIAL CONSIDERATIONS:

- This function acts upon all characters whose vertical position is less than or equal to that of the cursor, and whose horizontal position is less than or equal to that of the cursor.
- If the Retain Protected Fields option is set, protected (low intensity) characters are bypassed.
- The cursor position is unchanged.

**ERASE SOUTH EAST QUADRANT–**

Control Character sequence:

| | | |
|---|---|---|
| 1C | 1D | (hex) |
| 28 | 29 | (decimal) |
| ^\ | ^] | (keyboard) |

Function: Erase the screen quadrant below and to the right of the cursor.

SPECIAL CONSIDERATIONS:

- This function acts upon all characters whose vertical position is greater than or equal to that of the cursor, and whose horizontal position is greater than or equal to that of the cursor.

- If the Retain Protected Fields Option is set, protected (low intensity) characters are bypassed.
- The cursor position is unchanged.

ERASE SOUTH WEST QUADRANT—

Control Character sequence:

| | | |
|---|---|---|
| 1C | ØD | (hex) |
| 28 | 13 | (decimal) |
| ^\ | ^M | (keyboard) |

Function: To erase the screen quadrant below and to the left of the cursor.

SPECIAL CONSIDERATIONS:

- This function acts upon all characters whose vertical position is greater than or equal to that of the cursor, and whose horizontal position is less than or equal to that of the cursor.
- If the Retain Protected Fields option is set, protected (low intensity) characters are bypassed.
- The cursor position is unchanged.

ESCAPE—

Control Character sequence:

| | | | |
|---|---|---|---|
| 1B | followed by | [control character] | (hex) |
| 27 | followed by | [control character] | (decimal) |
| ^[ | followed by | [control character] | (keyboard) |

Function: Force the terminal to consider [control character] as a Data Character:

SPECIAL CONSIDERATIONS:

- If the Enable Escape Character option is not set, this function is ignored.
- The character following the Escape function may be any 7-bit value, and will be treated as data and displayed on the screen.

**FORM FEED–**

Control Character sequence: ØC (hex)
12 (decimal)
^L (keyboard)

Function: Clear the screen.

SPECIAL CONSIDERATIONS:

- This function causes the equivalent of a Home Up function followed by an Erase to End of Frame function. (For details, see Home Up and Erase To End of Frame.)

**GROUP B CONTROL SEQUENCE–**

Control Character sequence:
1C followed by [control character] (hex)
28 followed by [control character] (decimal)
•^\ followed by [control character] (keyboard)

Function: Provide a second group of control characters (GROUP B) for less commonly used control functions.

SPECIAL CONSIDERATIONS:

- A Group B Control Sequence consists of the Group B entry character followed by one of the Group B control characters.
- See documentation of Group B control functions for details.

**GROUP C CONTROL SEQUENCE–**

Control Character sequence:
1D followed by [control character] (hex)
29 followed by [control character] (decimal)
^] followed by [control character] (keyboard)

Function: Provide a third group of control characters (GROUP C) for less commonly used control functions.

SPECIAL CONSIDERATIONS:

- A Group C control sequence consists of the Group C entry character followed by one of the Group C control characters.
- See documentation of Group C control functions for details.

**HOME DOWN–**

Control Character sequence: Ø3 (hex)
3 (decimal)
^C (keyboard)

Function: Position the cursor in the lower left corner of the screen.

SPECIAL CONSIDERATIONS:

– The CURSOR ON position is cleared

**HOME DOWN TO RIGHT–**

Control Character sequence: 1C Ø3 (hex)
28 3 (decimal)
^\ ^C (keyboard)

Function: Move the cursor to the lower right corner of the screen.

SPECIAL CONSIDERATION:

– The CURSOR ON position is cleared.

**HOME UP–**

Control Character sequence: 1Ø (hex)
16 (decimal)
^P (keyboard)

Function: Position the cursor to the upper left hand corner of the screen.

SPECIAL CONSIDERATIONS:

– The CURSOR ON position is cleared.

**HOME UP TO RIGHT–**

Control Character sequence: 1C 1Ø (hex)
28 16 (decimal)
^\ ^P (keyboard)

Function: Set the cursor in the upper right corner of the screen.

SPECIAL CONSIDERATION:

– The CURSOR ON position is cleared.

**HONOR CHARACTER PROTECTION–**

Control Character sequence: 1E 17 (hex)
30 23 (decimal)
^^ ^W (keyboard)

Function: Configure the terminal to honor character protection during Data entry, erase, and transmit functions.

SPECIAL CONSIDERATION:

– If the Retain Protected Fields Option is set, protected (low intensity) characters will not be erased by Backspace or Erases.

## IGNORE CHARACTER PROTECTION–

Control Character sequence:

| | | |
|---|---|---|
| 1E | Ø7 | (hex) |
| 30 | 7 | (decimal) |
| ^^ | ^G | (keyboard) |

Function: Configure the terminal to ignore the protection status of characters displayed on the screen.

SPECIAL CONSIDERATIONS:

- The status of protected (low intensity) and unprotected (high intensity) characters remains unchanged.
- The Retain Protected Fields Option is cleared.

## INSERT CHARACTER, LEFT–

Control Character sequence:

| | | | |
|---|---|---|---|
| 1D | 18 | [character] | (hex) |
| 29 | 24 | [character] | (decimal) |
| ^] | ^X | [character] | (keyboard) |

Function: Insert a character into a field, shifting all characters to the left of the cursor left one character position.

SPECIAL CONSIDERATIONS:

- The shifted field contains the following characters.
  A. The character under the cursor.
  B. If the Retain Portected Fields option is clear, then all characters to the left of the cursor.
  C. If the Retain Protected Fields option is set, then characters to the left of the cursor up to but not including the first proted (low intensity) character, or up to the end of line, which ever occurs first.
- The field is shifted one character position to the left.
- The leftmost character in the field is lost.
- [character] is placed under the cursor.
- The cursor position is unchanged.
- If the cursor is positioned over a protected (low intensity) character, and the Retain Protected Fields option is set, this function is ignored.

## INSERT CHARACTER, RIGHT–

Control Character sequence:

| | | | |
|---|---|---|---|
| 1C | 18 | [character] | (hex) |
| 28 | 24 | [character] | (decimal) |
| ^\ | ^X | [character] | (keyboard) |

Function: Insert a character into a field, shifting all characters to the right of the cursor right one position.

SPECIAL CONSIDERATIONS:

- The shifted field contains the following characters:
  A. The character under the cursor.
  B. If the Retain Protected Fields Option is clear, then all characters to the right of the cursor.
  C. If the Retain Protected Fields Option is set, then characters to the right of the cursor up to but not including the first protected (low intensity) character, or up to the end of line, which ever occurs first.

- This field is shifted one character position to the right.
- The last character in the field is lost.
- [character] is placed under the cursor.
- The cursor is moved one position to the right.
- If the cursor is positioned over a protected (low intensity) character, and the Retain Protected Fields Option is set, the function is ignored.

**INSERT LINE, DOWN–**

Control Character sequence:

| | | |
|---|---|---|
| 1C | 19 | (hex) |
| 28 | 25 | (decimal) |
| ^\ | ^Y | (keyboard) |

Function: Insert a line at the current cursor position.

SPECIAL CONSIDERATIONS:

- The lines below the line containing the cursor are moved down.
- The line containing the cursor is moved down.
- The cursor position is unchanged.
- An empty line is inserted on the screen at the cursor position.
- The bottom line of the screen is lost.

**INSERT LINE, UP –**

Control Character sequence:

| | |
|---|---|
| 19 | (hex) |
| 25 | (decimal) |
| ^Y | (keyboard) |

Function: Insert an empty line at the current cursor postion.

SPECIAL CONSIDERATIONS:

- The top line of the screen is lost.
- The lines above the line containing the cursor are moved up one line.
- The line containing the cursor is moved up one line.
- An empty line is inserted on the screen at the cursor position.
- The cursor position is unchanged.

**INVERT GRAPHICS DOT–**

Control Character sequence:

| | | | | |
|---|---|---|---|---|
| 1D | 15 | [GX] | [GY] | (hex) |
| 29 | 21 | [GX] | [GY] | (decimal) |
| ^] | ^U | [GX] | [GY] | (keyboard) |

Function: Invert a pseudo-graphics pixel at the specified graphics position.

SPECIAL CONSIDERATIONS:

- GX specifies the horizontal position and GY specifies the vertical position of the pixel.
- GX must be between zero and one less than the maximum number of pixels per row (inclusive). (A row refers to a row of pixels.)
- GY must be between zero and one less than the maximum number of rows (inclusive).
- If GX or GY exceeds the allowable range, it is set to the maximum permissible value. (For maximum permissible values, reference the documentation on setting the graphics CRT format.)
- Row Ø specifies the top row of pixels on the screen.
- Column Ø specifies the leftmost pixel position in a row.
- The cursor position is unchanged.

**INVERT GRAPHICS LINE–**

| Control Character sequence: | 1D | Ø5 | [GU] | [GV] | [GX] | [GY] | (hex) |
|---|---|---|---|---|---|---|---|
| | 29 | 5 | [GU] | [GV] | [GX] | [GY] | (decimal) |
| | ^] | ^E | [GU] | [GV] | [GX] | [GY] | (keyboard) |

Function: Invert the pseudo-graphics pixels in a straight line between two points.

SPECIAL CONSIDERATIONS:

- GU specifies the horizontal position and GV specifies the vertical position of the pixel at the first end point.
- GX specifies the horizontal position and GY specifies the vertical position of the pixel at the second end point.
- GU and GX must be between zero and one less than the maximum number of pixels per row (inclusive). (A row refers to a row of pixels.)
- GV and GY must be between zero and one less than the maximum number of rows (inclusive).
- If GU, GV, GX or GY exceeds the allowable range, it is set to the maximum permissible value. (For maximum permissible values, reference the documentation on setting the graphix CRT format.)
- Row Ø specifies the top row of pixels on the screen.
- Column Ø specifies the leftmost pixel position in a row.
- The cursor position is unchanged.

**LINE FEED–**

| Control Character sequence: | ØA | (hex) |
|---|---|---|
| | 10 | (decimal) |
| | ^J | (keyboard) |

Function: Move the cursor to the next line of the screen.

SPECIAL CONSIDERATIONS:

- If the cursor is not on the bottom line of the screen, this function is treated like a bump cursor down function.
- If the cursor is on the bottom line of the screen and the rollup on the linefeed option is set. The screen will scroll upward by one line.
- The cursor's horizontal position is unchanged.
- The CURSOR ON position is cleared.

**LINE UNFEED–**

| Control Character sequence: | 1C | ØA | (hex) |
|---|---|---|---|
| | 28 | 10 | (decimal) |
| | ^\ | ^J | (keyboard) |

Function: Move the cursor to the previous line at the screen.

SPECIAL CONSIDERATIONS:

- If the cursor is not on the top line of the screen, this function is treated like a bump cursor up function.
- If the cursor is on the top line of the screen, and the rollup on linefeed option is set, the screen will scroll downward by one line.
- The cursor's horizontal position is unchanged.
- The CURSOR ON position is cleared.

MOVE CURSOR DOWN—

Control Character sequence:

| | | | |
|---|---|---|---|
| 1C | Ø2 | [count] | (hex) |
| 28 | 2 | [count] | (decimal) |
| ^\ | ^B | [count] | (keyboard) |

Function: Move the cursor downward on the screen by the number of lines specified in the count.

SPECIAL CONSIDERATIONS:

- If the count is zero, the cursor is not moved.
- If the specified count would result in the cursor being moved off the screen, the cursor is moved to the bottom line of the screen.
- The cursor's horizontal position is unchanged.
- The CURSOR ON position is cleared.

MOVE CURSOR LEFT—

Control Character sequence:

| | | | |
|---|---|---|---|
| 1C | Ø4 | [count] | (hex) |
| 28 | 4 | [count] | (decimal) |
| ^\ | ^D | [count] | (keyboard) |

Function: Move the cursor to the left by the amount specified in the count.

SPECIAL CONSIDERATIONS:

- If the count is zero, the cursor is not moved.
- If the specified count would result in moving the cursor off of the screen, the cursor is placed at the left edge of the screen.
- The cursor's vertical position is unchanged.
- The CURSOR ON position is cleared.

MOVE CURSOR RIGHT—

Control Character sequence:

| | | | |
|---|---|---|---|
| 1C | Ø9 | [count] | (hex) |
| 28 | 9 | [count] | (decimal) |
| ^\ | ^I | [count] | (keyboard) |

Function: Move the cursor to the right the number of positions specified in the count.

SPECIAL CONSIDERATIONS:

- If the count is zero, the cursor is not moved.
- If the specified count would result in moving the cursor off of the screen, the cursor is placed at the right edge of the screen.
- The cursor's vertical position is unchanged.
- The CURSOR ON position is cleared.

MOVE CURSOR UP—

Control Character sequence:

| | | | |
|---|---|---|---|
| 1C | Ø1 | [count] | (hex) |
| 28 | 1 | [count] | (decimal) |
| ^\ | ^A | [count] | (keyboard) |

Function: Move the cursor upward on the screen the number of lines specified in the count.

SPECIAL CONSIDERATIONS:

- If the count is zero, the cursor is not moved.
- If the specified count would result in the cursor being moved off the screen, the cursor is moved to the top line of the screen.
- The cursor's horizontal position is unchanged.
- The CURSOR ON position is cleared.

NULL–

Control Character sequence:

| | | |
|---|---|---|
| ØØ | (hex) |
| 0 | (decimal) |
| ^@ | (keyboard) |

Function: No operation (NOP) function for teletypewriter compatibility.

SPECIAL CONSIDERATIONS:

- This function is ignored by the terminal
- Transmitting NUL characters to the CT-82 as pad characters is ineffective. Timing considerations at speeds above 9600 baud must be handled by using the CTS and DTR modem control lines.

OPTION BIT CONTROL–

Control Character sequence:

| | | | |
|---|---|---|---|
| 1E | followed by | [control character] | (hex) |
| 30 | followed by | [control character] | (decimal) |
| ^^ | followed by | [control character] | (keyboard) |

Function: Provide a method for setting and clearing the 16 process option bits recognized by the CT-82 control ROM.

SPECIAL CONSIDERATIONS:

- The default configuration for all options is zero.
- For details, see documentation of the Group D control functions.

OPTION ROM CONTROL SEQUENCE–

Control Character sequence:

| | | | |
|---|---|---|---|
| 1F | followed by | [control character] | (hex) |
| 31 | followed by | [control character] | (decimal) |
| ^_ | followed by | [control character] | (keyboard) |

Function: Provide a method for executing control functions provided in the optional control ROM.

SPECIAL CONSIDERATIONS:

- This function provides a uniform entry method for control sequences decoded by the optional control ROM.
- This function is ignored if the optional ROM jumper is not installed.
- Individual functions are documented for the specific optional ROM installed.
- If the optional ROM jumper is installed, and the optional ROM is not installed, this function will produce unpredictable results.

**PITCH DOWN GRAPHICS SCREEN—**

Control Character sequence:

| | | |
|---|---|---|
| 1D | Ø9 | (hex) |
| 29 | 9 | (decimal) |
| ^] | ^I | (keyboard) |

Function: Roll the pseudo-graphics pixels on the screen downward by one pixel position.

SPECIAL CONSIDERATIONS:

- A blank row of pixels is brought down as the new top row.
- The bottom row of pixels on the screen is lost.
- The cursor position is unchanged.
- If the Retain Protected Fields option is set, ASCII characters on the screen are treated as protected and not moved.
- If the Retain Protected Fields option is clear, ASCII characters on the screen are replaced with blanks prior to moving the pseudo-graphics pixels.

**PITCH UP GRAPHICS SCREEN—**

Control Character sequence:

| | | |
|---|---|---|
| 1D | ØA | (hex) |
| 29 | 10 | (decimal) |
| ^] | ^J | (keyboard) |

Function: Roll the pseudo-graphics pixels on the screen upward by one pixel position.

SPECIAL CONSIDERATIONS:

- A blank row of pixels is brought up as the new bottom row.
- The top row of pixels on the screen is lost.
- The cursor position is unchanged.
- If the Retain Protected Fields option is set, ASCII characters on the screen are treated as protected and not moved.
- If the Retain Protected Field option is clear, ASCII characters on the screen are replaced with blanks prior to moving the pseudo-graphics pixels.

**PUNCH OFF—**

Control Character sequence:

| | |
|---|---|
| 14 | (hex) |
| 20 | (decimal) |
| ^T | (keyboard) |

Function: Produce a strobe suitable for deactivating an external device.

SPECIAL CONSIDERATIONS:

- The strobe is available on pin 5 of connector J5. (This is compatible to the SWTPC AC-30 interface.)
- The strobe is normally high and pulses low for at least ten microseconds.
- Devices should trigger on the trailing edge of the strobe.

**PUNCH ON—**

Control Character sequence:

| | |
|---|---|
| 12 | (hex) |
| 18 | (decimal) |
| ^R | (keyboard) |

Function: Produce a strobe suitable for activating an external device.

SPECIAL CONSIDERATIONS:

- The strobe is available on pin 8 of connector J5. (This is compatible to the SWTPC AC-30 interface.)
- The strobe is normally high and pulses low for at least ten microseconds.
- Devices should trigger on the trailing edge of the strobe.

## READ CURSOR POSITION–

Control Character sequence:

| | | |
|---|---|---|
| 1D | Ø1 | (hex) |
| 29 | 1 | (decimal) |
| ^] | ^A | (keyboard) |

Function: Transmit the current cursor position.

SPECIAL CONSIDERATIONS:

- This function transmits two characters, the first is the horizontal position and the second is the vertical position.
- Horizontal positions are between Ø and one less than the maximum number of characters per line (inclusive). For the maximum number of characters per line, reference the documentation on the CRT display formats.
- Vertical positions are between Ø and one less than the maximum number of lines (inclusive). For the maximum number of lines, reference the documentation on the CRT display formats.

## READ LIGHT PEN POSITION–

Control Character Sequence:

| | | |
|---|---|---|
| 1D | Ø2 | (hex) |
| 29 | 2 | (decimal) |
| ^] | ^B | (keyboard) |

Function: Transmit the screen position of the character touched by the light pen.

SPECIAL CONSIDERATIONS:

- This function transmits two characters, the first is the horizontal position and the second is the vertical position.
- If no light pen is attached, the light pen position is undefined. The two characters transmitted will have indeterminate values.
- Horizontal positions are between Ø and one less than the maximum number of characters per line (inclusive). For the maximum number of characters per line, reference the documentation on the CRT display formats.
- Vertical positions are between Ø and one less than the maximum number of lines (inclusive). For the maximum number of lines, reference the documentation on the CRT display formats.

**READER OFF–**

Control Character sequence: 13 (hex)
19 (decimal)
^S (keyboard)

Function: Produce a strobe suitable for deactivating an external device.

SPECIAL CONSIDERATIONS:

- The strobe is available on pin 4 of connector J5. (This is compatible with the SWTPC AC-30 interface.)
- The strobe is normally high and goes low for at least ten microseconds.
- Devices should trigger on the trailing edge of the strobe.

**READER ON–**

Control Character sequence: 11 (hex)
17 (decimal)
^Q (keyboard)

Function: Produce a strobe suitable for activating an external device.

SPECIAL CONSIDERATIONS:

- The strobe is available on pin 7 of connector J5. (This is compatible with the SWTPC AC-30 interface).
- The strobe is normally high and pulses low for at least ten microseconds.
- Devices should trigger on the trailing edge of the strobe.

**ROLL DOWN NORTH EAST QUADRANT–**

Control Character sequence: 1D 1E (hex)
29 30 (decimal)
^] ^^ (keyboard)

Function: Roll the screen quandrant above and to the right of the cursor down one line.

SPECIAL CONSIDERATIONS:

- This function acts on all characters whose vertical position is less than or equal to that of the cursor, and whose horizontal position is greater than or equal to that of the cursor.
- The bottom line of the affected area is lost.
- The other lines of the affected area are rolled down one line.
- A blank line is brought down as the new top line.
- The cursor position is unchanged.

**ROLL DOWN NORTH WEST QUADRANT–**

Control Character sequence: 1D ØE (hex)
29 14 (decimal)
^] ^N (keyboard)

Function: To roll the screen quadrant above and to the left of the cursor down one line.

SPECIAL CONSIDERATIONS:

- This function acts on all characters whose vertical position is less than or equal to that of the cursor, and whose horizontal position is less than or equal to that of the cursor.

- The bottom line of the affected area is lost.
- The other lines of the affected area are rolled down one line.
- a blank line is brought down as the new top line.
- The cursor position is unchanged.

### ROLL DOWN SOUTH EAST QUADRANT–

Control Character sequence:

| | | |
|---|---|---|
| 1D | 1F | (hex) |
| 29 | 31 | (decimal) |
| ^] | ^_ | (keyboard) |

Function: Roll the screen quadrant below and to the right of the cursor position down one line.

SPECIAL CONSIDERATIONS:

- This function acts on all characters whose vertical position is greater than or equal to that of the cursor, and whose horizontal position is greater than or equal to that of the cursor.
- The bottom line of the affected area is lost.
- The other lines of the affected area are rolled down one line.
- A blank line is brought down as the new top line.
- The cursor position is unchanged.

### ROLL DOWN SOUTH WEST QUADRANT–

Control Character sequence:

| | | |
|---|---|---|
| 1D | ØF | (hex) |
| 29 | 15 | (decimal) |
| ^] | ^O | (keyboard) |

Function: To roll the screen quadrant below and to the left of the cursor position down one line.

SPECIAL CONSIDERATIONS:

- This function acts on all characters whose vertical position is greater than or equal to that of the cursor, and whose horizontal position is less than or equal to that of the cursor.
- The bottom line of the affected area is lost.
- The other lines of the affected area are rolled down one line.
- A blank line is brought down as the new top line.
- The cursor position remains unchanged.

### ROLL UP NORTH EAST QUADRANT–

Control Character sequence:

| | | |
|---|---|---|
| 1C | 1E | (hex) |
| 28 | 30 | (decimal) |
| ^\ | ^^ | (keyboard) |

Function: Roll the screen quadrant above and to the right of the cursor upwards by one line.

SPECIAL CONSIDERATIONS:

- This function acts upon all characters whose vertical position is less than or equal to that of the cursor, and whose horizontal position is greater than or equal to that of the cursor.
- The top line of the affected area is lost.
- The other lines in the affected area are rolled upwards by one line.

