Digital Microsystems DMS™

DMS-5000 PROGRAMMER'S MANUAL

Version 1.0

COPYRIGHT

Digital Microsystems
1755 Embarcadero, Oakland, CA 94606
(415) 532-3686 TWX 910-366-7310

NOTICE

Digital Microsystems, Inc. reserves the right to make improvements to the products described in this manual at any time, without notice.

TRADEMARKS

HiNet, DMS-5000, DMS-5080, DMS-5086, DMS-3F, amd DMS-15, are trademarks of Digital Microsystems, Inc. CP/M, and CP/M86, are trademarks of Digital Research, Inc. MX-100 is a trademark of Epson, Inc. Microline-83A is a trademark of Okidata, Inc.

Manual written by:
Robert Weber and Alan H. Stacy

Design & Illustration by:
Deborah Russell

TABLE OF CONTENTS

LIST OF DIAGRAMS

1.0 THE DMS-5000 SYSTEM DESCRIPTION

The DMS-5000 series are general function microcomputer workstations for use with the HiNet local area network. There are two CPUs available in the series. The 5080 has an 8-bit CPU operating under CP/M-80 and the 5086 has a 16-bit CPU using CP/M-86. The unit contains a microcomputer with serial and parallel connections for I/O and network mass storage, a high-resolution bit mapped CRT display with intelligent controller and a detachable keyboard. The host computer and CRT controller are two separate PC boards within the 5000 package that are connected by a handshaking parallel communications port. The CRT and keyboard act as a user console to the host computer.

A DMS-5000 Workstation may be viewed by the programmer as a computer with a separate graphics CRT terminal. This manual is primarily concerned with the use of the CRT console and its features. For programming information on the host computer, consult the HiNet User Manual.

The DMS-5000 CRT, CRT controller and CPU boards are housed in a specially designed case that may be detached from its base and mounted with the CRT screen in either a vertical or a horizontal orientation. A mercury switch on the CRT controller senses the orientation in use and adjusts the character display accordingly.

The CRT uses a P-39, long-persistence phosphor for flicker-free operation. After about 15 minutes of inactivity, a timeout feature will dim the screen intensity down to a level that protects the phosphor from burnout. Touching any key--except Shift, Shift Lock or CTRL--or receiving any code from the host computer will bring the screen intensity back to its original level.

The CRT display is capable of character and fully bit-mapped graphics display plus:

- Variable character sets
- Inverse and underlined characters
- European character sets
- Line drawing
- Point plotting
- Bit-mapped block area fill
- Bit-mapped shape memorization

These various features are accessed by using ASCII Control and Escape character sequences embedded in the console output stream.

The console keyboard has a standard typewriter layout with a separate numeric/cursor -control keypad and a row of sixteen programmable function keys. Except for the ENTER key, all of the keys in the numeric keypad plus the three blank keys in the main keyboard are also programmable. See Diagram 3-1 for an illustration of the keyboard with all of the function keys numbered.

2.0 MODES OF OPERATION

There are four modes for graphics display in the 5000: Character Display, Line Drawing Mode, Point Plotting Mode and Graphic Input Mode. The 5000 can also be set to function in Half Duplex Mode and has a convenient Monitor Mode for debugging.

2.1 HALF DUPLEX MODE

Half Duplex Mode enables the 5000 to send characters to both the CRT screen and to the host CPU or to a modem through either serial port 0 or serial port 3. (The serial port used by communications packages is specified by the program.) This is useful in running communications packages when the 5000 is interfacing with a distant computer over a phone line.

The Half Duplex and Full Duplex Modes are selected with the Local Function keys. To enter Half Duplex Mode, hold down the Shift and Control keys while striking F11. To return to Full Duplex Mode, depress F12 while holding down both the Shift and Control keys.