- A blank line is brought up from the bottom and is made the last line of the affected area.
- The cursor position is unchanged.

### ROLL UP NORTH WEST QUADRANT–

Control Character sequence:

| | | |
|---|---|---|
| 1C | ØE | (hex) |
| 28 | 14 | (decimal) |
| ^\ | ^N | (keyboard) |

Function: To roll the screen quadrant above and to the left of the cursor upward by one line.

SPECIAL CONSIDERATIONS:

- This function acts upon all characters whose vertical position is less than or equal to that of the cursor, and whose horizontal position is less than or equal to that of the cursor.
- The top line of the affected area is lost.
- All other lines of the affected area are rolled up one line.
- A blank line is brought up from the bottom and is made the last line of the affected area.
- The cursor position is unchanged.

### ROLL UP SOUTH EAST QUADRANT–

Control Character sequence:

| | | |
|---|---|---|
| 1C | 1F | (hex) |
| 28 | 31 | (decimal) |
| ^\ | ^_ | (keyboard) |

Function: To roll the screen quadrant below and to the right of the cursor down by one line.

SPECIAL CONSIDERATIONS:

- This function acts upon all characters whose vertical position is greater than or equal to that of the cursor, and whose horizontal position is greater than or equal to that of the cursor.
- The top line of the affected area is lost.
- The other lines in the affected area are rolled up one line.
- The bottom line of the affected area is erased.
- A blank line is brought up from the bottom and is made the last line of the affected area.
- The cursor position is unchanged.

### ROLL UP SOUTH WEST QUADRANT–

Control Character sequence:

| | | |
|---|---|---|
| 1C | ØF | (hex) |
| 28 | 15 | (decimal) |
| ^\ | ^O | (keyboard) |

Function: To roll the screen quadrant below and to the left of the cursor upward by one line.

SPECIAL CONSIDERATIONS:

- This function acts upon all characters whose vertical position is greater than or equal to that of the cursor, and whose horizontal position is less than or equal to that of the cursor.

- The top line of the affected area is lost.
- All other lines of the affected area are rolled up by one line.
- A blank line is brought up from the bottom and is made the last line of the affected area.
- The cursor position is unchanged.

**SCROLL DOWN–**

Control Character sequence: ØF (hex)
15 (decimal)
^O (keyboard)

Function: Scroll the entire screen downwards by one line.

SPECIAL CONSIDERATIONS:

- A blank line is brought down as the new top line.
- The CURSOR ON position is cleared.
- The cursor position is unchanged.

**SCROLL UP–**

Control Character sequence: ØE (hex)
14 (decimal)
^N (keyboard)

Function: Scroll the entire screen upwards by one line.

SPECIAL CONSIDERATIONS:

- A blank line is brought up as the new bottom line.
- The CURSOR ON position is cleared.
- The cursor position is unchanged.

**SET BACKGROUND MODE–**

Control Character sequence: 1C Ø5 (hex)
28 5 (decimal)
^\ ^E (keyboard)

Function: Configure the terminal to ignore character protection and to write protected (low intensity) characters.

SPECIAL CONSIDERATIONS:

- All characters (including blanks) written to the screen while in background mode are written as protected (low intensity) characters.
- This function clears the Retain Protected Fields option so that protected (low intensity) characters may be erased or overwitten.

## SET BAUD RATE–

Control Character sequence:

| | | |
|---|---|---|
| 1D | 1D | [control character] (hex) |
| 29 | 29 | [control character] (decimal) |
| ^] | ^] | [control character] (keyboard) |

Function: Reconfigure the termianl to use a selected baud rate.

SPECIAL CONSIDERATION:

– The baud rates selected by the control character conform to the following table:

| Control Character | Baud Rate | Control Character | Baud Rate |
|---|---|---|---|
| ^@ – 00 | 50 | ^P – 10 | 1800 |
| ^A – 01 | 60 | ^Q – 11 | 2000 |
| ^B – 02 | 75 | ^R – 12 | 2400 |
| ^C – 03 | 100 | ^S – 13 | 3000 |
| ^D – 04 | 110 | ^T – 14 | 3600 |
| ^E – 05 | 120 | ^U – 15 | 4000 |
| ^F – 06 | 134.5 | ^V – 16 | 4800 |
| ^G – 07 | 150 | ^W – 17 | 6000 |
| ^H – 08 | 200 | ^X – 18 | 7200 |
| ^I – 09 | 240 | ^Y – 19 | 9600 |
| ^J – 0A | 300 | ^Z – 1A | 12,800 |
| ^K – 0B | 400 | ^[ – 1B | 14,400 |
| ^L – 0C | 450 | ^\ – 1C | 19,200 |
| ^M – 0D | 600 | ^] – 1D | 23,040 |
| ^N – 0E | 1000 | ^^ – 1E | 28,800 |
| ^O – 0F | 1200 | ^_ – 1F | 38,400 |

## SET BLINKING CURSOR –

Control Character sequence:

| | | |
|---|---|---|
| 1E | Ø3 | (hex) |
| 30 | 3 | (decimal) |
| ^^ | ^C | (keyboard) |

Function: Configure the cursor to blink when it is displayed.

## SET BLOCK CURSOR –

Control Character sequence:

| | | |
|---|---|---|
| 1E | Ø4 | (hex) |
| 30 | 4 | (decimal) |
| ^^ | ^D | (keyboard) |

Function: Configure the cursor to display as a block, as opposed to an underline.

## SET CHARACTER PROTECT BIT–

Control Character sequence:

| | | | | |
|---|---|---|---|---|
| 1D | 11 | [X] | [Y] | (hex) |
| 29 | 17 | [X] | [Y] | (decimal) |
| ^] | ^Q | [X] | [Y] | (keyboard) |

Function: Force the character under the cursor to assume protected (low intensity) status.

SPECIAL CONSIDERATIONS:

– If the Graphics Cursor Mode option is set, this function is ignored.
– The state of the Retain Protected Fields and Write Protected options are unchanged.

- X specifies the horizontal position and Y specifies the vertical position.
- X must be between zero and one less than the maximum number of characters per line on the screen (inclusive).
- Y must be between zero and one less than the maximum number of lines on the screen (inclusive).
- If X or Y exceeds the allowable range, it is set to the maximum permissible value. (For maximum permissible values, reference the documentation on the CRT display formats.)
- Line Ø specifies the top of the screen.
- Column Ø specifies the leftmost character position on a line.
- The cursor position is unchanged.

**SET CRT FORMAT I–**

Control Character sequence:

1C 11 (hex)
28 17 (decimal)
^\ ^Q (keyboard)

Function: Configure the terminal to use an 82-character by 16-line screen with the standard character generator ROM.

SPECIAL CONSIDERATIONS:

- The configuration process consists of:
  A. The graphics cursor mode is disabled.
  B. The Retain Protected Fields option is cleared.
  C. The CRT controller is configured. (This causes the equivalent of a Home Up control function.)
  D. An Erase End of Frame Control function is executed.

**SET CRT FORMAT II–**

Control Character sequence:

1C 12 (hex)
28 18 (decimal)
^\ ^R (keyboard)

Function: Configure the terminal to use an 82-character by 20-line screen with the standard character generator ROM.

SPECIAL CONSIDERATIONS:

- The configuration process consists of:
  A. The graphics cursor mode is disabled.
  B. The Retain Protected Fields option is cleared.
  C. The CRT controller is configured. (This causes the equivalent of a Home Up control function.)
  D. An Erase End of Frame control function is executed.

**SET CRT FORMAT III–**

Control Character sequence:

1C 13 (hex)
28 19 (decimal)
^\ ^S (keyboard)

Function: Configure the terminal to use an 82-character by 16-line screen with the alternate PROM character generator.

SPECIAL CONSIDERATIONS:

- The configuration process consists of:
  A. The graphics cursor mode is disabled.
  B. The Retain Protected Fields Option is cleared.

C. The CRT controller is configured. (This causes the equivalent of a Home Up control function.)
D. An Erase End of Frame control function is executed.

**SET CRT FORMAT IV–**

Control Character sequence:

| | | |
|---|---|---|
| 1C | 14 | (hex) |
| 28 | 20 | (decimal) |
| ^\ | ^T | (keyboard) |

Function: Configure the terminal to use an 82-character by 20-line screen with the alternate PROM character generator.

SPECIAL CONSIDERATIONS:

– The configuration process consists of:
  A. The graphics cursor mode is disabled.
  B. The Retain Protected Fields Option is cleared.
  C. The CRT controller is configured. (This causes the equivalent of a Home Up control function.)
  D. An Erase End of Frame control function is executed.

**SET FOR CONVERSATIONAL MODE–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | ØF | (hex) |
| 30 | 15 | (decimal) |
| ^^ | ^O | (keyboard) |

Function: Configure the terminal to conversational mode.

**SET FULL DUPLEX–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | ØE | (hex) |
| 30 | 14 | (decimal) |
| ^^ | ^N | (keyboard) |

Function: Configure the terminal to full duplex mode.

SPECIAL CONSIDERATIONS:

– Characters are output through the interface but are not displayed unless echoed back by the interfaced device.

**SET CURSOR POSITION (X, Y)–**

Control Character sequence:

| | | | |
|---|---|---|---|
| ØB | [X] | [Y] | (hex) |
| 11 | [X] | [Y] | (decimal) |
| ^K | [X] | [Y] | (keyboard) |

Function: Position the cursor at a specified vertical and horizontal position.

SPECIAL CONSIDERATIONS:

– X specifies the horizontal position and Y specifies the vertical position.
– X must be between zero and one less than the maximum number of characters per line on the screen (inclusive).
– Y must be between zero and one less than the maximum number of lines on the screen (inclusive).

- If X or Y exceeds the allowable range, it is set to the maximum permissible value. (For maximum permissible values, reference the documentation on the CRT display formats.)
- Line Ø specifies the top line of the screen.
- Column Ø specifies the leftmost character position on a line.
- The CURSOR ON position is cleared.

**SET CURSOR POSITION (Y, X)–**

Control Character sequence:

| | | | | |
|---|---|---|---|---|
| 1C | ØB | [Y] | [X] | (hex) |
| 28 | 11 | [Y] | [X] | (decimal) |
| ^\ | ^K | [Y] | [X] | (keyboard) |

Function: Position of the cursor at a specified vertical and horizontal position.

SPECIAL CONSIDERATIONS:

- Y specifies the vertical and X specifies the horizontal position.
- Y must be between zero and one less than the maximum number of lines (inclusive).
- X must be between zero and one less than the maximum number of characters per line (inclusive).
- If Y or X exceed the allowable range, it is set to the maximum permissible value. (For maximum permissible values, reference the documentation on the CRT Display formats.)
- Line Ø specifies the top line of the screen.
- Column Ø specifies the leftmost character position on a line.
- The CURSOR ON position is cleared.

**SET ESCAPE DATA MODE–**

Control Character sequence:

| | | |
|---|---|---|
| 1E | 11 | (hex) |
| 30 | 17 | (decimal) |
| ^^ | ^Q | (keyboard) |

Function: Turn on the Escape Data Mode (control character display).

SPECIAL CONSIDERATIONS:

- Control characters will be displayed on the screen before they are processed as functions by the control program.
- Functions depending rigorously on cursor position may behave in an unexpected manner. For example, Backspace will behave peculiarly if Escape Data Mode is set.

**SET FOREGROUND MODE–**

Control Character sequence:

| | | |
|---|---|---|
| 1C | 15 | (hex) |
| 28 | 21 | (decimal) |
| ^\ | ^U | (keyboard) |

Function: Configure the terminal to honor character protection, and to write unprotected (high intensity) characters.

SPECIAL CONSIDERATIONS:

- All characters written to the screen while in Foreground Mode are written as unprotected (high intensity) characters.
- This function sets the Retain Protected Field Option so that protected (low intensity) characters may not be erased or overwritten.

## SET GRAPHICS CRT DISPLAY FORMAT–

Control Character sequence:

| | | |
|---|---|---|
| 1D | 16 | (hex) |
| 29 | 22 | (decimal) |
| ^] | ^V | (keyboard) |

Function: Configure the terminal to use a 92 character by 22 line screen with the pseudo-graphics character generator ROM.

SPECIAL CONSIDERATIONS:

- The configuration process consists of:
  A. The graphics cursor mode is enabled.
  B The Retain Protected Fields option is cleared.
  C. The CRT controller is configured. (This causes the equivalent of a home up control function.
  D. An Erase End of Frame control function is executed.
  E. The graphics control functions are enabled.
- Since a character consists of 3 rows of 2 pixels each, the maximum number of rows is 66 and the maximum number of pixels per row is 184.
- If graphics CRT format is selected and the optional pseudo-graphics character generator ROM is not installed, the results will be unpredictable.

## SET GRAPHICS CURSOR MODE–

Control Character sequence:

| | | |
|---|---|---|
| 1E | 12 | (hex) |
| 30 | 18 | (decimal) |
| ^^ | ^R | (keyboard) |

Function: Configure the terminal for graphics mode scrolling and cursor functions.

SPECIAL CONSIDERATION:

- This mode is normally set by the Set Graphics Mode control sequence.

## SET GRAPHICS DOT–

Control Character sequence:

| | | | | |
|---|---|---|---|---|
| 1D | 13 | [GX] | [GY] | (hex) |
| 29 | 19 | [GX] | [GY] | (decimal) |
| ^] | ^S | [GX] | [GY] | (keyboard) |

Function: Turn on a pseudo-graphics pixel at the specified graphics position.

SPECIAL CONSIDERATIONS:

- GX specifies the horizontal position and GY specifies the vertical position of the pixel.
- GX must be between zero and one less than the maximum number of pixels per row (inclusive). (A row refers to a row of pixels.)
- GY must be between zero and one less than the maximum number of rows (inclusive).
- If GX or GY exceeds the allowable range, it is set to the maximum permissible value. (For maximum permissible values, reference the documentation on setting the graphics CRT format.)
- Row Ø specifies the top row of pixels on the screen.
- Column Ø specifies the leftmost pixel position in a row.
- The cursor position is unchanged.

SET GRAPHICS LINE—

Control Character sequence:

| | | | | | | |
|---|---|---|---|---|---|---|
| 1D | Ø3 | [GU] | [GV] | [GX] | [GY] | (hex) |
| 29 | 3 | [GU] | [GV] | [GX] | [GY] | (decimal) |
| ^] | ^C | [GU] | [GV] | [GX] | [GY] | (keyboard) |

Function: Turn on the pseudo-graphics pixels in a straight line between two points.

SPECIAL CONSIDERATIONS:

- GU specifies the horizontal position and GV specifies the vertical position of the pixel at the first end point.
- GX specifies the horizontal position and GY specifies the vertical position of the pixel at the second end point.
- GU and GX must be between zero and one less than the maximum number of pixels per row (inclusive). (A row refers to a row of pixels.)
- GV and GY must be between zero and one less than the maximum number of rows (inclusive).
- If GY, GV, GX or GY exceeds the allowable range, it is set to the maximum permissible value. (For maximum permissible values, reference the documentation on setting the graphic CRT format.)
- Row Ø specifies the top row of pixels on the screen.
- Column Ø specifies the leftmost pixel position in a row.
- The cursor position is unchanged.

SET HALF DUPLEX—

Control Character sequence:

| | | |
|---|---|---|
| 1E | 1E | (hex) |
| 30 | 30 | (decimal) |
| ^^ | ^^ | (keyboard) |

Function: Configure the terminal to operate in the half duplex mode.

SPECIAL CONSIDERATIONS:

- Data characters are displayed on the screen and output to the interfaced device.
- Control characters cause the appropriate control function to be executed in addition to being output to the interfaced device.

SET HORIZONTAL CURSOR POSITION—

Control Character Sequence:

| | | | |
|---|---|---|---|
| 1C | 17 | [X] | (hex) |
| 28 | 23 | [X] | (decimal) |
| ^\ | ^W | [X] | (keyboard) |

Function: Set the horizontal cursor position to the specified value.

SPECIAL CONSIDERATIONS:

- X specifies the horizontal position.
- X must be between zero and one less than the maximum number of characters per line.
- If X exceeds the allowable range, it is set to the maximum permissible value. (For maximum permissible values, reference the documentation on the CRT display formats.)
- Column Ø specifies the leftmost character position on a line.
- The cursor's vertical position is unchanged.
- The CURSOR ON position is cleared.

SET LEADIN CHARACTER–

Control Character sequence:

| | | | |
|---|---|---|---|
| 1C | 1B | [control character] | (hex) |
| 28 | 27 | [control character] | (decimal) |
| ^\ | ^[ | [control character] | (keyboard) |

Function: Configure the terminal to require a control function leadin character.

SPECIAL CONSIDERATIONS:

- Setting [control character] to NUL (zero) indicates that no Leadin Character is required.
- If [control character] is non-zero, no control sequence will be recognized unless preceded by this Leadin Character.
- Any control characters not preceded by the Leadin Character are treated as data characters.

SET NON-BLINKING CURSOR–

Control Character sequence:

| | | |
|---|---|---|
| 1E | 13 | (hex) |
| 30 | 19 | (decimal) |
| ^^ | ^S | (keyboard) |

Function: Configure the terminal to use a non-blinking cursor.

SPECIAL CONSIDERATIONS:

- The cursor position is unchanged.
- The CURSOR ON position is unchanged.

SET PAGED EDIT MODE–

Control Character sequence:

| | | |
|---|---|---|
| 1E | 1F | (hex) |
| 30 | 31 | (decimal) |
| ^^ | ^_ | (keyboard) |

Funciton: Configure the terminal to paged mode.

SPECIAL CONSIDERATIONS:

- Characters typed in appear on the screen.
- No transmission is made from the terminal until one of two conditions is encountered:
  A. The XMIT key is pressed.
  B. A transmit character is received from the computer.

SET UNDERLINE CURSOR–

Control Character sequence:

| | | |
|---|---|---|
| 1E | 14 | (hex) |
| 30 | 20 | (decimal) |
| ^^ | ^T | (keyboard) |

Function: Configure the terminal to display the cursor as an underline character.

SET UPPER CASE ONLY–

Control Character sequence:

| | | |
|---|---|---|
| 1E | 1C | (hex) |
| 30 | 28 | (decimal) |
| ^^ | ^\ | (keyboard) |

Function: Configure the terminal keyboard so that alphabetic characters are forced to upper case.

SPECIAL CONSIDERATIONS:

- The shift key is ignored for alphabetic characters.
- Non-alphabetic characters are unaffected.

### SET VERTICAL CURSOR POSITION–

Control Character sequence:

| | | | |
|---|---|---|---|
| 1C | Ø7 | [Y] | (hex) |
| 28 | 7 | [Y] | (decimal) |
| ^\ | ^G | [Y] | (keyboard) |

Function: Set the vertical cursor position to the specified value.

SPECIAL CONSIDERATIONS:

- Y specifies the vertical position
- Y must be between zero and one less than the maximum number of lines on the screen (inclusive).
- If Y exceeds the allowable range, it is set to the maximum permissible value. (For maximum permissible values, reference the documentation on the CRT Display formats.)
- The CURSOR ON position is cleared.

### SLIDE SCREEN LEFT–

Control Character sequence:

| | | |
|---|---|---|
| 1D | ØC | (hex) |
| 29 | 12 | (decimal) |
| ^] | ^L | (keyboard) |

Function: Shift the entire screen to the left by one character position.

SPECIAL CONSIDERATIONS:

- A blank column is brought in from the right.
- The leftmost column of characters on the screen is lost.
- The cursor position is unchanged.

### SLIDE SCREEN RIGHT–

Control Character sequence:

| | | |
|---|---|---|
| 1D | ØD | (hex) |
| 29 | 13 | (decimal) |
| ^] | ^M | (keyboard) |

Function: Shift the entire screen to the right by one character position.

SPECIAL CONSIDERATIONS:

- A blank column is brought in from the left.
- The rightmost column of characters on the screen is lost.
- The cursor position is unchanged.

### SUPRESS CURSOR DISPLAY–

Control Character sequence:

| | | |
|---|---|---|
| 1E | 15 | (hex) |
| 30 | 21 | (decimal) |
| ^^ | ^U | (keyboard) |

Function: Turn off the displayed cursor.

SPECIAL CONSIDERATION:

- This function is different from the Cursor Off function in that the CURSOR ON position is unchanged.

TRANSMIT (SCREEN READ)–

Control Character Sequence: 17 (hex)
23 (decimal)
^W (keyboard)

Function: Transmit the contents of the screen, from the upper left hand corner to the current cursor position.

SPECIAL CONSIDERATIONS:

- The transmit function will not transmit imbedded control characters, ie., control characters that are displayed on the screen.
- Transmission of data starts at the upper left hand corner of the screen and continues up to, but not including the current position of the cursor.
- Trailing blanks in a line are replaced by a Carriage Return character.
- If the Retain Protected Fields option is set, only unprotected characters are transmitted.
- If the Retain Protected Fields option is not set, protected characters are transmitted with their most significant bit (bit 7) set to a one.

WRITE PROTECTED CHARACTERS–

Control Character sequence: 1E 16 (hex)
30 22 (decimal)
^^ ^V (keyboard)

Function: Configure the terminal to write protected (low intensity) characters to the screen.

SPECIAL CONSIDERATIONS:

- All characters (including blanks) displayed subsequent to the Write Protected characters function are written at low intensity.
- The state of the Retain Protected fields Option is unchanged.

WRITE UNPROTECTED CHARACTERS–

Control Character sequence: 1E Ø6 (hex)
30 6 (decimal)
^^ ^F (keyboard)

Function: Configure the terminal to write characters on the screen in unprotected (high intensity) format.

SPECIAL CONSIDERATION:

- The Retain Protected Fields Option is unchanged.

## YAW LEFT GRAPHICS SCREEN—

Control Character sequence:

| | | |
|---|---|---|
| 1D | 19 | (hex) |
| 29 | 25 | (decimal) |
| ^] | ^Y | (keyboard) |

Function: Shift the pseudo-graphics pixels on the screen left by one pixel position.

SPECIAL CONSIDERATIONS:

- A blank column of pixels is brought in from the right.
- The leftmost column of pixels on the screen is lost.
- The cursor position is unchanged.
- If the Retain Protected Fields option is set, ASCII characters on the screen are treated as protected and not moved.
- If the Retain Protected Fields option is clear, ASCII characters on the screen are replaced with blanks prior to moving the pseudo-graphics pixels.

## YAW RIGHT GRAPHICS SCREEN—

Control Character sequence:

| | | |
|---|---|---|
| 1D | 1A | (hex) |
| 29 | 26 | (decimal) |
| ^] | ^Z | (keyboard) |

Function: Shift the pseudo-graphics pixels on the screen right by one pixel position.

SPECIAL CONSIDERATIONS:

- A blank column of pixels is brought in from the left.
- The rightmost column of pixels on the screen is lost.
- The cursor position is unchanged.
- If the Retain Protected Fields option is set, ASCII characters on the screen are treated as protected and not moved.
- If the Retain Protected Fields option is clear, ASCII characters on the screen are replaced with blanks prior to moving the pseudo-graphics pixels.

## DATA CHARACTER—

Character Sequence: [data character]

Function: Display a data character on the screen.

### SPECIAL CONSIDERATIONS:

- [data character] will be written as a protected character (low intensity), if the Write Protected Characters option is set.
- [data character] will be written as an unprotected character (high intensity), if:
  1. the Write Unprotected Characters option is selected, or
  2. the Graphics Cursor Mode is selected.
- If character protection is to be ignored, [data character] is displayed at the current cursor position and the cursor is bumped to the next cursor position.
- If character protection is to be honored, the character already under the cursor is checked. If it is protected, it is skipped by bumping to the next character. If the next character is protected, it is skipped also. This continues until an unprotected character is encountered. The unprotected character under the cursor is replaced with [data character] and the cursor is bumped to the next cursor position.
- Bumping the cursor to the next cursor position involves the following:
  1. If the cursor is not positioned to the rightmost column, the cursor is bumped to the right by one position.
  2. If the cursor is positioned in the rightmost column, the following actions occur:
     A. If the Automatic New Line on Overflow option is not set, no further action is taken.
     B. If the CT-82 is configured for an Automatic New Line on Overflow, the cursor is positioned to the leftmost column and a Line Feed function is generated.

# APPENDIX C.

## Glossary

**Algorithm—**

A step-by-step sequence of instructions which will accomplish a specific task.

**ASCII Character—**

An abbreviation for American Standard Code for Information Interchange. This is a convention which associates a numerical value with all of the letters, numbers, mathematical symbols, etc. recognized by those I/O devices which make use of this convention. The CT-82 makes use of the ASCII convention. An appendix contains a complete table of the characters and their associated values.

**Baud—**

See Baud Rate.

**Baud Rate—**

The speed at which bits of information are transmitted over a serial data line. The baud rate is defined as the number of bits per second transmitted over the line. An approximate value for the amount of time between characters can be calculated from the baud rate. For example, if for each character transmitted over the line there is 8 bits of data and 2 bits of control information, then for 9600 baud the time between characters is 10/9600 seconds per character or slightly over 1 millisecond per character.

**Binary—**

A number expressed in base 2 notation.

**Bit—**

A single digit of a binary number. A bit may be either Ø (cleared) or 1 (set).

**Blank—**

ASCII character 2Ø (hex). See space.

**Buffer—**

An interface which compensates for timing differences between the two things being interfaced. Information coming into the buffer is held by the buffer until such time as the information can be accepted from the buffer.

**Bump—**

A verb meaning to move something by one. For example: Bump the cursor left means move the cursor position one to the left.

**Character—**

See ASCII character, data character and control character.

**Character Generator—**

A device used internally which converts from the ASCII (or other) character set to the set of dots which make up that character on the screen.

**Character Position—**

The space occupied by one character on the screen.

**Character Protection—**

There is a flag associated with each character displayed on the screen which indicates whether the character is protected or unprotected. When displayed on the screen, unprotected characters will appear to be brighter than protected characters. If the CT-82 is configured to pay attention to character protection, some of the control functions will treat protected characters

differently than unprotected characters. For example, erases will not erase protected characters.

**Clear—**

See the definition of bit.

**Column—**

1.) When the CT-82 is configured for a format other than graphics, a column refers to a line of character positions arranged vertically.
2.) When the CT-82 is configured for graphics, a column refers to a line of pixels arranged vertically.

**Control Character—**

An ASCII character that has a control function associated with it by an I/O device. Unless it is configured otherwise, the CT-82 treats ASCII characters in the range ØØ - 1F (hex) as control characters. Sending a control character or a sequence of control characters causes the CT-82 to execute the associated control function.

**Control Function—**

The data transmitted to and from the CT-82 must be manipulated in the manner desired by the user. Control functions is the method provided to force the CT-82 to manipulate the data as desired. When a control function is executed by the CT-82, data displayed on the screen may be modified, data may be transmitted by the CT-82, or a number of other actions may be taken. A complete description of all the control functions that may be executed by the CT-82 is provided in Appendix A.

**Conversational Mode—**

See the description of the Conversational Mode/Paged Edit Mode option and the Half/Full duplex Mode option.

**CPU—**

An abbreviation for Central Processing Unit. This term is used to refer to the actual computer in a computer system.

**CRT—**

An abbreviation of the words Cathode Ray Tube. A CRT is the display device (picture tube) used in terminals, television sets and so on.

**Current Cursor Position—**

The cursor position prior to executing a function which modifies the cursor position.

**Cursor—**

The cursor is an indicator on the display screen used to call the operator's attention to a single character position on the screen. When the CT-82 is turned on, the cursor may be seen blinking in the upper left corner of the screen. The CT-82 allows the appearance of the cursor to be modified to suit the user's needs. For example, the cursor may look like an underline or it may be a block occupying the entire character position.

**CURSOR ON Position—**

The cursor's horizontal position at the time when the cursor is turned on is remembered as the CURSOR ON position. This is not the cursor position. The Cursor position indicates where the cursor is, while the CURSOR ON position indicates where the cursor was turned on. The CURSOR ON position is provided as a limit for the Backspace and Cancel control functions, it is not possible to backspace to the left of the CURSOR ON position. Most of the control functions that reposition the cursor cause the CURSOR ON position to be reset to the leftmost column.

**Cursor Position—**

The character position on the display screen where the cursor will be displayed if it is turned on. The cursor position is defined even when the cursor is not displayed. The cursor position is defined with two numbers relative to the upper left corner of the screen, the horizontal position and the vertical position. For example, the upper left corner of the screen is at horizontal position Ø and vertical position Ø, the forty-first column on the top line of the screen is position (4Ø, Ø), the first column on the tenth line is position (Ø, 9) and the eighty-second column on the sixteenth line is position (81, 15).

**Data Character—**

Any ASCII character that is treated as nothing but data. A data character does not have a control function associated with it. The CT-82 always treats ASCII characters in the range 2Ø - 7F (hex) as data characters. Unless it is configured otherwise, the CT-82 treats ASCII characters in the range ØØ - 1F (hex) as control characters. Sending a data character to the CT-82 causes it to be displayed on the screen.

**Data Link Escape—**

A leadin character that informs the terminal that the following character is to be treated as a data character no matter what its value.

**Data Path—**

The path by which information gets from one device to another.

**Decimal—**

A number expressed in base 1Ø notation.

**DEL—**

A term sometimes used for the ASCII character associated with the number 7F (hex). See also, Rubout.

**Delete—**

A verb describing the process of removing data from the screen and moving the surrounding data to fill the space vacated by the removed data.

**Device—**

This term is used to describe, in a general way, any piece of hardware which is being discussed. Usually, when the term device is used, the function of the piece of hardware is of more interest than what the hardware looks like. Also, this term is sometimes used to describe a sub-program (piece of a program), where the function of the sub-program is of more interest than the details of the sub-program.

**Display—**

A verb meaning to put a data character on the screen so that it is visible to the operator.

**Echo—**

A verb used to describe an action taken by a computer in response to an input. When a computer receives a character from a terminal and echoes it back to the terminal, it is simply sending that same character back to the terminal to be displayed on the screen.

**Enter—**

A verb used when refering to the characters being typed in from a keyboard.

**Erase—**

The CT-82 erases a character on the screen by replacing the character to be erased with a ASCII blank, 2Ø (hex).

**Escape—**

See the definition of Data Link Escape.

**Execute—**

A verb used to describe a CPU processing a sequence of instructions.

**Field—**

A sequence of characters on a single line treated as a unit by the CT-82. A field usually consists of characters that are all protected or all unprotected. One of the following definitions of a field applies, depending on the control function being executed.

1.) The beginning of the field is the first character of the sequence having the same protection as the rest of the field. The end of the field is the last character of the sequence having the same protection as the rest of the field.

2.) The beginning of the field is the character under the cursor. The end of the field is the last character of the sequence having the same protection as the rest of the field.

3.) The beginning of the field is the character under the cursor. The end of the field depends on the Retain Protected Fields option flag. If the flag is set, the end of the field is the last character of the sequence having the field's protection. If the flag is clear, the end of the field is the end of the line.

**Firmware—**

Programs written for execution by a computer and kept in Read Only Memory. For example, the CT-82 includes as part of its hardware a Motorola 6802 micro-computer. The control functions are implemented in firmware which is executed on the 6802.

**Flag—**

A flag is a value stored in the computer which may be either set or cleared. Typically, a flag is used to decide which of two actions is to occur.

**Frame—**

Another name for the entire screen.

**Full Duplex—**

A mode of operation in a terminal referring to the data paths used by the terminal. Full duplex describes the mode where data typed at the keyboard is sent to the interfaced computer without being displayed on the terminal. This implies that if the data typed at the keyboard is to be displayed on the terminal's screen, the data must be echoed by the interfaced computer.

**Function—**

See Control Function.

**[function byte] —**

A number used to refer to one of the control functions available on the CT-82. For example, the number 53 (hex) refers to the Set Graphics Dot function, the number 28 (hex) refers to the Delete Character, Right control function and the number 18 (hex) refers to the Cancel control function.

**Graphics Control Function—**

A control function which only applies to the CT-82 pseudo-graphics capability.

**Graphics Position—**

The space occupied by one pixel on the screen.

**Group A Control Function—**

The group of 32 control functions that are initially associated with the ASCII characters in the range ØØ - 1F (hex). For a list of the Group A control functions, see Appendix I.

**Group B Control Functions—**

The group of 32 control functions that may be executed by using the GRB Leadin character. For a list of the Group B control functions, see Appendix I.

**Group C Control Functions—**

The group of 32 control functions that may be executed by using the GRC Leadin character. For a list of the Group C control functions, see Appendix I.

**Group D Control Functions—**

The group of 32 control functions used to set or clear the 16 option flags in the CT-82. The GRD Leadin character is followed by a byte specifying which flag is to be modified and whether it is to be set or cleared. For a list of the Group D control functions, see Appendix I. Appendix F contains a description of the option flags.

**Half Duplex—**

A mode of operation in a terminal referring to the data paths used by the terminal. Half duplex describes the mode where data typed at the keyboard is not only sent to the interfaced computer, but is displayed on the terminal's screen with no additional action required by the interfaced computer to echo the data.

**Hardware—**

The actual circuitry that makes up the computer.

**Hex—**

A shortened form of the word hexadecimal, which refers to a number expressed in base 16 notation.

Ø (hex) = Ø (decimal)
1 (hex) = 1 (decimal)
.
.
.
9 (hex) = 9 (decimal)
A (hex) =10 (decimal)
B (hex) =11 (decimal)
.
.
.
F (hex) =15 (decimal)

**Home—**

A verb used to describe positioning the cursor in one of the corners of the screen.

**Horizontal Cursor Position—**

The left to right (horizontal) part of a cursor position. A horizontal cursor position of Ø refers to the leftmost character position in a line.

**Horizontal Position—**

See Horizontal Cursor Position.

**Insert—**

A verb describing the process of moving data already on the screen for the purpose of leaving a blank space on the screen where new data may be displayed.

**Interface—**

When referring to hardware, an interface is a device that allows two other devices to communicate with each other. When referring to software or firmware, an interface is the means of communicating between two processes in the program.

**Inversion, Shift—**

See Shift (Key) Inversion.

**Invert—**

A verb applied to a bit of information. To invert a bit is to change its value to the other value. A Ø is changed to a 1 or vice versa. See the definition of Bit.

**I/O—**

An abbreviation of the words Input/Output. This is a term used when discussing a device which may possibly be an input device, an output device or both an input and an output device.

**Leadin Character—**

A character sent to a terminal that has the single purpose of informing the terminal to treat the following character in a special manner. The leadin character sent to the terminal determines the manner in which the following character is treated. See Data Link Escape.

**Line—**

A line refers to a sequence of character positions arranged horizontally on the screen.

**Memory—**

A device or devices used to save information.

**Mode—**

A mode of operation. If the terminal is operating in one mode, then it will handle the items which are affected by the mode in a manner defined by that mode. Switching to a different mode causes the items to be affected in another manner.

**New Line—**

As used by the CT-82, New Line is defined to mean: insure that the cursor is positioned in the leftmost column of the next line. The CT-82 will execute Carriage Return and Line Feed functions as necessary to get to the next line.

**Pitch—**

A verb used to describe rolling the screen in graphics mode. Pitching the screen up refers to rolling the screen upwards by one pixel position.

**Pixel—**

The smallest unit that can be displayed while using graphics.

**Position—**

See Character Position.

**Protection—**

See Character Protection.

**Pseudo-Graphics—**

Pseudo-graphics refers to a graphics capability in a terminal that is produced using a special character generator and character oriented hardware. A terminal with full graphics capability uses dot oriented hardware instead. The CT-82's graphics capability is provided using pseudo-graphics.

**Quadrant—**

An area on the screen defined by the position of the cursor. Two imaginary lines drawn through the cursor position, one vertical and the other horizontal, divide the screen into four quadrants. Each quadrant includes the line and the column where the cursor is positioned. The upper left quadrant is called the North West Quadrant, the lower left is the South West Quadrant, the lower right is the South East Quadrant and the upper right is the North East Quadrant. This is the same as a map where North is upwards on the screen.

**Retain Protected Fields Option—**

See the "Ignore/Honor Character Protection" option flag.

**Roll—**

A verb meaning to move a group of characters on the screen upwards or downwards by one character position. Rolling the screen upwards forces the top line of the affected area to be lost with a blank line brought up as the bottom line of the affected area. Rolling the screen downwards forces the bottom line of the affected area to be lost with a blank line brought down as the top line of the affected area.

**ROM—**

An abbreviation for Read Only Memory. This term is used to describe a memory that cannot be written into without special hardware. It is used primarily to store information used by a processor where it is not necessary to modify the information saved in memory.

**Row—**

A row refers to a sequence of graphics positions arranged horizontally on the screen.

**Rubout—**

A term sometimes used for the ASCII character associated with the number 7F (hex). See also, DEL.

**Scroll—**

A verb meaning to roll the entire screen. See the definition of Roll.

**Set—**

See the definition of Bit.

**Shift (Key) Inversion—**

Reverse the function of the SHIFT key on the keyboard for alphabetic characters (A - Z). For example, with shift key inversion in effect, a shifted "A" will be transmitted to an "a". For more details, see the "Enable/Disable Shift Inversion" option.

**Shifted Characters—**

A shifted character is the character transmitted by a keyboard when the SHIFT key is depressed while typing another key.

**Slide—**

A verb meaning to move every character horizontally on the screen by one character position. Sliding the screen to the right forces the rightmost column on the screen to be lost with a blank column as the leftmost column of the screen. Sliding the screen to the left forces the leftmost column to be lost with a blank column at the right edge of the screen.

**Software—**

Programs that may be loaded into a computer's memory for execution.

**Space—**

1) ASCII character 2Ø (hex). Displaying this character leaves the character position blank.

2) An area (in memory, on the screen, etc) made available for later use.

**Transmit—**

The process of moving information from one device to another. Typically the CT-82 will transmit information to a CPU with the CPU transmitting information back to the CT-82.

**Under the Cursor—**

"Under the Cursor" is a phrase used to refer to the character that is displayed at the cursor position.

**Vertical Cursor Position—**

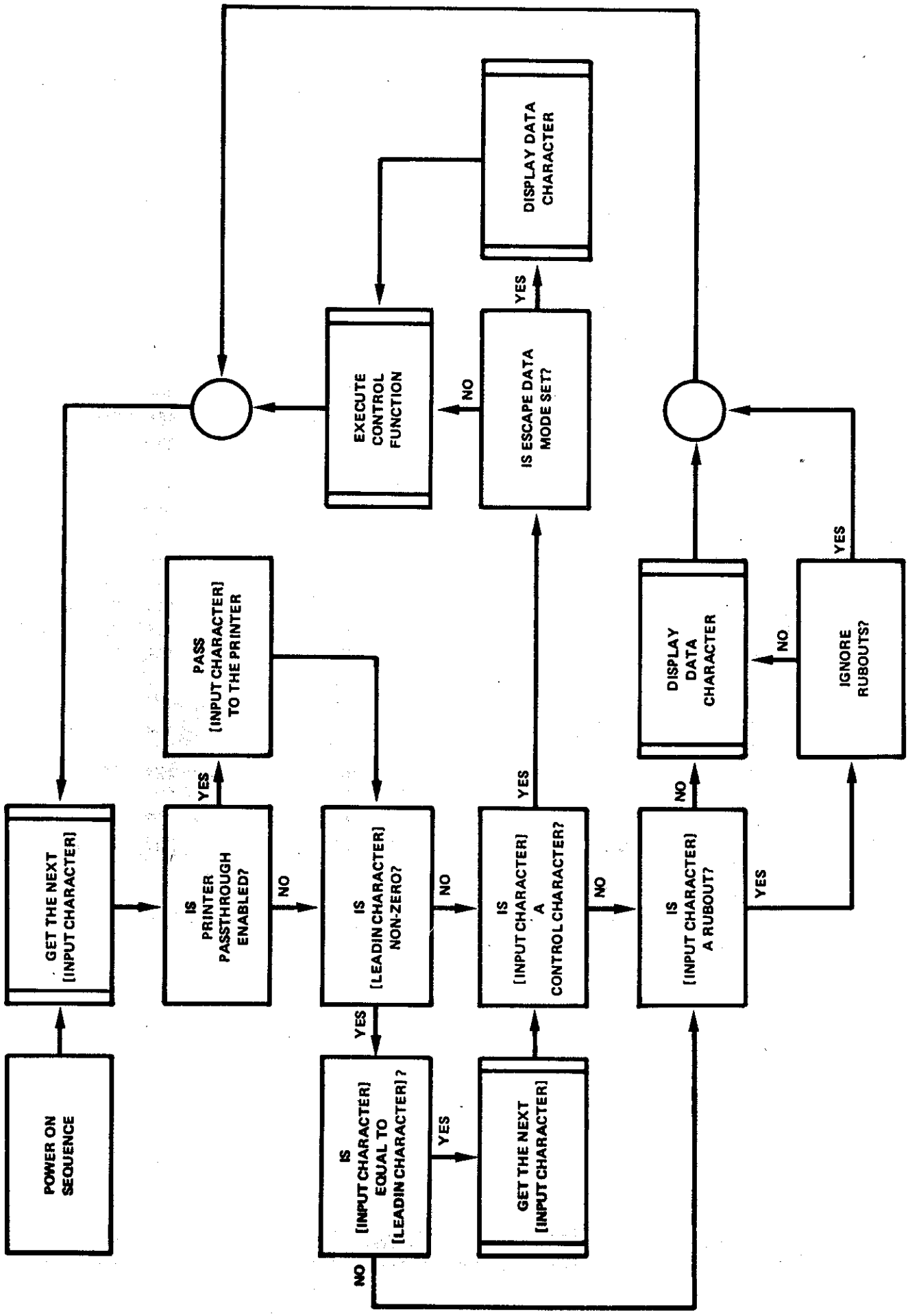
The top to bottom (vertical) part of a cursor position. A vertical cursor position of Ø refers to the top character position in a column.

**Vertical Position—**

See Vertical Cursor Position.

**Yaw—**

A verb used to describe sliding the screen in graphics mode. Yawing the screen to the right refers to sliding the screen to the right by one pixel position.