CTRL/SHIFT F11 -- Half Duplex Mode

CTRL/SHIFT F12 -- Full Duplex Mode

2.2 MONITOR MODE

The DMS-5000 can be put into Monitor Mode by sending an **ESC Q** (1BH,51H) or by pressing

CTRL/SHIFT F4. Once in Monitor Mode, ESC or CTRL sequences will not be processed by the CRT controller but will instead be displayed on the screen as abbreviated codes. This allows the programmer to see a program on the CRT and check for errors. The abbreviations for the codes are listed in Appendix B. To exit Monitor Mode and return to Full Duplex Mode, send **ESC q** (1BH,71H) or press CTRL/SHIFT F7.

ESC Q (1BH,51H)	Enter Monitor Mode
ESC q (1BH,71H)	Exit Monitor Mode
CTRL/SHIFT F4	Enter Monitor Mode from keyboard.
CTRL/SHIFT F7	Exit Monitor Mode from keyboard.

2.3 CHARACTER DISPLAY ON THE 5000 CONSOLE

Upon initialization (power-up or reset condition) the console is set to Character Display Mode. This mode operates in the same way as standard CRT terminals. The alpha cursor, seen as a flashing rectangular block, resides at the current display position. Characters are displayed sequentially as input, left to right, from the starting cursor position. When the cursor comes to the end of a line, it moves to the beginning of the next line. At the end of the screen, the display scrolls up one line, the top line is discarded and display continues. The maximum length and number of lines that can be

displayed at one time depends upon the orientation of the CRT and/or the character size selected.

CHARACTER SIZES

In the vertical orientation the characters are displayed in an 80-column by 66-line format, utilizing a 5x7-bit character mapped into a 7x12-bit character cell. The 5x7 character can be dropped 2 pixels to provide for a descender.

In the horizontal orientation, two character sizes and three "page" lengths are available, giving three screen formats:

An 80 column by 24- OR 26-line format using a large 8x13 bit character (4 bit descender) in a 10x21-bit cell.

A 132-column by 50-line format using a small 5x7-bit character (2-bit descender) in a 6x11-bit cell.

In the horizontal orientation, selection between character formats is accomplished with the following escape codes:

ESC H (1BH,48H) Set 80-col. x 24-line format.

ESC S (1BH,53H) Set 132-col. x 50-line format.

When either of these codes are called, the scroll counters are reset and the proper size alpha cursor will appear in the upper left-hand corner of the display. The screen is not

cleared, therefore characters of both sizes can appear on the screen simultaneously.

In horizontal mode the 5000 can be set to display either the default 24 lines or 26 lines of text without resetting the scroll counters when the format is changed.

ESC ; (1BH,3BH) 80-col. x 26-line format.

ESC : (1BH,3AH) 80-col. x 24-line format.

If you want both character sizes to be displayed at the same time, the screen must not have been scrolled since the last characters were written or interlace inversion--even and odd scan lines are interchanged--will occur. If you do not want both fonts to be displayed, you should still clear the screen (**CTRL L**) before or after changing formats, just to be safe.

2.3.1 VERT. MODE DISPLAY IN HORIZ. ORIENTATION

If the 5000 is operating in horizontal mode, sending an **ESC v** will change the display mode and characters to the vertical orientation even though the CRT remains physically horizontal. To return to horizontal mode, send either **ESC H** for 80-col. x 24-line format or **ESC S** for 132-col. x 50-line format. Remember that ESC H and ESC S will reset the scroll counters.

ESC v (1BH,76H) vertical characters in horizontal orientation.

2.3.2 CLEARING THE SCREEN

There are three codes that can also be used for clearing the screen. **CTRL L** (0CH) will clear the entire screen and home the cursor. **ESC K** (1BH,4BH) will clear all characters after the cursor position to the end of the current line. **ESC k** (1BH,6BH) will clear all characters after the cursor position to the end of the screen.

CTRL L (0CH)	Clear entire screen, home cursor.
ESC K (1BH,4BH)	Clear to end of line.
ESC k (1BH,6BH)	Clear to end of screen.

2.3.3 POSITIONING THE CURSOR

The alpha mode cursor may be positioned on the screen by the following commands:

CTRL A (01H)	Homes cursor to upper left position of screen.
CTRL L (0CH)	Clears screen and homes cursor.
BACKSPACE (**CTRL H**) (08H)	Moves cursor one column to the left.
CTRL F (06H)	Moves cursor one column to the right.
LINEFEED (0AH) **CTRL J**	Moves cursor one line down.