POWER ON SEQUENCE
GET THE NEXT [INPUT CHARACTER]
IS PRINTER PASSTHROUGH ENABLED?
YES
PASS [INPUT CHARACTER] TO THE PRINTER
NO
IS [LEADIN CHARACTER] NON-ZERO?
NO
YES
IS [INPUT CHARACTER] EQUAL TO [LEADIN CHARACTER]?
YES
GET THE NEXT [INPUT CHARACTER]
NO
IS [INPUT CHARACTER] A CONTROL CHARACTER?
YES
IS ESCAPE DATA MODE SET?
YES
DISPLAY DATA CHARACTER
NO
EXECUTE CONTROL FUNCTION
NO
IS [INPUT CHARACTER] A RUBOUT?
NO
DISPLAY DATA CHARACTER
YES
IGNORE RUBOUTS?
YES

# APPENDIX E

## ASCII Character Set

| DEC | HEX | ASCII | DEC | HEX | ASCII | DEC | HEX | ASCII | DEC | HEX | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 00 | ^@ | 32 | 20 | | 64 | 40 | @ | 96 | 60 | ` |
| 1 | 01 | ^A | 33 | 21 | ! | 65 | 41 | A | 97 | 61 | a |
| 2 | 02 | ^B | 34 | 22 | " | 66 | 42 | B | 98 | 62 | b |
| 3 | 03 | ^C | 35 | 23 | # | 67 | 43 | C | 99 | 63 | c |
| 4 | 04 | ^D | 36 | 24 | $ | 68 | 44 | D | 100 | 64 | d |
| 5 | 05 | ^E | 37 | 25 | % | 69 | 45 | E | 101 | 65 | e |
| 6 | 06 | ^F | 38 | 26 | & | 70 | 46 | F | 102 | 66 | f |
| 7 | 07 | ^G | 39 | 27 | ' | 71 | 47 | G | 103 | 67 | g |
| 8 | 08 | ^H | 40 | 28 | ( | 72 | 48 | H | 104 | 68 | h |
| 9 | 09 | ^I | 41 | 29 | ) | 73 | 49 | I | 105 | 69 | i |
| 10 | 0A | ^J | 42 | 2A | * | 74 | 4A | J | 106 | 6A | j |
| 11 | 0B | ^K | 43 | 2B | + | 75 | 4B | K | 107 | 6B | k |
| 12 | 0C | ^L | 44 | 2C | , | 76 | 4C | L | 108 | 6C | l |
| 13 | 0D | ^M | 45 | 2D | – | 77 | 4D | M | 109 | 6D | m |
| 14 | 0E | ^N | 46 | 2E | . | 78 | 4E | N | 110 | 6E | n |
| 15 | 0F | ^O | 47 | 2F | / | 79 | 4F | O | 111 | 6F | o |
| 16 | 10 | ^P | 48 | 30 | 0 | 80 | 50 | P | 112 | 70 | p |
| 17 | 11 | ^Q | 49 | 31 | 1 | 81 | 51 | Q | 113 | 71 | q |
| 18 | 12 | ^R | 50 | 32 | 2 | 82 | 52 | R | 114 | 72 | r |
| 19 | 13 | ^S | 51 | 33 | 3 | 83 | 53 | S | 115 | 73 | s |
| 20 | 14 | ^T | 52 | 34 | 4 | 84 | 54 | T | 116 | 74 | t |
| 21 | 15 | ^U | 53 | 35 | 5 | 85 | 55 | U | 117 | 75 | u |
| 22 | 16 | ^V | 54 | 36 | 6 | 86 | 56 | V | 118 | 76 | v |
| 23 | 17 | ^W | 55 | 37 | 7 | 87 | 57 | W | 119 | 77 | w |
| 24 | 18 | ^X | 56 | 38 | 8 | 88 | 58 | X | 120 | 78 | x |
| 25 | 19 | ^Y | 57 | 39 | 9 | 89 | 59 | Y | 121 | 79 | y |
| 26 | 1A | ^Z | 58 | 3A | : | 90 | 5A | Z | 122 | 7A | z |
| 27 | 1B | ^[ | 59 | 3B | ; | 91 | 5B | [ | 123 | 7B | { |
| 28 | 1C | ^\ | 60 | 3C | < | 92 | 5C | \ | 124 | 7C | \| |
| 29 | 1D | ^] | 61 | 3D | = | 93 | 5D | ] | 125 | 7D | } |
| 30 | 1E | ^^ | 62 | 3E | > | 94 | 5E | ^ | 126 | 7E | ~ |
| 31 | 1F | ^_ | 63 | 3F | ? | 95 | 5F | _ | 127 | 7F | DEL |

# APPENDIX F

## Configuration Options

### Screen Formats



### Option Flags

# APPENDIX I

## Numeric List of Control Functions with Mnemonics

### GROUP A CONTROL FUNCTIONS

| | | | | Page |
|---|---|---|---|---|
| ^@ | NUL | Ø0 | null | A21 |
| ^A | BCU | Ø1 | bump cursor up | A 2 |
| ^B | BCD | Ø2 | bump cursor down | A 2 |
| ^C | HD | Ø3 | home down | A16 |
| ^D | BCL | Ø4 | bump cursor left | A 2 |
| ^E | COF | Ø5 | cursor off | A 5 |
| ^F | EOL | Ø6 | erase to end of line | A12 |
| ^G | BEL | Ø7 | bell | A 1 |
| ^H | BSP | Ø8 | backspace | A 1 |
| ^I | BCR | Ø9 | bump cursor right | A 2 |
| ^J | LF | ØA | line feed | A19 |
| ^K | SXY | ØB | set cursor position (X,Y) | A30 |
| ^L | FF | ØC | form feed | A15 |
| ^M | CR | ØD | carriage return | A 3 |
| ^N | SU | ØE | scroll up | A27 |
| ^O | SD | ØF | scroll down | A27 |
| ^P | HU | 10 | home up | A16 |
| ^Q | XON | 11 | reader on | A24 |
| ^R | PON | 12 | punch on | A22 |
| ^S | XOF | 13 | reader off | A24 |
| ^T | POF | 14 | punch off | A22 |
| ^U | CON | 15 | cursor on | A15 |
| ^V | EOF | 16 | erase to end of frame | A12 |
| ^W | XMT | 17 | transmit (screen read) | A36 |
| ^X | CAN | 18 | cancel | A 3 |
| ^Y | ILU | 19 | insert line, up | A18 |
| ^Z | DLU | 1A | delete line, up | A 7 |
| ^[ | ESC | 1B | escape | A14 |
| ^\ | GRB | 1C | Group B control sequence | A15 |
| ^] | GRC | 1D | Group C control sequence | A15 |
| ^^ | GRD | 1E | option bit control | A21 |
| ^_ | OPT | 1F | option ROM control sequence | A21 |

### GROUP B CONTROL FUNCTIONS

| | | | | Page |
|---|---|---|---|---|
| ^@ | DCC | Ø0 | display control character | A 8 |
| ^A | MCU | Ø1 | move cursor up | A20 |
| ^B | MCD | Ø2 | move cursor down | A20 |
| ^C | HDR | Ø3 | home down to right | A16 |
| ^D | MCL | Ø4 | move cursor left | A20 |
| ^E | SBG | Ø5 | set background mode | A27 |
| ^F | EBL | Ø6 | erase to beginning of line | A12 |
| ^G | SY | Ø7 | set vertical cursor position | A35 |
| ^H | DCR | Ø8 | delete character, right | A 6 |
| ^I | MCR | Ø9 | move cursor right | A20 |
| ^J | LUF | ØA | line unfeed | A19 |
| ^K | SYX | ØB | set cursor position (Y,X) | A31 |
| ^L | ENW | ØC | erase north west quadrant | A13 |
| ^M | ESW | ØD | erase south west quadrant | A14 |
| ^N | UNW | ØE | roll up north west quadrant | A26 |
| ^O | USW | ØF | roll up south west quadrant | A26 |
| ^P | HUR | 10 | home up to right | A16 |
| ^Q | CF1 | 11 | set CRT format I | A29 |
| ^R | CF2 | 12 | set CRT format II | A29 |
| ^S | CF3 | 13 | set CRT format III | A29 |
| ^T | CF4 | 14 | set CRT format IV | A29 |
| ^U | SFG | 15 | set foreground mode | A31 |
| ^V | EBF | 16 | erase to beginning of frame | A11 |
| ^W | SX | 17 | set horizontal cursor position | A33 |
| ^X | ICR | 18 | insert character, right | A17 |
| ^Y | ILD | 19 | insert line, down | A18 |
| ^Z | DLD | 1A | delete line, down | A 7 |
| ^[ | SLC | 1B | set leadin character | A34 |
| ^\ | ENE | 1C | erase north east quadrant | A13 |
| ^] | ESE | 1D | erase south east quadrant | A13 |
| ^^ | UNE | 1E | roll up north east quadrant | A25 |
| ^_ | USE | 1F | roll up south east quadrant | A26 |

### GROUP C CONTROL FUNCTIONS

| | | | | Page |
|---|---|---|---|---|
| ^A | RCP | Ø1 | read cursor position | A23 |
| ^B | RLP | Ø2 | read light pen position | A23 |
| ^C | SGL | Ø3 | set graphics line | A33 |
| ^D | CGL | Ø4 | clear graphics line | A 5 |
| ^E | IGL | Ø5 | invert graphics line | A19 |
| ^F | EFL | Ø6 | erase field | A12 |
| ^G | PTF | Ø7 | disable terminal printer passthru | A 8 |
| ^H | DCL | Ø8 | delete character, left | A 6 |
| ^I | PDG | Ø9 | pitch down graphics screen | A22 |
| ^J | PUG | ØA | pitch up graphics screen | A22 |
| ^K | PTN | ØB | enable terminal printer passthru | A11 |
| ^L | SSL | ØC | slide screen left | A35 |
| ^M | SSR | ØD | slide screen right | A35 |
| ^N | DNW | ØE | roll down north west quadrant | A24 |
| ^O | DSW | ØF | roll down south west quadrant | A25 |
| ^P | DLE | 10 | data link escape | A 6 |
| ^Q | SPB | 11 | set character protect bit | A28 |
| ^R | CPB | 12 | clear character protect bit | A 4 |
| ^S | SGD | 13 | set graphics dot | A32 |
| ^T | CGD | 14 | clear graphics dot | A 4 |
| ^U | IGD | 15 | invert graphics dot | A18 |
| ^V | CFG | 16 | set graphics CRT display format | A32 |
| ^W | XLT | 17 | change control character | A 3 |
| ^X | ICL | 18 | insert character, left | A17 |
| ^Y | YLG | 19 | yaw left graphics screen | A37 |
| ^Z | YRG | 1A | yaw right graphics screen | A37 |
| ^[ | HEX | 1B | display byte in hex | A 8 |
| ^\ | DEC | 1C | display number in decimal | A 9 |
| ^] | SBR | 1D | set baud rate | A28 |
| ^^ | DNE | 1E | roll down north east quadrant | A24 |
| ^_ | DSE | 1F | roll down south east quadrant | A25 |

### GROUP D CONTROL FUNCTIONS

| | | Page |
|---|---|---|
| ^@ .....@0 | disable escape character | A 7 |
| ^A .....@1 | clear escape data mode | A 4 |
| ^B .....@2 | clear graphics cursor mode | A 4 |
| ^C .....@3 | set blinking cursor | A28 |
| ^D .....@4 | set block cursor | A28 |
| ^E .....@5 | display cursor | A 9 |
| ^F .....@6 | write unprotected characters | A36 |
| ^G .....@7 | ignore character protection | A17 |
| ^H .....@8 | enable scroll on line feed | A10 |
| ^I .....@9 | disable automatic line feed | A 7 |
| ^J .....@A | enable automatic new line | A10 |
| ^K .....@B | enable rubout as data | A10 |
| ^L .....@C | enable upper and lower case | A11 |
| ^M .....@D | enable shift inversion | A10 |
| ^N .....@E | set full duplex | A30 |
| ^O .....@F | set for conversational mode | A30 |
| ^P .....10 | enable escape character | A10 |
| ^Q .....11 | set escape data mode | A31 |
| ^R .....12 | set graphics cursor mode | A32 |
| ^S .....13 | set non-blinking cursor | A34 |
| ^T .....14 | set underline cursor | A34 |
| ^U .....15 | suppress cursor display | A35 |
| ^V .....16 | write protected characters | A36 |
| ^W .....17 | honor character protection | A16 |
| ^X .....18 | disable scroll on line feed | A 8 |
| ^Y .....19 | enable automatic line feed | A10 |
| ^Z .....1A | disable automatic new line | A 7 |
| ^[ .....1B | disable rubout as data | A 8 |
| ^\ .....1C | set upper case only | A34 |
| ^] .....1D | disable shift inversion | A 8 |
| ^^ .....1E | set half duplex | A33 |
| ^_ .....1F | set paged edit mode | A34 |

# APPENDIX H

## Alphabetic List of Control Functions

# APPENDIX J

## Computer System Interconnection

The CT-82 terminal system is designed to operate with computer systems and modems which are designed to communicate via an RS-232-C serial data path. If your equipment is not RS-232-C serial, then the CT-82 will not operate correctly. A brief description of RS-232-C vs other interface methods is given in Appendix O. Appendix S lists each pin of the CT-82's input/output connector and a brief description of each line. Below is a detailed description of each line.

PROTECTIVE GROUND—This line is electrically connected to the CT-82 chassis.

TRANSMITTED DATA—This line carries the data output from the CT-82 to the modem or computer. Output levels on this line are RS-232-C compatible.

RECEIVED DATA—This line carries the data output from the computer or modem to the CT-82. Input signals to this line should be RS-232-C compatible.

RTS—The request to send line is an output from the CT-82 which is raised high (+12 V Nominal) when the CT-82 is ready to transmit a character. The line will remain high until the CTS line is brought high. When the CT-82 has initiated character transmission, the RTS line will be brought low. The RTS line is RS-232-C compatible (active high).

CTS—The clear to send line is an input to the terminal that indicates that the computer or modem is ready to accept data. The CT-82 will not transmit characters when the CTS line is low. If the computer or modem being used does not have provisions for CTS and RTS lines, the CT-82 CTS line should be connected to the CT-82 RTS line. The CTS line is RS-232-C compatible (active high).

DSR—The data set ready line is an RS-232-C compatible input to the terminal. Currently the status of this line has no effect on terminal operation and may or may not be connected to an outside device, as desired.

GROUND—These lines are the signal return lines from the computer or modem.

RLSD—The received line signal detect line is an RS-232-C compatible input to the terminal. Currently the status of this line has no effect on terminal operation and may or may not be connected to an outside device, as desired.

DTR—The data terminal ready line is an RS-232-C compatible output line from the terminal that tells the computer whether or not the CT-82 is ready to accept more characters. Normally the DTR line is held high, but when the CT-82's character buffer is filled (during transmissions at high, >9600, baud rates) the DTR line is brought low until the terminal is ready for more characters.

CLOCK IN—The clock in line is a TTL level compatible input line to the terminal which allows the serial receiver circuitry of the CT-82 to operate at the baud rate specified by the frequency of the input clock. The frequency of the clock should be 16 multiplied by the baud rate. (4800 Hz for 300 baud). This line is normally not used for most applications. Appendix V should be consulted for additional information.

CLOCK OUT—The clock out is a TTL compatible output signal which is identical to the internal baud rate clock used by the serial interface integrated circuit of the CT-82. The frequency of this clock signal is 16 times the baud rate.

READ ON/OFF, PUNCH ON/OFF—These four lines are the decoded lines to control the operation of an optional tape storage device, such as an SWTPC AC-30 cassette interface. These lines are TTL compatible and are low going strobes 20 microseconds wide. A strobe is generated each time the proper control character command is received by the CT-82.

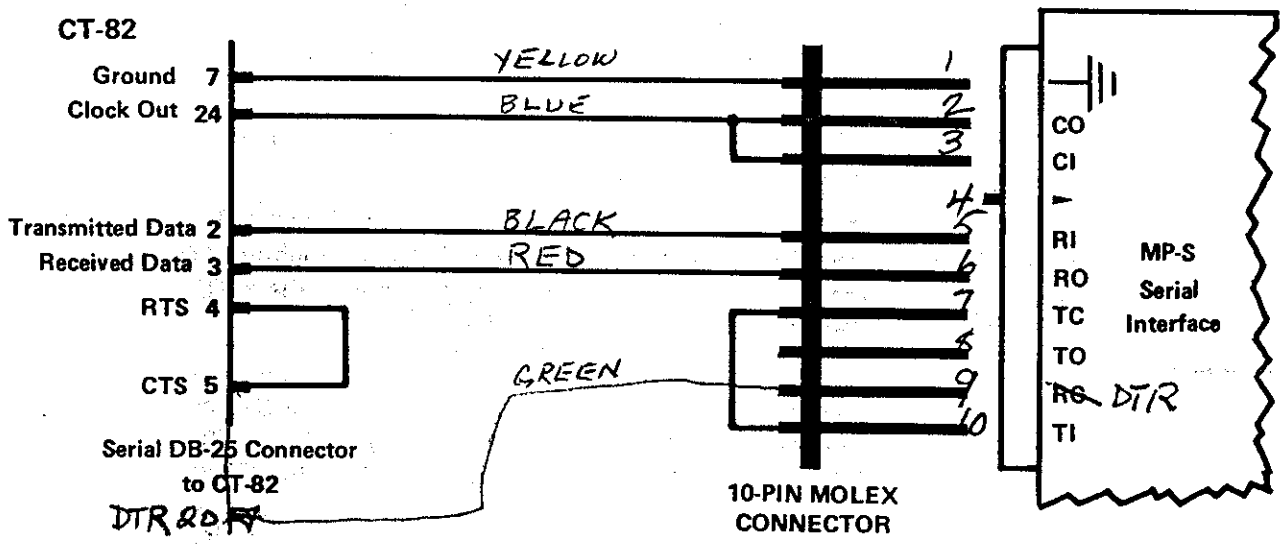
OUT1—This line is a spare output from the CT-82 that any future control ROM's could use for additional functions. Output from this line is RS-232-C compatible.

## Connector Types

All connections to the CT-82 should be done via a 25-pin "D" type connector. The mating connector for the serial RS-232-C cable is designated a DB-25 P. This connector is readily available from most computer and electronic supply stores.

## Connection to a SWTPC 6800 Computer System

To connect a CT-82 terminal to a SWTPC 6800 computer system, the following method is recommended.

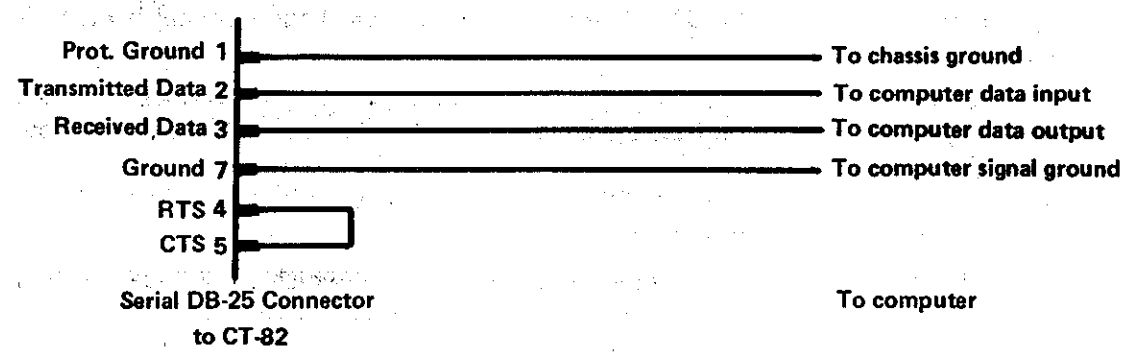



When using the above connection scheme, the MP-S serial interface board of the computer will be driven by the clock of the CT-82. Because of this, be sure to remove all baud rate jumpers on the MP-S board. The above configuration will work to rates of up to 9600 baud. For higher baud rates, consult Appendix L.

## Connections to Other Computers

Because of the large number of computer systems on the market, it is impossible to outline all connection schemes in this manual. Connection to other devices will therefore require knowledge of each CT-82 input/output line and of the equipment to which the CT-82 will be connected.

In most cases the following simple configuration will work:




In the above connection the only lines used are the transmitted data, received data and grounds. This is sufficent for most computers. Notice that pins 4 and 5 (RTS and CTS) on the terminal must be connected together.

## Connections to Most Modems

Most modems are supplied with a 25-pin DB-25 type connector like the one on the CT-82. The following pins should be connected on a one on one basis with the CT-82 serial connector:

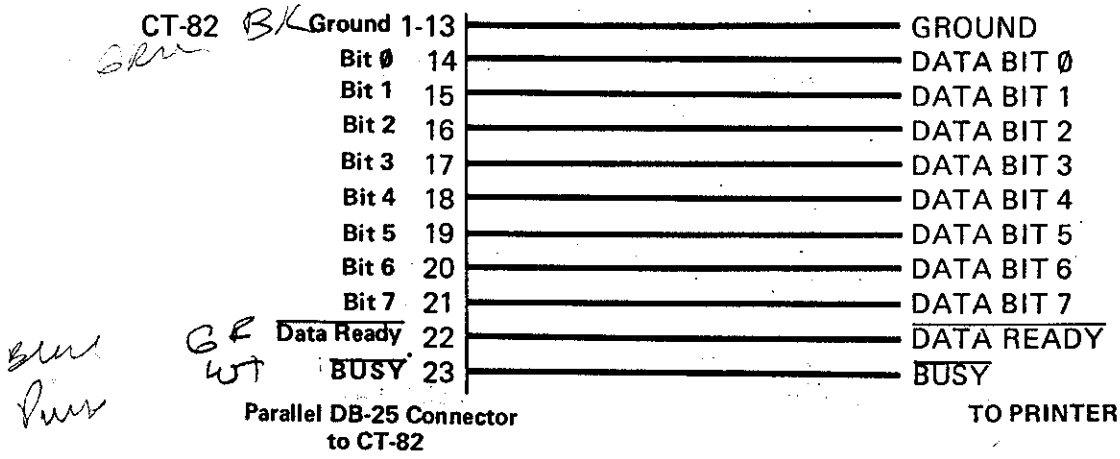
| DB-25 Pin (Modem and CT-82) | Function |
|---|---|
| 1 | Protective ground |
| 2 | Transmitted Data |
| 3 | Received Data |
| 4. | RTS |
| 5 | CTS |
| 6 | DSR |
| 7 | Signal Ground |
| 8 | DCD |
| 17 | *Receiver Clock |
| 20 | DTR |
| 22 | RI |
| 24 | *XMIT Clock |

* Do not connect 17 and 24 if modem uses an internal clock (usually the case).

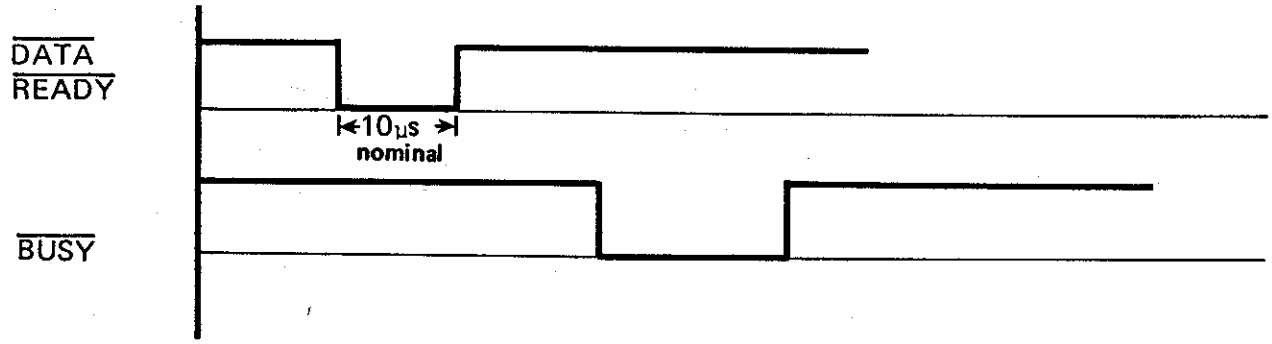
## APPENDIX K

### Parallel Printer Connection

The CT-82 has provisions for attaching a printer via a parallel type interface to allow simultaneous printing on the CRT and the printer. All connections to the printer should be made via the 25-pin PARALLEL connector on the rear of the CT-82. Below is a typical connection drawing.




The DATA READY line is an output from the CT-82 which tells the printer that there is valid data on data lines Ø - 7. This line essentially commands the printer to load the presented data into its buffer. This line is normally high and goes low when the data is valid. The BUSY line is an output from the printer to the terminal which tells the terminal that the printer has accepted the previous character. The CT-82 looks for a high to low transition on this line.




All printer output and input lines are TTL compatible. A DB-25S type connector should be used to connect to the CT-82 parallel port.

# APPENDIX L

## High Baud Rate Operation

The CT-82 terminal system contains enough buffering to allow normal operation up to 9600 baud. If desired, baud rates of up to 38,400 baud may be obtained by using the DATA TERMINAL READY line of the terminal. At high baud rates when the input buffer of the terminal is about to overflow, the CT-82 will automatically toggle the DTR line to its RS-232-C low state. This transition can be used by some computer systems to momentarily halt character transmission to the terminal. When the buffer is again ready to accept characters, the DTR line will be returned to its RS-232-C high state. This type of handshaking is also very useful if a parallel printer is used since most printers are not able to keep up with speeds anywhere near 9600 baud. Documentation on the particular computer the CT-82 is being used with should be consulted to determine if the DTR handshaking scheme can be used. Below are detailed instructions for modifying a SWTPC 6800 Computer System MP-S board to allow DTR operation.

### SWTPC 6800 MP-S Serial Interface Board Modifications to Allow High Baud Rate Operation with a CT-82 Terminal System

In order for the CT-82 Terminal System to operate at baud rates in excess of 9600 baud with the SWTPC 6800 Computer System MP-S Serial Interface board, several simple board changes are necessary on the MP-S board. All changes should be made on the BOTTOM side of the board.

1.) Turn the computer system off and unplug the computer's line cord from the AC wall receptacle.

2.) Remove the MP-S board from the computer mainframe.

3.) Remove resisitor R2 from the circuit board. This may be done by using a solder removal device or by cutting the resistor leads with wire cutters.

4.) Remove resistor R9 from the circuit board in the same manner.

5.) Using a sharp knife or razor blade, carefully cut the trace going to pin 1 of integrated circuit IC3. This cut should be made on the bottom side of the board.

6.) Carefully cut the trace going to pin 24 of integrated circuit IC1. Notice that pins 23 and 24 are connected together and a large ground trace goes to the pin 23-24 connection. Care should be taken to cut only the foil going to pin 24. This should be done on the bottom side of the board.

7.) With a short piece of light gauge insulated wire, connect integrated circuit IC3 pin 1 to the connector pin marked RC on the 10-pin connector along the top edge of the board.

7.5) ADD PULL UP RESISTOR IC3, PIN 1 TO +5 @ PIN 14 (1K - 10K)

8.) Connect a jumper wire from integrated circuit IC3 pin 3 to IC1 pin 24.

9.) Remove all jumpers in the baud rate positions 110, 150, 300, 600 and 1200 on the MP-S board.

10.) Reinstall the board in the correct computer I/O card slot.

This completes the modification of the MP-S board. The data terminal ready line (DTR) from the CT-82 should then be connected to the RC pin position on the connector of the MP-S board, and the CT-82 clock out line should be connected to the MP-S clock in (CI) line on the connector.

NOTE: When making the above changes the use of the reader control line (RC) will be lost. This will usually cause no problems since no SWTPC supplied software supports the reader control line. Since the baud rate clock circuitry of the computer is limited to a maximum of 9600 baud, the clock out line of the CT-82 should be connected to the CLOCK IN (CI) pin of the MP-S board. This will run the MP-S board on the clock derived inside the CT-82 and will allow a data exchange rate at up to 38,400 baud.

## APPENDIX M

### Use of the CONFIGURE SWITCH

### Power-up Default Condition Programming

When the CT-82 is powered up the internal circuitry will go through one of two initialization procedures depending on the position of the CONFIGURE switch on the rear panel. If the switch is set to AUTO, the terminal will initialize to operate at 9600 baud, full duplex, conversational mode with 16 lines per screen. If the switch is set to PROGRAMMABLE the terminal will initialize to whatever conditions are selected by a DIP switch on a small circuit board connected to the microprocessor controller board. Below is a table showing the various possible combinations and information on how to program the switch.

### POWER UP SWITCH PROGRAMMING

| BAUD RATE | SWITCH (bit 43210) Ø = SWITCH ON | HEX | DECIMAL | ASCII |
|---|---|---|---|---|
| 50 | 00000 | 00 | 0 | ^@ |
| 60 | 00001 | 01 | 1 | ^A |
| 75 | 00010 | 02 | 2 | ^B |
| 100 | 00011 | 03 | 3 | ^C |
| 110 | 00100 | 04 | 4 | ^D |
| 120 | 00101 | 05 | 5 | ^E |
| 134.5 | 00110 | 06 | 6 | ^F |
| 150 | 00111 | 07 | 7 | ^G |
| 200 | 01000 | 08 | 8 | ^H |
| 240 | 01001 | 09 | 9 | ^I |
| 300 | 01010 | 0A | 10 | ^J |
| 400 | 01011 | 0B | 11 | ^K |
| 450 | 01100 | 0C | 12 | ^L |
| 600 | 01101 | 0D | 13 | ^M |
| 1000 | 01110 | 0E | 14 | ^N |
| 1200 | 01111 | 0F | 15 | ^O |
| 1800 | 10000 | 10 | 16 | ^P |
| 2000 | 10001 | 11 | 17 | ^Q |
| 2400 | 10010 | 12 | 18 | ^R |
| 3000 | 10011 | 13 | 19 | ^S |
| 3600 | 10100 | 14 | 20 | ^T |
| 4000 | 10101 | 15 | 21 | ^U |
| 4800 | 10110 | 16 | 22 | ^V |
| 6000 | 10111 | 17 | 23 | ^W |
| 7200 | 11000 | 18 | 24 | ^X |
| 9600 | 11001 | 19 | 25 | ^Y |
| 12800 | 11010 | 1A | 26 | ^Z |
| 14400 | 11011 | 1B | 27 | ^[ |
| 19200 | 11100 | 1C | 28 | ^\ |
| 23040 | 11101 | 1D | 29 | ^] |
| 28800 | 11110 | 1E | 30 | ^^ |
| 38400 | 11111 | 1F | 31 | ^_ |

| SWITCH BIT POSITION | SWITCH ON | SWITCH OFF |
|---|---|---|
| 5 | conversational | paged edit |
| 6 | full duplex | half duplex |
| 7 | 16 lines/screen | 20 lines/screen |

# APPENDIX N

## Wiring the CT-82 for Various Power Line Voltages

The CT-82 terminal system has an AC power supply that may be wired for 100, 120, 220 or 240 VAC operation. To reconfigure the power supply the cover and the processor and video circuit boards must be removed to gain access to the power supply. **Be sure** the terminal's AC line cord is not plugged into an AC receptacle.

### 120 VAC

When the terminal is wired for **120 VAC** operation, both the BROWN and BLACK transformer primary wires are connected to fuseholder F1's side lug. The BROWN-YELLOW and BLAC-YELLOW wires both connect to power switch S1's inside lug. Neither the BROWN-RED nor BLACK-RED wires are connected and should be insulated.

### 240 VAC

To wire the terminal for **240 VAC** operation, only the BROWN primary wire should connect to fuseholder F1's side lug. Only the BLACK-YELLOW wire should connect to power switch S1's inside lug. The BROWN-YELLOW and BLACK wires should be connected together and insulated. Neither the BROWN-RED nor BLACK-RED wires are connected and should be insulated.

### 100 VAC

To wire the terminal for **100 VAC** operation, both the BROWN and BLACK transformer wires are connected to fuseholder F1's side lug. The BROWN-RED and BLACK-RED wires both connect to power switch S1's inside lug. Neither the BROWN-YELLOW nor BLACK-YELLOW wires are connected and should be insulated.

### 220 VAC

To wire the terminal for **220 VAC** operation, only the BROWN primary wire should connect to fuseholder F1's side lug. Only the BLACK-YELLOW wire should connect to power switch S1's inside lug. The BROWN-RED and BLACK wires should be connected together and insulated. Neither the BROWN-YELLOW nor BLACK-RED wires are connected and should be insulated.

### IMPORTANT NOTE

Fans, as supplied in the United States, are designed for 100 - 120 VAC 60 Hz use. For other power systems, a substitute fan must be used.

## APPENDIX O

### RS-232-C, Current Loop and TTL Interface Levels

In order to help standardize on the electrical interface levels between computers and computer peripheral equipment, the Electronic Industries Association adopted a standard known as RS-232-C. The standard involves much more than is practical to discuss in this manual, but basically it states that a transmitted binary 1 will be a voltage of a certain level and a binary Ø will be a voltage of another level. Nominal RS-232-C levels in the CT-82 terminal, as measured across one average RS-232-C load, are as follows:

| | | | |
|---|---|---|---|
| Binary 1 | — | −12 VDC | (Referred to a LOW level in this manual) |
| Binary Ø | — | +12 VDC | (Referred to as HIGH level in this manual) |

In the CT-82 the transmitted data, received data, DTR, RTS, CTS, RLSD and DSR lines are all RS-232-C level compatible lines.

Some input/output lines in the CT-82 are what is known as TTL compatible lines. Nominal levels for TTL compatible lines in the CT-82 are as follows:

| | | | |
|---|---|---|---|
| Binary 1 | — | +5 VDC | (Referred to as HIGH level) |
| Binary Ø | — | Ø VDC | (Referred to as LOW level) |

It should be noted that a TTL compatible line should not be directly be connected to a RS-232-C compatible line.

Some computer equipment will interface only to peripherals designed to operate in a current loop mode. As opposed to RS-232-C or TTL, the current loop standard differentiates 1's and Ø's by the amount of current flowing through a loop. The CT-82 is not designed to operate in this manner.

# APPENDIX P

## Optional Character Generator

The CT-82 has the provision for using an optional character generator, like the graphics character generator sold by SWTPC. If desired, a user with the capability of programming 5 volt only, 2716 μv erasable PROMs can make his own custom character generators. This is not recommended for beginners or persons not well versed in this area. Custom EPROMs will require a computer system and PROM programmer along with software and hardware expertise.

To see how the information in the ROM must be formatted, take for example the special characters NU and SH.

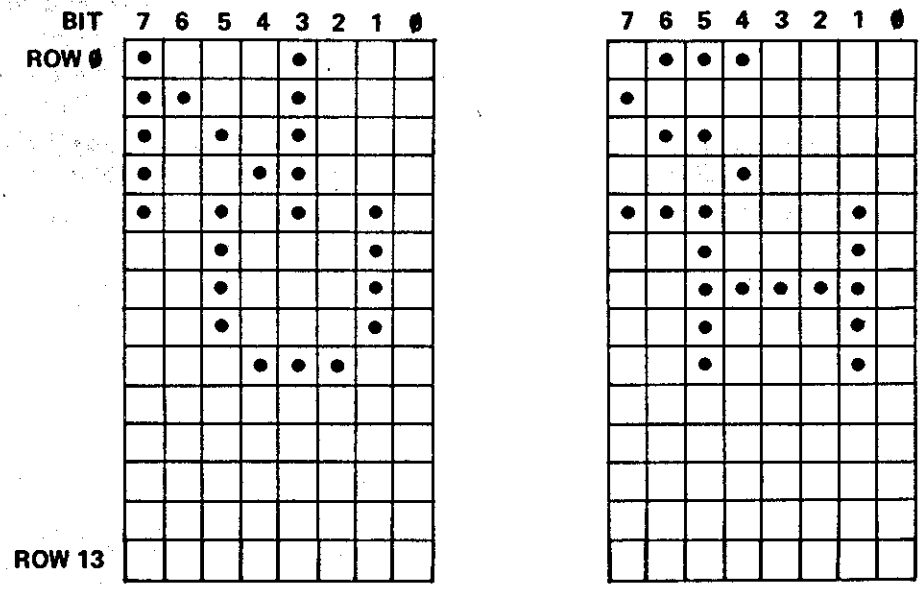



Figure A

Notice that each character occupies a "block" in the ROM which is 8 columns wide and 14 rows deep. When using CRT Format III and CRT Format IV, the software in the CT-82 forces one blank dot position between each character. This effectively generates a character nine columns wide with the ninth column always Ø. For ease of reading, two dot spaces between characters is usually desirable. In the NU and SH examples, the eighth column is set to Ø to give an additional dot spacing. When using the Graphics CRT format the CT-82 will not force any blank dot spaces between characters. Unused rows, in the above case rows 9 - 13 should be left blank (set to Ø).

The information to be stored in the ROM should be stored as follows:

BYTES Ø - 127 (Row Ø character Ø), (Row Ø character 1),.....(Row Ø character 127)
BYTES 128 - 255 (Row 1 character Ø), (Row 1 character 1),.....(Row 1 character 127)
BYTES 256 - 383 (Row 2 character 4), (Row 2 character 1),.....(Row 2 character 127)

≀

BYTES 1664 - 1791 (Row 13, character Ø), (Row 13 character 1),.....(Row 13 character 127)

All remaining BYTES should be made Ø.

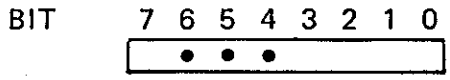
Where the byte number is in decimal, the row number is as shown in Figure A and the character number is as designated by the ASCII code for that particular character. In the above examples an ASCII ØØ is a null (NU ) and an ASCII Ø1 is an SOH control character (designated SH ).

Since byte Ø is defined above to be (row Ø character Ø), the data relating to row Ø of the NU character, row Ø of this character is as follows:

BIT 7 6 5 4 3 2 1 0

therefore the data to be stored in byte Ø will be a binary 1ØØØ1ØØØ. (A 1 designates positions where a dot is to be displayed.)

Byte 1 would then be row Ø of the next character




Ø1110ØØØ.

This process is continued until all rows of all 127 characters are calculated.

The example characters shown above are called 7 x 9 matrix characters since the actual character takes up 7 columns and 9 rows. Bit Ø in the above example was always Ø to allow an additional dot spacing between characters. All selected CRT formats display all 8 columns. The number of rows displayed, however, depends on the CRT format selected. CRT Format III, as described in Appendix A, will select an 82 character by 16 line screen. This means that the total character block displayed is 8 columns by 14 rows with a blank dot spacing forced between each character. For CRT Format IV, the character block displayed is 8 columns by 12 rows with a blank dot spacing forced between each character. For the graphics format, 8 columns by 9 rows are displayed with no additional dot spacing between characters. All blank or unused rows should always be set to Ø.

The optional character generator socket is set up to use 5 v only 2716 type EPROMS or a masked equivalent. After a 2716 EPROM has been programmed, it should be installed in the optional generator socket on the video controller board.

## APPENDIX Q

### Adjusting the Video Monitor

The video monitor has several controls which may need to be adjusted for optimum performance. These controls consist of HORIZONTAL HOLD, HORIZONTAL SIZE, BRIGHTNESS, CONTRAST, FOCUS, VERTICAL SIZE, VERTICAL HOLD and VERTICAL LINEARITY. Since high voltages are present in the monitor care should be exercised when adjusting the monitor. Also be very careful not to short out any parts of the monitor when performing the adjustment—doing so could result in serious damage to the monitor. When making the adjustments you may have to set the cover on top of the monitor to help in alignment. Some adjustments (the most common) may be done by using a long non-metallic screwdriver thru the holes provided in the rear of the chassis. Others require monitor removal. In any case, any adjustments should only be done by competent technical personnel.

1.) Fill the screen with characters. There should be a mixture of both high and low intensity characters.

2.) Adjust the BRIGHTNESS until the characters are on a faint white background, then lower the brightness until the background disappears, or to the desired level.

3.) The video monitor comes pre-adjusted from the factory and adjustment of the HORIZ. HOLD should not be necessary to obtain a stable display. If the control must be adjusted, note the position of the control before turning. The position can be noted by looking straight down from the top of the monitor and observing the small plastic slide that travels along the length of the potentiometer. If a stable display can not be obtained, return the HORIZ. HOLD adjustment to its original setting.

4.) Adjust the CONTRAST for the desired display.

5.) If necessary, the VERTICAL SIZE control may be adjusted to give the most attractive display.

6.) If the VERT. SIZE is adjusted you may have to adjust the vertical linerarity. The VERTICAL LINEARITY control should be adjusted so that characters at the top of the screen are equal in height to those at the bottom of the screen.

7.) If necessary, the HORIZ. SIZE control can be adjusted with a non-metallic alignment tool to give the desired display. The HORIZ. SIZE control is the coil marked as L1 in the monitor.

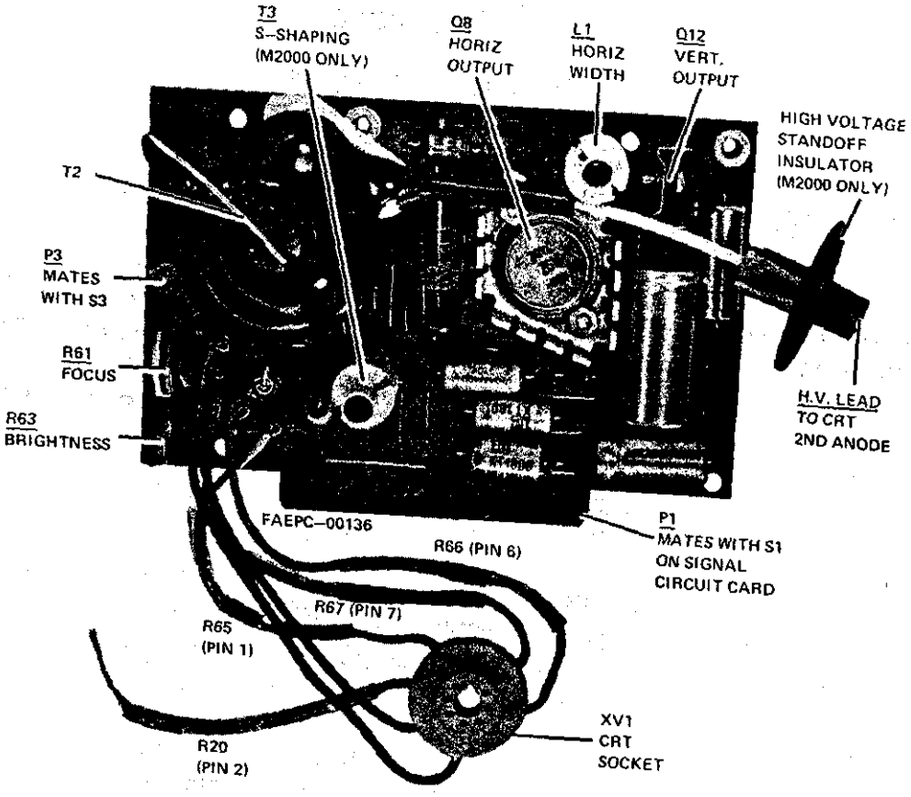
8.) If problems are experienced with vertical sync (the display rolls), the VERTICAL HOLD control can be adjusted for stable display.

9.) Finally, the FOCUS control should be adjusted for optimum focus of characters on all areas of the screen.

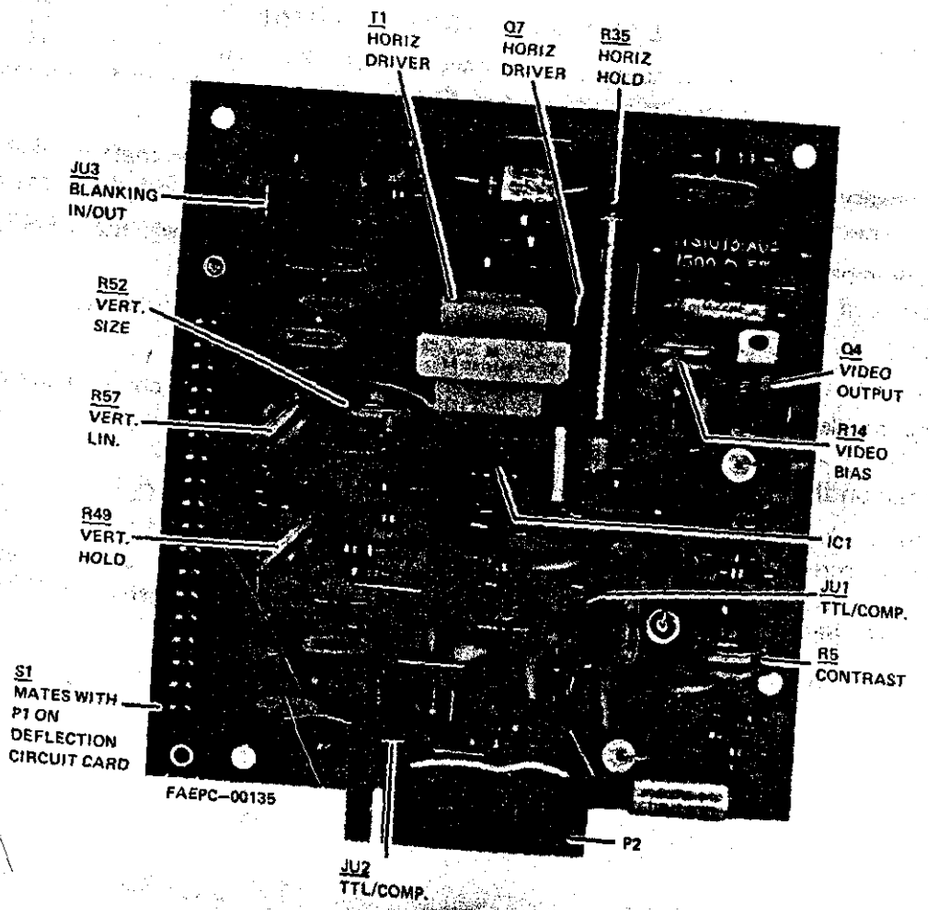
10.) If the display on the screen is positioned incorrectly the two small copper rings on the neck of the CRT may need slight adjustment. To adjust, simply turn the ring on the CRT by grasping the tab on the side of the ring.

11.) If the display is tilted on the screen the YOKE may need adjusting. Carefully loosen the YOKE holding screw and rotate the YOKE as necessary to straighten the display. When finished, CAREFULLY tighten the YOKE screw. Tightening the screw too far could cause the bracket to crack the neck of the CRT, so tighten only enough to securely fasten the YOKE.

Any other monitor adjustments should not be attempted.




**Deflection Circuit Card – Component Side**




**Signal Circuit Card – Component Side**

**Photos courtesy of Motorola Data Products Inc.**

# APPENDIX R

## Basic Theory of Operation

A complete theory of operation on the CT-82 terminal system would be far more lengthy than could be combined with this manual; but, what is included is a basic description of the general aspects of the design of the terminal. A thorough knowledge of micrprocessor and digital circuits is required for competent troubleshooting of the terminal.