CTRL Z (1AH) Moves cursor one line up.

<CR> (0DH)
(CTRL M) Moves cursor to beginning of current line.

ESC Y (1BH,59H) **[row+20h,col+20h]**
Sets absolute cursor position.

Cursor position starts from col 0, row 0.

To calculate the absolute cursor position:

row number(hex) + 20 hex = x coordinate
column number(hex) + 20 hex = y coordinate.

Example: to set cursor to row 4, column 12:

row 4 = 04H column 12 = 0CH

ESC Y (20+04=24) (20+0C=2C)
or
1B 59 24 2C Hex

The cursor may also be positioned on pixel boundaries instead of row and column intersections. See section 2.4, Graphic Line Drawing Mode.

2.3.4 CHARACTER ENHANCEMENTS

The available character attributes are underline and inverse video. These are set with the following codes:

ESC R (1BH,52H) Turns on inverse video for all succeeding characters.

No BLINK

No HALF-BRIGHT

ESC U (1BH,55H)	Turns on underline mode for all succeeding characters.
ESC O (1BH,4FH)	Turns on character overstrike mode. Allows characters to be written over by another character.
ESC N (1BH,4EH)	Returns all succeeding characters to normal. (Turns off ESC R, ESC U and ESC O.)
ESC T (1BH,54H)	Toggles whole screen inverse video on/off.

2.3.5 CURSOR DISPLAY CONTROLS

Upon power up, the character-mode cursor (alpha cursor) appears as a flashing block the size of a character cell. (The vertical-mode cursor is about 3/4 the size of a character cell.) The cursor can be set not to flash or it can be turned off altogether.

The cursor display codes are:

ESC D (1BH,44H)	Makes the cursor invisible.
ESC E (1BH,45H)	Makes the cursor visible again.
ESC f (1BH,66H)	Non-flashing cursor.
ESC F (1BH,46H)	Flashing cursor.

When Local mode is entered, the cursor changes from a flashing block to a non-flashing

block. The cursor will return to flashing mode when Local mode is exited. If the cursor is moved fast enough, it will time out and become invisible until it stops for 0.5 seconds.

See page 2-21, section 2.4, for important information on cursor positioning when exiting Graphic Line Drawing Mode.

2.3.6 SCROLLING CONTROL

AN IMPORTANT NOTE ABOUT SCROLLING

To increase efficiency and speed of scrolling on the bit-mapped display, hardware scrolling counters have been incorporated into the CRT controller design. These counters point to the address in the screen memory where the screen scan will begin. Thus, to scroll the screen one line up, the counters are loaded with the address of the beginning of the next character line from the top. When the screen is cleared, or the character format is changed, the scroll counters are reset to the original positions.

The DMS-5000 CRT uses an interlaced-field display scheme. The scanning beam must make two complete passes to refresh the screen entirely. The first pass scans the even lines and the second pass scans the odd lines. Television works in much the same way.

The combination of hardware scrolling with interlaced raster scan may cause problems for the unwary programmer. For instance, if the

scrolling counters are reset after the screen has been scrolled, the characters that have been written after scrolling will jump to the top of the screen, and their scan lines will be inverted, giving the text a very jumbled appearance.

Another possible problem can occur because all graphics functions assume that the screen has not been scrolled since entering graphics mode. Any graphics display written to a scrolled screen will appear to be wrapped around the screen.

These problems can be completely avoided by following these rules:

1. Clearing the screen (**CTRL L** (0CH)) will reset the scroll counters. Always clear the screen before using any graphics functions.

2. Changing between large-character (80 x 24) mode and small-character (132 x 50) mode will reset the counters. Clear the screen when changing formats. If you want both character sizes to appear on the screen, don't clear the screen between fonts, but make sure that the screen is not scrolled before changing. (Do not write beyond the line limit of the format you are using.)