The CT-82 consists of five major and separate parts. These parts consist of 1.) the video monitor, 2.) the keyboard, 3.) the power supply, 4.) the video controller board and 5.) the universal microprocessor controller board. Each part will be discussed separately.

### The Video Monitor

The video monitor supplied with the CT-82 is a standard 9 inch video monitor with a 12 Mhz video bandwidth. Signals to the monitor are supplied by the video controller board and consist of video data, horizontal sync and vertical sync. Composite video is not used by this monitor. The video monitor operates at 12 volts DC and requires approximately 1A of current. Monitor adjustments are described in a separate section of this manual.

### The Keyboard

The keyboard is the interface between the user and the electronics of the CT-82. The keyboard consists of 70 keyswitches arranged in a matrix. When a key is depressed the electronics on the small board underneath the main keyboard decodes the key pressed into its ASCII translation and sends the character to the microprocessor controller board for processing.

### The Power Supply

Power for all of the electronics in the CT-82 is supplied by the power supply located underneath the video controller board. This supply delivers the regulated +12 volts required for the video monitor, the regulated ±12 volts required by the keyboard and microprocessor controller board and the +8 volts nominal unregulated necessary for all on-board regulators.

### The Video Controller Board

The video controller board contains the circuitry necessary to display on the video monitor the information stored in the video refresh buffer (memory). Most of the work of the video board is done by a video controller integrated circuit. This IC is responsible for generating the required sync pulses, the video data and the cursor. This board also contains the circuitry necessary for the sharing of memory between the video controller IC and the microprocessor.

### The Microprocessor Controller Board

The microprocessor controller board is the heart of the CT-82 terminal system. This board is actually a small computer system with both read only and random access memory attached. The microprocessor is responsible for the initialization of all other devices in the system and is responsible for accepting and working upon all data input from the keyboard and from the attached computer. The operation of this terminal is different from most others in that all terminal functions are controlled by a program in a ROM rather than by hardwired discrete parts. This makes major changes in the terminal's operation a relatively easy task. The microprocessor controller board also contains the circuitry necessary for interfacing to the keyboard electronics, an optional printer and the computer.

## Appendix S

This appendix contains connector information that should be useful to a competent technician making repairs on a CT-82 terminal system.

### Controller Board Bus Pin Assignments

### Edge Connector

| | |
|---|---|
| pin 1 | NMI (non-maskable interrupt) INPUT |
| pin 2 | +8 VDC unregulated OUTPUT |
| pin 3 | GROUND |
| pin 4 | GROUND |
| pin 5 | +12 VDC regulated @100 ma OUTPUT |
| pin 6 | −12 VDC regulated @100 ma OUTPUT |
| pin 7 | X4 6802 clock INPUT |
| pin 8 | ENABLE OUTPUT |
| pin 9 | $\overline{\text{RESET}}$ OUTPUT |
| pin 10 | $\overline{\text{IRQ}}$ (maskable interrupt) INPUT |
| pin 11 | $\overline{\text{CRT SELECT}}$ OUTPUT |
| pin 12 | $\overline{\text{MEMORY SELECT}}$ OUTPUT |
| pin 13 | R/$\overline{\text{W}}$ OUTPUT |
| pin 14 | COLUMN SELECT OUTPUT |
| pin 15 | ROM SELECT OUTPUT |
| pin 16 | HALF SELECT OUTPUT |
| pin 17 | ADDRESS LINE A11 |
| pin 18 | ADDRESS LINE A10 |
| pin 19 | ADDRESS LINE A9 |
| pin 20 | ADDRESS LINE A8 |
| pin 21 | ADDRESS LINE A7 |
| pin 22 | ADDRESS LINE A6 |
| pin 23 | ADDRESS LINE A5 |
| pin 24 | ADDRESS LINE A4 |
| pin 25 | ADDRESS LINE A3 |
| pin 26 | ADDRESS LINE A2 |
| pin 27 | ADDRESS LINE A1 |
| pin 28 | ADDRESS LINE AØ |
| pin 29 | DATA LINE D7 |
| pin 30 | DATA LINE D6 |
| pin 31 | DATA LINE D5 |
| pin 32 | DATA LINE D4 |
| pin 33 | DATA LINE D3 |
| pin 34 | DATA LINE D2 |
| pin 35 | DATA LINE D1 |
| pin 36 | DATA LINE DØ |

## Connector Legend—6802A Controller Board

### J1 — Serial Interface Connector

| | |
|---|---|
| pin 1 | −12 VDC regulated @60 ma max OUTPUT |
| pin 2 | +12 VDC regulated @60 ma max. OUTPUT |
| pin 3 | +5 VDC regulated @50 ma max. OUTPUT |
| pin 4 | DATA TERMINAL READY OUTPUT |
| pin 5 | DATA SET READY INPUT |
| pin 6 | RS-232 SERIAL INPUT |
| pin 7 | REQUEST TO SEND OUTPUT |
| pin 8 | GROUND |
| pin 9 | RECEIVER LINE SIGNAL DETECT INPUT |
| pin 10 | RS-232 SERIAL OUTPUT |
| pin 11 | GROUND |
| pin 12 | CLEAR TO SEND INPUT |

### J2 — Beeper Configure/Switch Connector on CT-82

| | |
|---|---|
| pin 1 | no connection |
| pin 2 | PAØ of PIA at ØØ8Ø (configure switch for CT-82) |
| pin 3 | GROUND |
| pin 4 | PA1 of PIA at ØØ8Ø |
| pin 5 | +5 VDC regulated OUTPUT |
| pin 6 | OUTPUT FROM 555 TIMER (beeper output on CT-82) |

### J3 — Flat Ribbon Cable Connector (keyboard connector on CT-82)

| | |
|---|---|
| 34, 32, 30, 28, 26 | +5 VDC regulated @100 ma. OUTPUT |
| 24, 22, 20, 18 | +12 VDC regulated @50 ma. OUTPUT |
| 16, 14, 12, 10 | −12 VDC regulated @50 ma. OUTPUT |
| 8, 6, 4, 2 | GROUND |
| 5 | PBØ on PIA at ØØ8Ø (Keyboard bit Ø on CT-82) |
| 7 | PB1 " " " " (Keyboard bit 1 on CT-82) |
| 9 | PB2 " " " " (Keyboard bit 2 on CT-82) |
| 11 | PB3 " " " " (Keyboard bit 3 on CT-82) |
| 13 | PB4 " " " " (Keyboard bit 4 on CT-82) |
| 15 | PB5 " " " " (Keyboard bit 5 on CT-82) |
| 17 | PB6 " " " " (Keyboard bit 6 on CT-82) |
| 19 | PB7 " " " " (Keyboard bit 7 on CT-82) |
| 21 | CB1 " " " " (Keyboard input strobe on CT-82) |
| 23 | CB2 " " " " |
| 25 | PA7 on PIA at ØØ8Ø (cursor bit Ø on CT-82) |
| 27 | PA6 " " " " (cursor bit 1 on CT-82) |
| 29 | PA5 " " " " (cursor bit 2 on CT-82) |
| 21 | PA4 " " " " (cursor bit 3 on CT-82) |
| 33 | CA1 " " " " (cursor input strobe on CT-82) |

### J4 — Power Connector

| | |
|---|---|
| pin 1 | +7 to +9 VDC unregulated @0.75 A INPUT |
| pin 2 | GROUND |
| pin 3 | —12 VDC regulated @100 ma INPUT |
| pin 4 | same as pin 2 |
| pin 5 | same as pin 2 |
| pin 6 | +12 VDC regulated @100 ma INPUT |

### J5 — Control Character Decoder Outputs on CT-82

| | |
|---|---|
| pin 1 | EXT 16X BAUD RATE CLOCK INPUT TO TERMINAL |
| pin 2 | NO CONNECTION |
| pin 3 | +5 VDC REGULATED @ 10mA OUTPUT |
| pin 4 | bit 1 of latch at ØØ8C (READ OFF DECODE ON CT-82) |
| pin 5 | bit 3 of latch at ØØ8C (PUNCH OFF DECODE ON CT-82) |
| pin 6 | GROUND |
| pin 7 | bit Ø of latch at ØØ8C (READ ON DECODE ON CT-82) |
| pin 8 | bit 2 of latch at ØØ8C (PUNCH ON DECODE ON CT-82) |
| pin 9 | buffered 16X baud rate clock output from CT-82) |

### J6—8-Bit Bi-directional Input/Output Connector (printer driver on CT-82)

| | |
|---|---|
| pin 1 | +5VDC regulated @20 ma max. OUTPUT |
| pin 2 | GROUND |
| pin 3 | PA7 on PIA at ØØ98 (OUTPUT bit 7 on CT-82) |
| pin 4 | PAØ " " " " (OUTPUT bit Ø on CT-82) |
| pin 5 | PA6 " " " " (OUTPUT bit 6 on CT-82) |
| pin 6 | PA5 " " " " (OUTPUT bit 5 on CT-82) |
| pin 7 | PA1 " " " " (OUTPUT bit 1 on CT-82) |
| pin 8 | PA4 " " " " (OUTPUT bit 4 on CT-82) |
| pin 9 | PA3 " " " " (OUTPUT bit 3 on CT-82) |
| pin 10 | PA2 " " " " (OUTPUT bit 2 on CT-82) |
| pin 11 | CA1 " " " " (HANDSHAKE INPUT on CT-82) |
| pin 12 | CA2 " " " " (HANDSHAKE OUTPUT on CT-82) |

### J7—8-Bit Bi-directional Input/Output Connector (baud rate select on CT-82)

| | |
|---|---|
| pin 1 | +5 VDC regulated @20 ma max. OUTPUT |
| pin 2 | GROUND |
| pin 3 | PB7 on PIA at ØØ98 (I/O bit 7 on CT-82) |
| pin 4 | PBØ " " " " (I/O bit Ø on CT-82) |
| pin 5 | PB6 " " " " (I/O bit 6 on CT-82) |
| pin 6 | PB5 " " " " (I/O bit 5 on CT-82) |
| pin 7 | PB1 " " " " (I/O bit 1 on CT-82) |
| pin 8 | PB4 " " " " (I/O bit 4 on CT-82) |
| pin 9 | PB3 " " " " (I/O bit 3 on CT-82) |
| pin 10 | PB2 " " " " (I/O bit 2 on CT-82) |
| pin 11 | CB1 " " " " (HANDSHAKE INPUT on CT-82) |
| pin 12 | CB2 " " " " (HANDSHAKE OUTPUT on CT-82) |

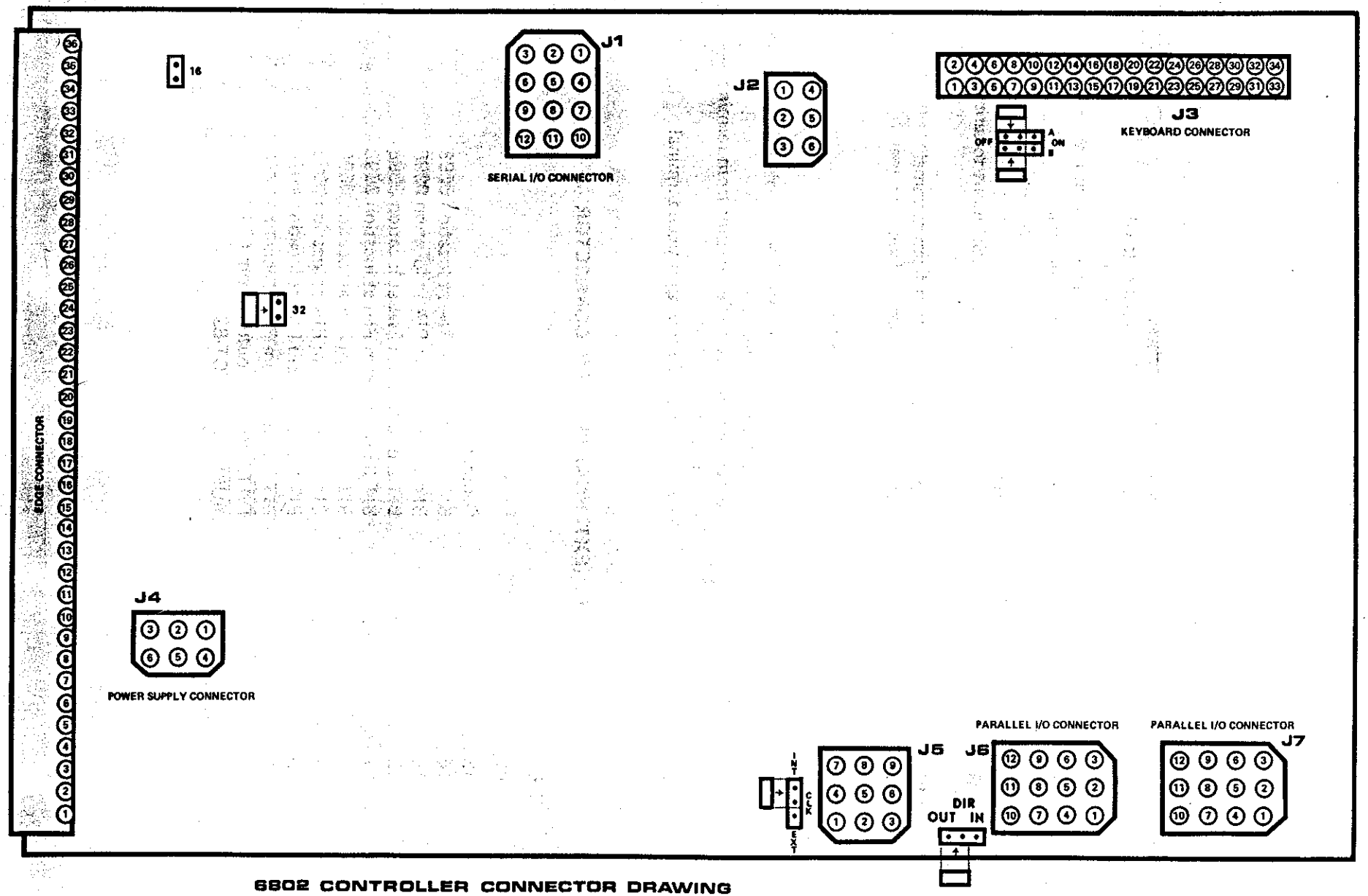
DB-25 Connector Assignments
CT-82 Terminal System

## SERIAL RS-232 CONNECTOR

| DB-25 pin | Internal Connector | Function |
|---|---|---|
| 1 | | Protective ground (chassis) |
| 2 | J1-10 | Transmitted data (Terminal data OUT) |
| 3 | J1-6 | Received data (terminal data IN) |
| 4 | J1-7 | Request to send (RTS output from terminal) |
| 5 | J1-12 | Clear to send (CTS input to terminal) |
| 6 | J1-5 | Data Set Ready (DSR) input to terminal |
| 7 | J1-8, 11 | Ground |
| 8 | J1-9 | Received line signal detect (RLSD input to terminal) |
| 10 | J5-7 | Decoded READ ON output from terminal |
| 11 | J5-8 | Decoded PUNCH ON output from terminal |
| 17 | J5-1 | CLOCK IN input to CT-82 |
| 18 | J5-4 | Decoded READ OFF output from terminal |
| 19 | J5-2 | OUT1 output from terminal |
| 20 | J1-4 | Data Terminal Ready (DTR) output from terminal |
| 24 | J5-9 | Buffered CT-82 CLOCK OUT |
| 25 | J5-5 | Decoded PUNCH OFF output from terminal |

## PARALLEL (EXTERNAL PRINTER) CONNECTOR

| DB-25 pin | Internal Connector | Function |
|---|---|---|
| 1-13 | J6-2 | Ground |
| 14 | J6-4 | BIT Ø output from PIA at location ØØ98 |
| 15 | J6-7 | BIT 1 output from PIA at location ØØ98 |
| 16 | J6-10 | BIT 2 output from PIA at location ØØ98 |
| 17 | J6-9 | BIT 3 output from PIA at location ØØ98 |
| 18 | J6-8 | BIT 4 output from PIA at location ØØ98 |
| 19 | J6-6 | BIT 5 output from PIA at location ØØ98 |
| 20 | J6-5 | BIT 6 output from PIA at location ØØ98 |
| 21 | J6-3 | BIT 7 output from PIA at location ØØ98 |
| 22 | J6-12 | $\overline{\text{DATA READY}}$ output from CT-82 |
| 23 | J6-11 | $\overline{\text{BUSY}}$ input to CT-82 |




6802 CONTROLLER CONNECTOR DRAWING

# Appendix T

## Circuit Board Parts Lists and Component Layout Drawings

### Parts List—6802A Controller Board

#### Integrated Circuits

| | | |
|---|---|---|
| —— | *IC1 | 5 volt only 2716 pinout control ROM or mask equivalent |
| —— | *IC2 | 5 volt only 2716 pinout optional ROM or mask equivalent |
| —— | *IC3 | 1489 quad RS-232 receiver |
| —— | *IC4 | 1488 quad RS-232 transmitter |
| —— | *IC5 | INS8250 Asynchronous Communications element |
| —— | *IC6 | 555 timer |
| —— | *IC7 | 555 timer |
| —— | *IC8 | 6820 or 6821 PIA |
| —— | *IC9 | 6802 controller |
| —— | *IC10 | 74LS139 dual 1 of 4 decoder |
| —— | *IC11 | 74LS273 octal D flip-flop |
| —— | *IC12 | 6820 or 6821 PIA |
| —— | *IC13 | 74LS00 quad NAND gate |
| —— | *IC14 | 74LS20 dual NAND gate |
| —— | *IC15 | 74LS32 quad OR gate |
| —— | *IC16 | 74LS241 octal tri-state buffer |
| —— | *IC17 | 74LS245 octal tri-state buffer |
| —— | *IC18 | 74LS245 octal transceiver |
| —— | *IC19 | 7805 5 volt regulator |

#### Resistors

| | | |
|---|---|---|
| —— | R1 | 1 M ohm ¼ watt resistor |
| —— | R2 | 1 M ohm ¼ watt resistor |
| —— | R3 | 1 M ohm ¼ watt resistor |
| —— | R4 | 10K ohm ¼ watt resistor |
| —— | R5 | 2.2 M ohm ¼ watt resistor |
| —— | R6 | 10K ohm ¼ watt resistor |
| —— | R7 | 3.3K ohm ¼ watt resistor |
| —— | R8 | 3.3K ohm ¼ watt resistor |
| —— | R9 | 10K ohm ¼ watt resistor |

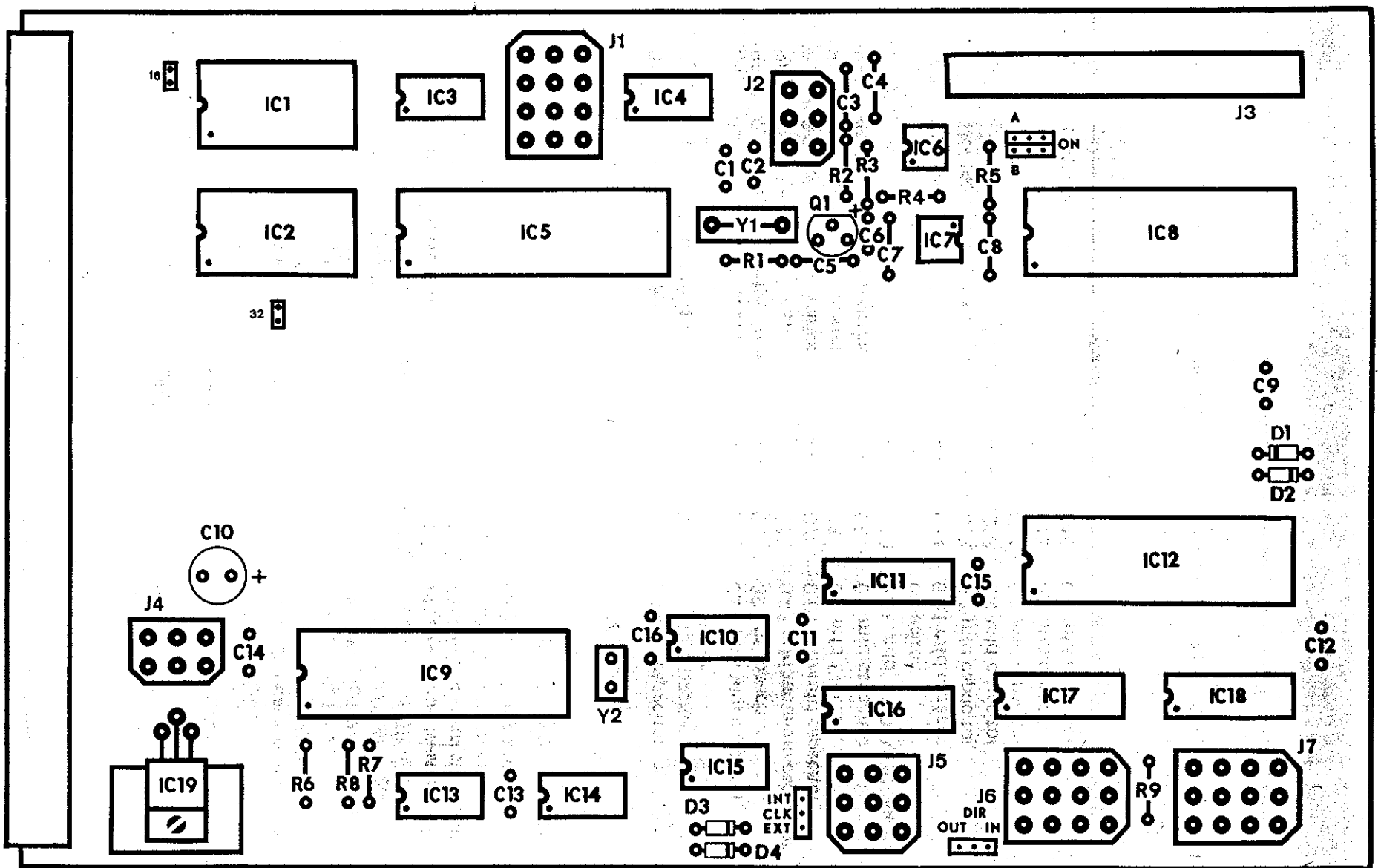
#### Miscellaneous

| | | |
|---|---|---|
| —— | *Q1 | 2N5129 transistor |
| —— | *D1 | 1N4148 silicon diode |
| —— | *D2 | " " " |
| —— | *D3 | " " " |
| —— | *D4 | " " " |
| —— | Y1 | 1.8432 mHz crystal |
| —— | Y2 | 4.0 mHz crystal (not used on CT-82 system) |

#### Capacitors

| | | |
|---|---|---|
| —— | C1 | 0.1 mfd capacitor |
| —— | C2 | 47 pfd capacitor |
| —— | C3 | 0.1 mfd film capacitor |
| —— | C4 | 0.01 mfd film capacitor |
| —— | C5 | 39 pfd capacitor |
| —— | C6 | 0.47 mfd tantalum capacitor |
| —— | C7 | 0.01 film mfd capacitor |
| —— | C8 | 0.1 mfd film capacitor |
| —— | C9 | 0.1 mfd capacitor |
| —— | C10 | 220 mfd @10 VDC electrolytic capacitor |
| —— | C11 | 0.01 mfd capacitor |
| —— | C12 | 0.1 mfd capacitor |
| —— | C13 | 0.1 mfd capacitor |
| —— | C14 | 0.1 mfd capacitor |
| —— | C15 | 0.1 mfd capacitor |
| —— | C16 | 0.1 mfd capacitor |

*Note: All components flagged with a *must be oriented as shown in the component layout drawing.


16
IC1
IC3
J1
IC4
J2
C3
C4
J3
A
ON
B
IC6
C1
C2
R2
R3
R5
R4
Q1
Y1
C6
C7
IC7
C8
IC8
R1
C5
IC2
IC5
32
C9
D1
D2
C10
+
J4
C14
IC9
Y2
C16
IC10
C11
IC11
C15
IC12
C12
IC16
IC17
IC18
IC19
R6
R8
R7
IC13
C13
IC14
IC15
J5
J7
J6
R9
D3
INT
CLK
EXT
DIR
OUT
IN
D4

## Parts List— CT-82 Video Board

### Integrated Circuits

| | | |
|---|---|---|
| —— | *IC1 | 6845 CRT Controller |
| —— | *IC2 | 74LS157 data selector |
| —— | *IC3 | " " " |
| —— | *IC4 | " " " |
| —— | *IC5 | 2114L30 or 4045-30 1K x 4 memory |
| —— | *IC6 | " " " " " |
| —— | *IC7 | " " " " " |
| —— | *IC8 | " " " " " |
| —— | *IC9 | 74LS245 bidirectional transceiver |
| —— | *IC10 | " " " |
| —— | *IC11 | 74LS374 three state latch |
| —— | *IC12 | " " " " |
| —— | *IC13 | 66750 character generator |
| —— | *IC14 | 5 volt only 2716 (optional) character generator |
| —— | *IC15 | 74LS32 quad OR gate |
| —— | *IC16 | 74LS241 three state octal buffer |
| —— | *IC17 | 74LS166 shift register |
| —— | *IC18 | 74LS86 quad exclusive OR gate |
| —— | *IC19 | 74LS04 hex inverter |
| —— | *IC20 | 74S02 quad NOR gate |
| —— | *IC21 | 74LS42 1 of 10 decode |
| —— | *IC22 | 74LS163 Binary Counter |
| —— | *IC23 | 74S74 dual D flip flop |
| —— | *IC24 | 74LS10 triple NAND gate |
| —— | *IC25 | 74LS00 quad NAND gate |
| —— | *IC26 | 74LS241 the state octal buffer |
| —— | *IC27 | 74LS273 octal D flip-flop |
| —— | *IC28 | 74LS08 quad AND gate |
| —— | *IC29 | 74S74 dual D flip-flop |
| —— | *IC30 | 7805 5 volt regulator |

### Resistors

| | | |
|---|---|---|
| —— | R1 | 220 ohm ¼ watt resistor |
| —— | R2 | 100 ohm ¼ watt resistor |
| —— | R3 | 220 ohm ¼ watt resistor |
| —— | R4 | 10K ohm ¼ watt resistor |
| —— | R5 | 10K ohm ¼ watt resistor |
| —— | R6 | 330 ohm ¼ watt resistor |
| —— | R7 | 680 ohm ¼ watt resistor |
| —— | R8 | 1K ohm ¼ watt resistor |

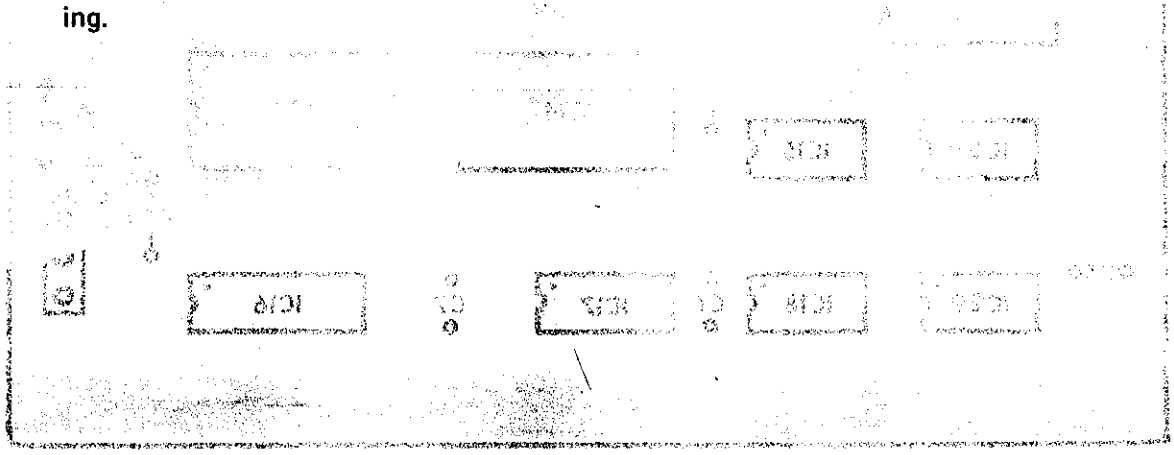
### Capacitors

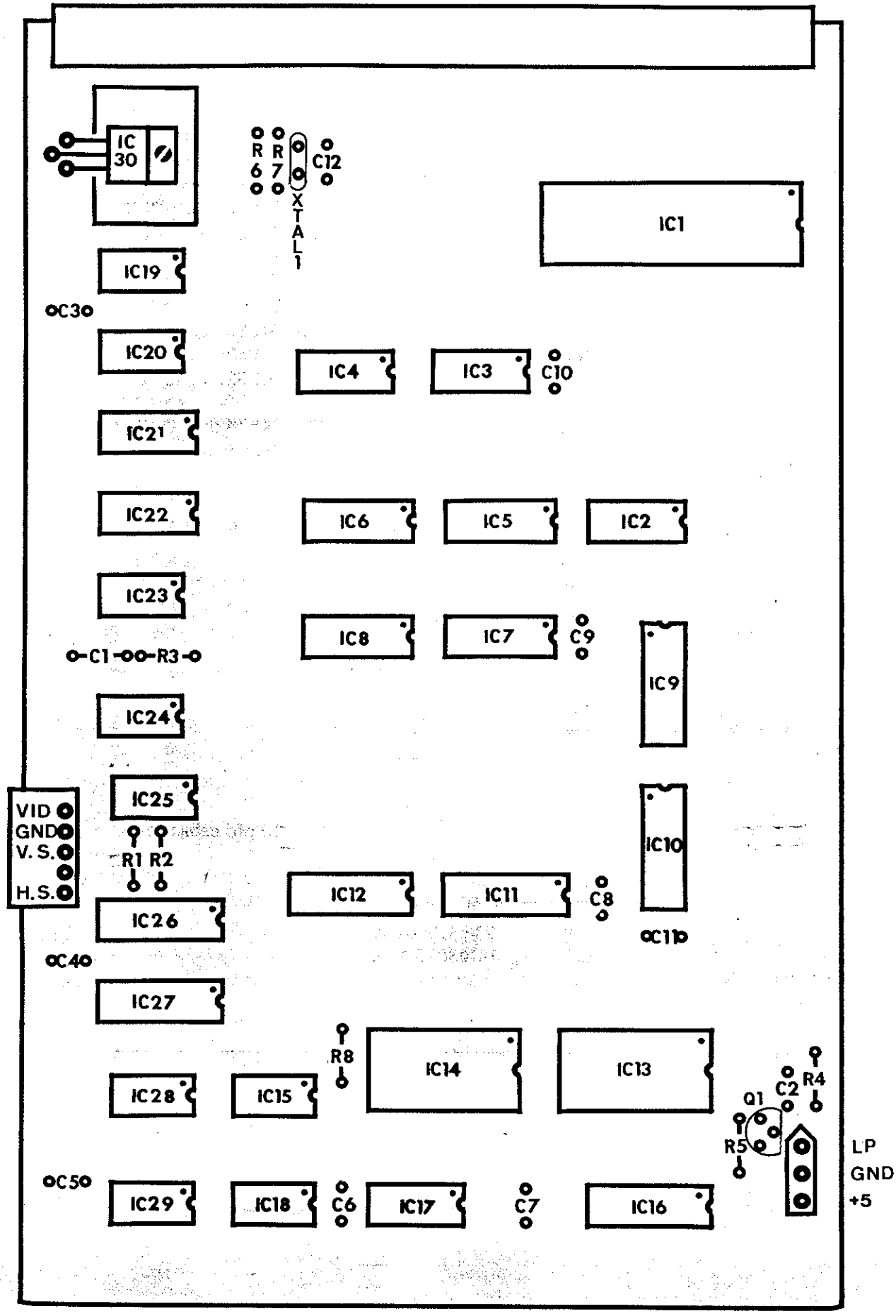
| | | |
|---|---|---|
| —— | C1 | 470 pfd capacitor |
| —— | C2 | 4.7 pfd capacitor |
| —— | C3 | 0.1 mfd capacitor |
| —— | C4 | " " |
| —— | C5 | " " |
| —— | C6 | " " |
| —— | C7 | " " |
| —— | C8 | " " |
| —— | C9 | " " |
| —— | C10 | " " |
| —— | C11 | " " |
| —— | C12 | 10 pfd capacitor |

### Miscellaneous

| | | |
|---|---|---|
| —— | *Q1 | 2N5129 transistor |
| —— | XTAL1 | 14.43096 MHz crystal |

*Note: All components flagged with a * must be oriented as shown in the component layout drawing.




IC 30
R 6
R 7
C12
XTAL 1
IC1
IC19
C30
IC20
IC4
IC3
C10
IC21
IC22
IC6
IC5
IC2
IC23
IC8
IC7
C9
C1
R3
IC9
IC24
IC25
VID
GND
V. S.
H. S.
R1
R2
IC10
IC12
IC11
C8
IC26
C40
C10
IC27
R8
IC14
IC13
Q1
C2
R4
R5
IC28
IC15
LP
GND
+5
C50
IC29
IC18
C6
IC17
C7
IC16

## Parts List P-300 CT Power Supply

### Capacitors

| | | |
|---|---|---|
| —— | *C1 | 5000 mfd @10 VDC electrolytic capacitor |
| —— | *C2 | 3900 mfd @20VDC electrolytic capacitor |
| —— | *C3 | 1000 mfd @25VDC " " |
| —— | C4 | 0.1 mfd film capacitor |
| —— | *C5 | 5000 mfd @ 10 VDC electrolytic capacitor |
| —— | C6 | 0.1 mfd film capacitor |
| —— | C7 | 0.1 mfd film capacitor |
| —— | *C8 | 500 mfd @ 15 VDC electrolytic capacitor |

### Diodes

| | | |
|---|---|---|
| —— | *D1 | 1N5402 silicon diode |
| —— | *D2 | " " " |
| —— | *D3 | " " " |
| —— | *D4 | " " " |
| —— | *D5 | " " " |
| —— | *D6 | 1N4002 " " |
| —— | *D7 | 1N5402 " " |
| —— | *D8 | 1N4002 " " |
| —— | *D9 | 1N4742 12 V zener diode |
| —— | *D10 | 1N4148 silicon diode |

### Regulators

| | | |
|---|---|---|
| —— | *Q1 | MJE3055 transistor |
| —— | *Q2 | 79L12 –12.0VDC @0.1A regulator |

### Miscellaneous

| | | |
|---|---|---|
| —— | *T1 | 120/240 VAC 50 to 60 Hz primary, 24.0 VAC @ 1.25A and 7.25 VAC @ 1.5A secondaries |

### Resistors

| | | |
|---|---|---|
| —— | R1 | 100 ohm ¼ watt resistor |

* All components flagged with a * must be oriented as shown in the component layout drawings.

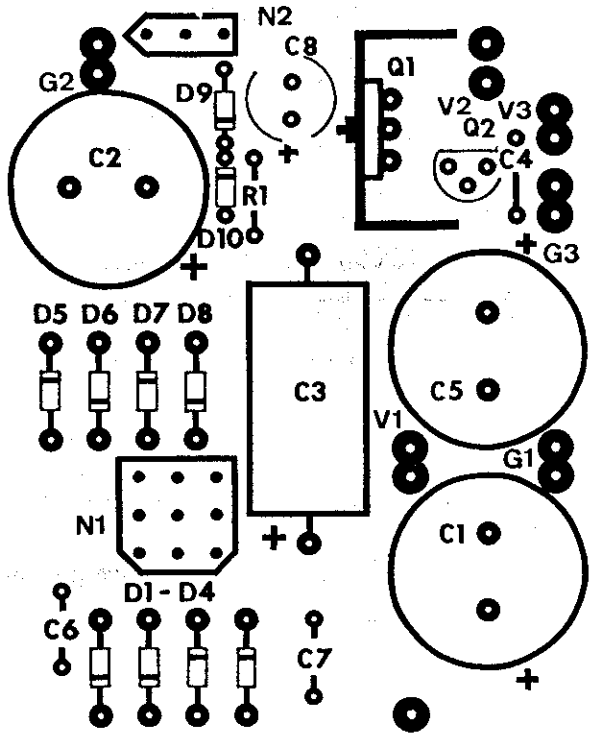
## Parts List — CNTL1 Programming Board

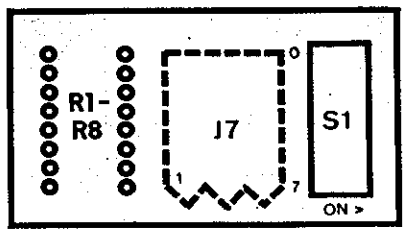
### Resistors

| | | |
|---|---|---|
| —— | R1 | 10K ohm ¼ watt resistor |
| —— | R2 | " " " " " |
| —— | R3 | " " " " " |
| —— | R4 | " " " " " |
| —— | R5 | " " " " " |
| —— | R6 | " " " " " |
| —— | R7 | " " " " " |
| —— | R8 | " " " " " |

### Switches

| | | |
|---|---|---|
| —— | S1 | 8-position SPST DIP switch |


N2
C8
G2
D9
Q1
V2
V3
Q2
C4
C2
R1
D10
G3
D5 D6 D7 D8
C3
C5
V1
G1
N1
C1
D1 - D4
C6
C7


R1-
R8
J7
S1
ON >

## Parts List - KB-6E

### Resistors

| | | | | | |
|---|---|---|---|---|---|
| —— | R1 | 150 ohm ¼ watt resistor (optional) | —— | R5 | 10K ohm ¼ watt resistor |
| —— | R2 | 20K ohm ¼ watt resistor | —— | R6 | 150K ohm ¼ watt resistor |
| —— | R3 | 100K " " " " | —— | R7 | 47K " " " " |
| —— | R4 | 680K " " " " | —— | R8 | 10K " " " " |
| | | | —— | R9 | 20K " " " " |

### Capacitors

| | | | | | |
|---|---|---|---|---|---|
| —— | C1 | 0.01 mfd capacitor | —— | C7 | 1000 pfd capacitor |
| —— | *C2 | 2.2 mfd @ 63 volt electrolytic | —— | C8 | 0.001 mfd capacitor |
| —— | C3 | 0.1 mfd capacitor | —— | C9 | 47 pfd capacitor |
| —— | *C4 | 0.47 mfd tantalum capacitor | —— | C10 | 0.1 mfd capacitor |
| —— | C5 | 0.05 mfd capacitor | —— | C11 | 0.01 mfd capacitor |
| —— | C6 | 0.1 mfd capacitor | —— | *C12 | 0.47 mfd tantalum capacitor |
| | | | —— | C13 | 1000 pfd capacitor |

### Integrated Circuits

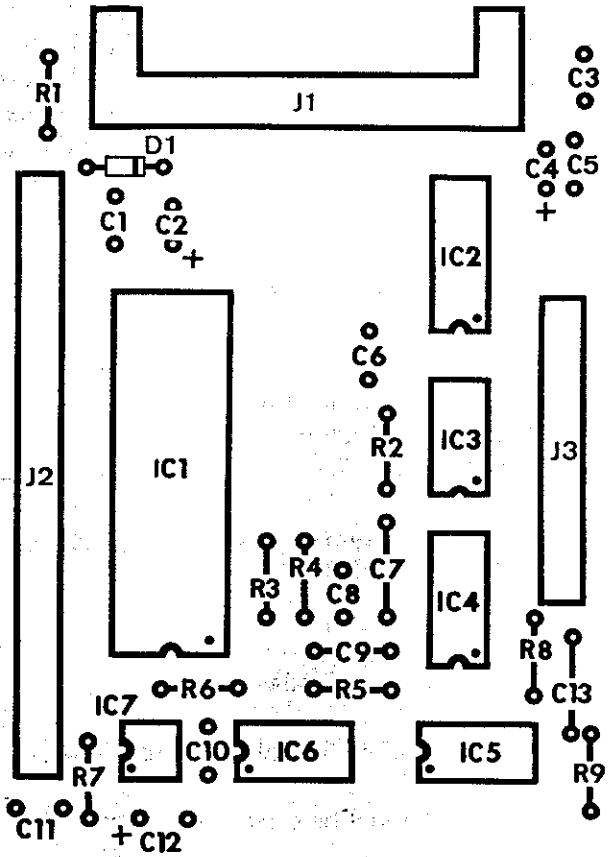
| | | |
|---|---|---|
| —— | *IC1 | AY-5-2376 keyboard encoder |
| —— | *IC2 | 74C922 keyboard encoder |
| —— | *IC3 | 14541 timer |
| —— | *IC4 | 14541 timer |
| —— | *IC5 | 4011 or 14011 CMOS quad NAND gate |
| —— | *IC6 | 4011 or 14011 CMOS quad NAND gate |
| —— | *IC7 | 555 timer |

### Diodes

| | | |
|---|---|---|
| —— | *D1 | 1N4742 12 volt zener (optional) |

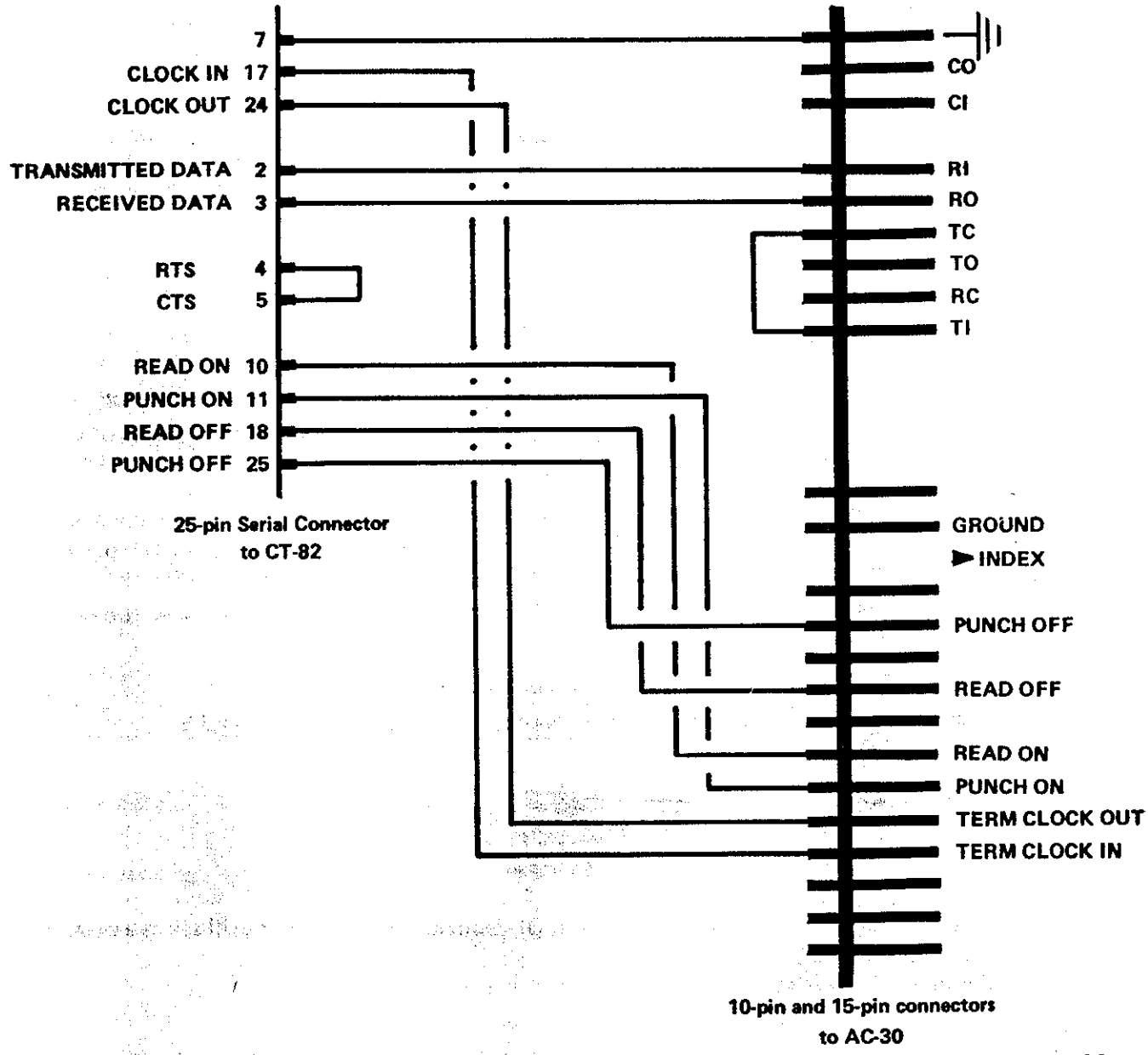
*Note: All components flagged with a * must be oriented as shown on the component layout drawing.

NOTE: Those parts designated as "OPTIONAL" should only be installed when using the board with a non-regulated 14 volt power supply. When using the KB-6E with a CT-82 terminal system, the optional parts should not be installed.


R1
J1
C3
D1
C4
C5
C1
C2
IC2
C6
R2
IC3
J3
J2
IC1
R3
R4
C8
C7
IC4
C9
R8
C13
R6
R5
IC7
C10
IC6
IC5
R7
R9
C11
C12

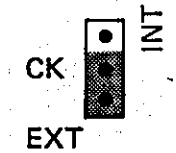
# APPENDIX U

## Using the CT-82 with a SWTPC AC-30 Cassette Tape Interface and 6800 Computer System




Shown above is the wiring diagram to connect a CT-82 terminal system to an SWTPC AC-30 cassette tape interface. When wiring, the AC-30 instructions should be consulted for additional information.

Since the CT-82 will use the clock of the AC-30 during some operations, the CT-82 should be configured to operate with an external clock. This is done by moving the CK jumper on the microprocessor controller board to the EXT position.




Please note that after you move this jumper the CT-82 will not function unless an external clock signal is provided at the I/O connector. If you ever attach the terminal to a device not providing this clock (and most don't) you will have to move the CK jumper to the INT (internal) position for proper operation.

When using the CT-82 with an AC-30 the DTR line is not necessary since the unit will be running at 300 baud.

## APPENDIX V

### 6800 Universal Microprocessor Controller Board Jumpers

The universal controller board contains six sets of jumpers which may be programmed for different hardware configurations. Jumper changes are made by unplugging and re-plugging a small shorting block onto pins on the board. The various jumpers and their use are as follows:

#### 16 Jumper

The two pins marked **16** should have a shorting block installed if 2K x 8 (2716 type) control ROMs are being used.

#### 32 Jumper

The two pins marked **32** should have a shorting block installed if a 4K x 8 control ROM is being used. The CT-82 as supplied should have this block installed.

#### CK Jumper

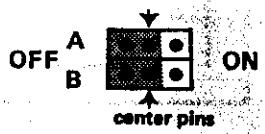
The CK jumper controls which clock the serial interface receiver circuitry of the CT-82 uses. With the CK umper in the **INT** position the serial interface receiver will use an internally generated baud rate clock and this clock signal will be presented on the CLOCK OUT pin on the 25-pin I/O connector of user use.

With the CK jumper in the EXT position, the serial interface receiver will use the clock supplied by the user on the CLOCK IN pin on the 25-pin I/O connector. The CLOCK OUT pin will still contain the internally generated clock and the CT-82 will transmit characters at this rate.

Unless the CT-82 is being used with the AC-30 Cassette Interface, the CK jumper should be installed in the INT position.

#### A and B Jumpers

Near the 34-pin ribbon cable connector is a six-pin jumper block marked A and B.

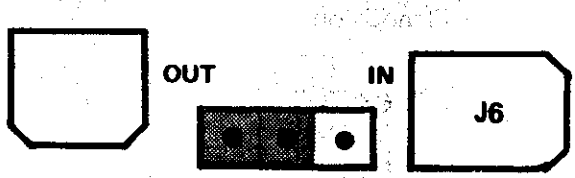



The A jumper should normally be left in the OFF mode. The ON mode will select a possible future optional control ROM.

The B jumper should also be left in the OFF mode at this time.

#### Direction Select Jumper

Next to J6 on the controller board is a set of three pins which selects the direction of the buffer on PIA IC17. The jumper should be programmed as shown below.



## APPENDIX W

### Operation on 50 Hz Power Systems

Proper operation of the CT-82 on 50 Hz power line systems requires a different crystal and control ROM from the ones supplied in the Continental United States. Although the CT-82 will work at 50 Hz as supplied, SWTPC will not guarantee proper performance.

## APPENDIX X

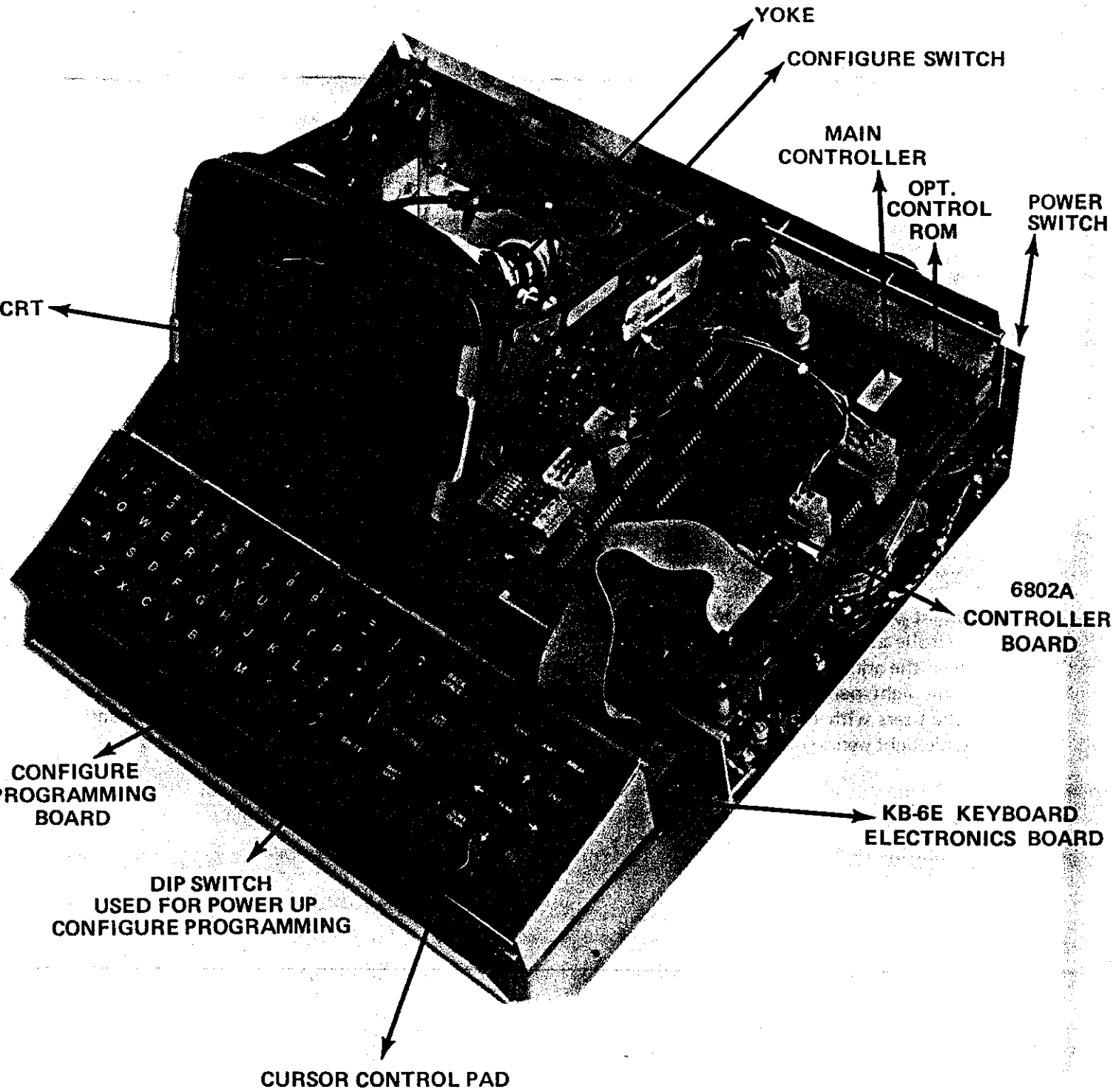
### Light Pen Input

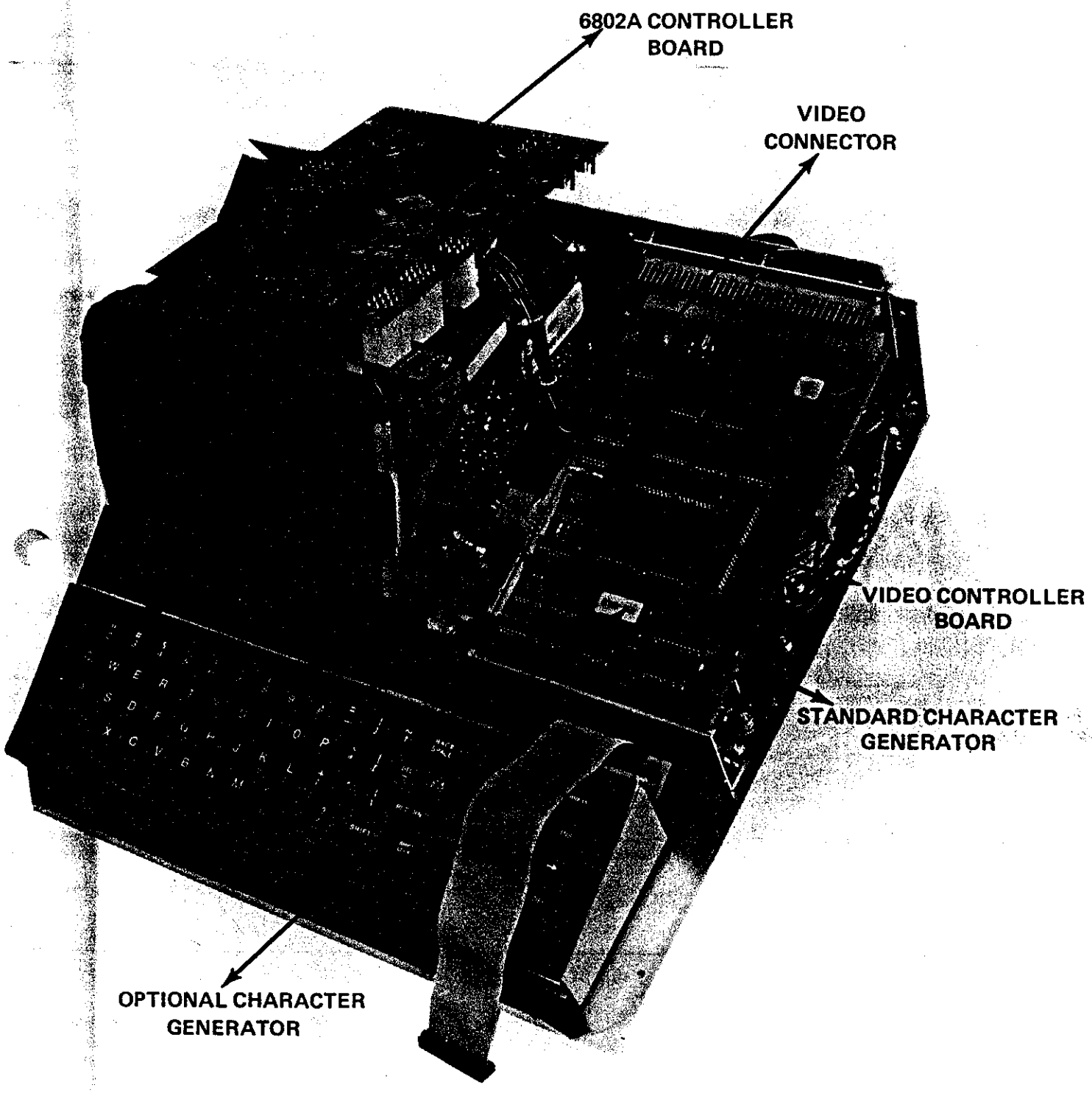
The hardware of the video controller board has been designed to allow connection of a light pen. Light pin input (3-pin connector on video controller board) should be a TTL compatible low going strobe at least 100 nS wide. Using the Read Light Pen Register command (see Appendix A) will yield the address of the refresh buffer at the time the light pen stobe went low.

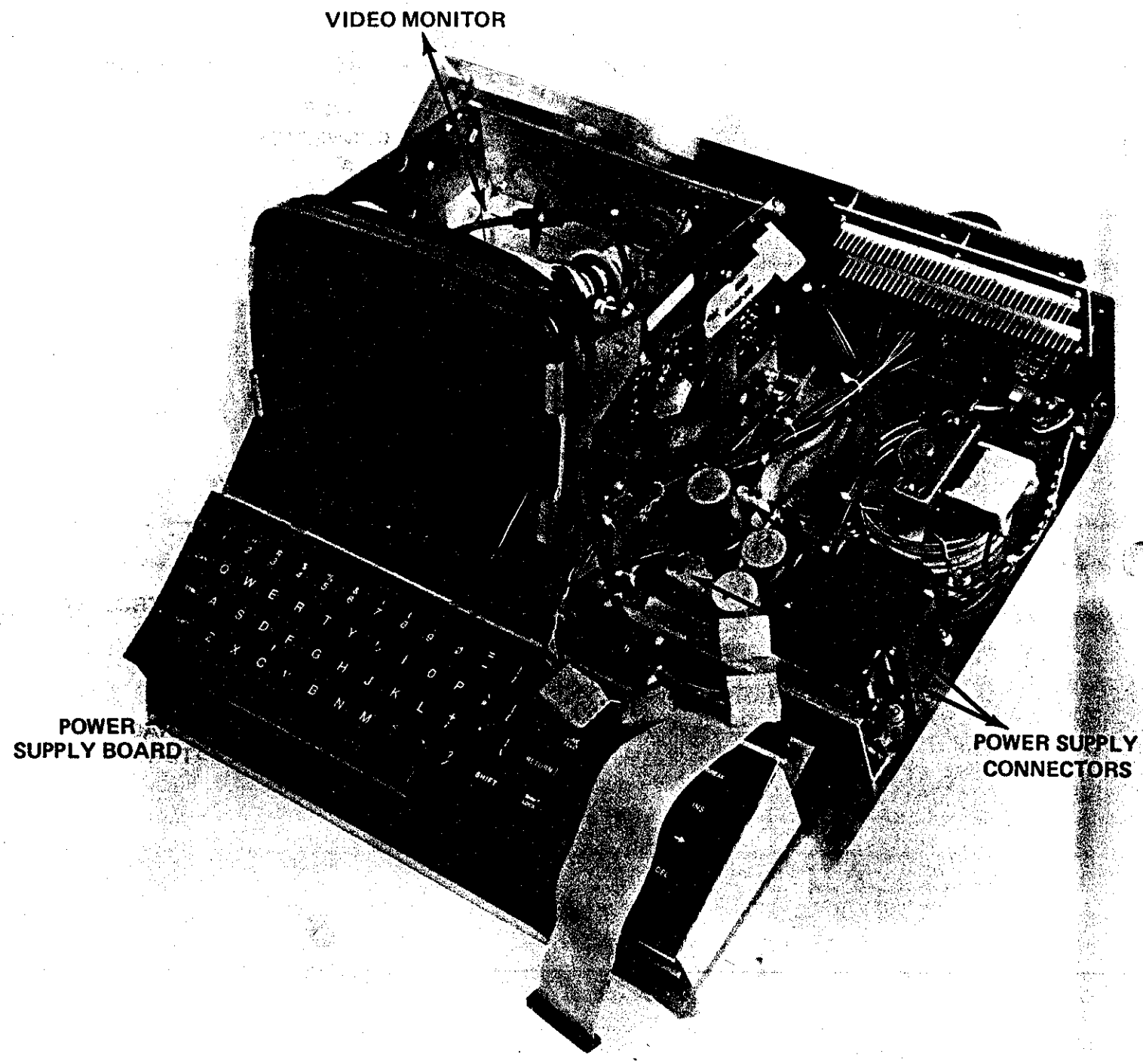
This light pen input is provided for convenience in custom applications and should be used only the users with the proper technical background. SWTPC does not sell and can not recommend a suitable light pen.

## APPENDIX Y

### CT-82 Internal Photographs




6802A CONTROLLER BOARD
VIDEO CONNECTOR
VIDEO CONTROLLER BOARD
STANDARD CHARACTER GENERATOR
OPTIONAL CHARACTER GENERATOR


VIDEO MONITOR
POWER
SUPPLY BOARD
POWER SUPPLY
CONNECTORS

# APPENDIX Z
# VERSION A1 AND VERSION B1 INCOMPATIBILITIES

The CT-82 control ROM has been changed to enhance the terminal. The CT-82 with the old ROM is called the version A1 while the CT-82 with the new ROM is called the version B1 terminal. Some of the changes introduced differences between the two terminals. These differences are listed below. All revisions to this manual are marked with a vertical bar in the margin.

**Hardware Modifications**

- To allow the keypad to be configurable between cursor control and numeric, a jumper needs to be installed between pin 19 of connector J3 and pin 19 of IC8 on the 6802A controller board. Version A1 terminals will not operate correctly with the jumper installed and Version B1 terminals require it to be in place.
- Another required modification consists of cutting pin 8 of the keyboard connector loose from its ground connection and running a jumper wire from IC8 pin 17 to pin 8 of the keyboard connector.
- For 50 Hz operation, the Version B1 terminal may be modified by disconnecting pin 39 of IC5 (on the 6802A controller board) from +5 volts and reconnecting it to ground.

**Configuration Changes**

- The CONFIGURE switch on the back panel has been changed to become the DUPLEX switch. Whether it is labeled CONFIGURE or DUPLEX, the Version B1 terminal uses the switch to select between half and full duplex. The old PROGRAMMABLE setting is the new FULL DUPLEX setting and the old AUTO setting is the HALF DUPLEX setting.
- Reset (Power–On only) on the Version A1 terminal either used the DIP switch or a default to determine the reset configuration depending on the CONFIGURE switch. The Version B1 terminal always uses the DIP switch to determine the terminal's reset (Power-On or Programmable) configuration.
- The parity used by the Version A1 terminal was restricted to 8 bits of data with no parity bit and one stop bit. The parity used by the Version B1 terminals is configurable, depending on the setting of bits 6 and 7 of the DIP switch and on the positions of jumper blocks A and B. The table below describes the available options and how to select them.

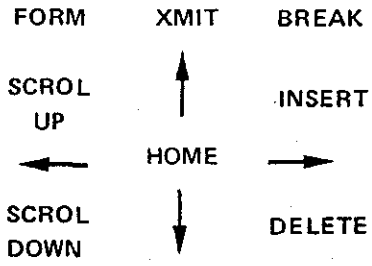
**Selecting the Appropriate Parity**

| BIT 6 | BIT 7 | JUMPER B | JUMPER A | DESCRIPTION |
|---|---|---|---|---|
| ON | – | – | ON | 7-bit data, no parity bit, 2 stop bits |
| ON | – | – | OFF | 8-bit data, no parity bit, 2 stop bits |
| OFF | ON | ON | ON | 7-bit data, odd parity, 2 stop bits |
| OFF | ON | ON | OFF | 8-bit data, odd parity, 2 stop bits |
| OFF | ON | OFF | ON | 7-bit data, parity bit = 1, 2 stop bits |
| OFF | ON | OFF | OFF | 8-bit data, parity bit = 1, 2 stop bits |
| OFF | OFF | ON | ON | 7-bit data, even parity, 2 stop bits |
| OFF | OFF | ON | OFF | 8-bit data, even parity, 2 stop bits |
| OFF | OFF | OFF | ON | 7-bit data, parity bit = Ø, 2 stop bits |
| OFF | OFF | OFF | OFF | 8-bit data, parity bit = Ø, 2 stop bits |

**Changes in Use of Software**

- The Optional ROM Control Sequence control function (1F) provided with the Version A1 terminal has been changed to the Group E Control Sequence control function in the Version B1 terminal.
- The Optional ROM Control Sequence in the Version B1 terminal is 1F ØF.

— The Set Full Duplex control function (1E ØE) provided with the Version A1 terminal has been changed to the Select Cursor Control Keypad control function in the Version B1 terminal. The format of the Cursor Control Keypad is given below.




— The Set Half Duplex control function (1E 1E) provided with the Version A1 terminal has been changed to the Select Numeric Keypad control function in the Version B1 terminal. The format of the Numeric Keypad is given below.

| | | |
|---|---|---|
| 7 | 8 | 9 |
| 4 | 5 | 6 |
| 1 | 2 | 3 |
| Ø | Ø | |

**New Control Functions.**

A fifth group (Group E) of control functions has been added to the Version B1 terminal. The Group E control functions are accessed by preceding the control sequence with the Group E leadin character (1F). The Group E control functions are summarized below.

BACKSPACE, NON-DESTRUCTIVE (^_ ^H)—Backspace the cursor without erasing.

BUMP TO NEXT FIELD (^_ ^L)—Bump to the beginning of the next field. Consecutive characters with the same protection are considered a field.

BUMP TO NEXT FIELD OF SAME PROTECTION (^_ ^M)—Bump to the beginning of the next field having the same protection as the current field.

DISPLAY NUMBER IN HEX (^_ ^K [MSP] [LSP]—Display a 16-bit binary value in hexadecimal format.

FILL SCREEN (^_ ^C)—Fill the screen with rubout characters.

MARK CONTROL CHARACTER (^_ ^D [control character])—Mark the control character for treatment as a special case.

MOVE CURSOR RIGHT, DESTRUCTIVE (^_ ^I [count])—Move the cursor to the right by the amount specified in the count, erasing all characters between the initial cursor position and the final cursor position.

PROGRAMMABLE RESET (^_ ^A)—Reset the terminal to its initial configuration.

RECALL STORED CURSOR POSITION (^_ ^G)—Restore the cursor position saved by a previous Store Cursor Position control function.

RESET TRANSLATE TABLE (^_ ^B)—Reset the translate table to its initial configuration.

SELECT CURSOR CONTROL KEYPAD (^^ ^N)—Characters transmitted from the small keypad will be cursor control characters.

SELECT NUMERIC KEYPAD (^^ ^^)—The small keypad will be treated like an adding machine keyboard.

SET CURSOR POSITION BIAS (^_ ^J [X])—Specify a new value to be used as a cursor position bias.

STORE CURSOR POSITION (^_ ^F)—Store the current cursor position for later use.

UNMARK CONTROL CHARACTER (^_ ^E [control character])—Remove the mark used to specify treatment of a control character as a special case.