3. To avoid these problems while creating complex displays using graphics and mixed text fonts, you must first clear the screen (**CTRL L** (0CH)), and turn the scrolling off (**ESC (** (1BH,28H)) before putting anything on the screen. Be sure to turn the scrolling back on

(**ESC)** (1BH,29H)) before the end of your application.

Scrolling may be disabled when it is necessary to combine graphics and text or, when in the horizontal orientation, two different character sizes must show up on the same screen. Whenever complex displays using both text and graphics are to be drawn, it is advisable to clear the screen first and disable scrolling. This will guard against interlace inversion and graphics wrap-around.

Scrolling commands:

ESC ((1BH,28H) Turn scrolling off. (Home cursor at end of screen.)

ESC ! (1BH,21H) Scroll bottom three lines only.

ESC) (1BH,29H) Turn scrolling on. (Return to normal scrolling mode.)

2.3.7 ALTERNATE CHARACTER SETS

Various character sets are implemented in the DMS-5000 Series. Several sets of standard European characters are included in the character generator. To access these sets send:

ESC L (1BH,4CH) [**Character set code**]

Character set codes:

A = ASCII	**H** = DUTCH (HOLLAND)
B = BRITISH	**I** = ITALIAN
D = DANISH	**N** = NORWEGIAN
F = FRENCH	**S** = SWEDISH
G = GERMAN	

2.3.8 CHARACTER SET FORMATS

An entirely new character set, whether supplied with the 5000 by DMS or designed by the user, may be loaded into the CRT controller from the host computer. An Escape code is sent, followed by a two-byte count for the absolute length of the set, and then the bit maps for the 96 printing characters in ASCII beginning with 20H. See the Data Structures for ASCII characters in Appendix (A) for the actual format of a down-loadable character set.

The code for loading a character set is:

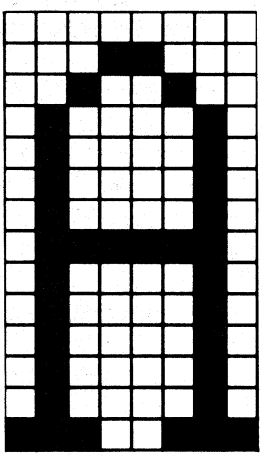
ESC c (1BH,63H) [**Download character set**]

[download: least significant bit to most significant bit]

The length and format of a character set depend upon the current display orientation and the selected character size. Each set consists of 96 characters, with a given number of bytes in its bit map. The character sets can be categorized into three types: horizontal large format, horizontal small format and vertical format.

HORIZONTAL LARGE FORMAT CHARACTERS

The horizontal large-format character set is a block of 1344 (decimal) bytes. Each character set is broken down into 96 characters that are each 14 bytes in length. The characters are written from these bit maps from the top down. The far left pixel in each scan line is the most significant bit in the byte. A capital "A" would look like this:

PIXEL PATTERN	HEX CODE
	00H 18H 24H 42H 42H 42H 42H 7EH 42H 42H 42H 42H 42H E7H

A line in assembly code for this character would look like this:

```
DB
00H,18H,24H,42H,42H,42H,7EH,42H,42H,42H,42H,42H,I
;A
```

To assemble a character set that can be downloaded from the host, 96 of these lines

would be written for the printing codes in the ASCII sequence, beginning with the SPACE character, 20H. (This character is usually filled with all zeros.)

A four-byte header, including the ESC code and the character set length in bytes, is placed in the beginning of the file to let the CRT controller know that a character set is to be loaded.

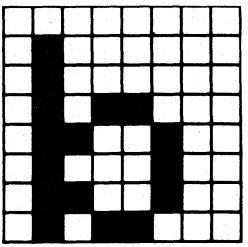
```
DB      1BH,63H          ; ESC c - LOAD CHAR. SET
DW      1344             ; CHAR. SET LENGTH.
```

This file is assembled and made into a binary image file. When this binary image is transmitted to the console, the character set is loaded into the controller's memory. From that point on it is used until: a new character set is loaded, the screen format is changed, or the work station is reset.

HORIZONTAL SMALL-FORMAT CHARACTERS

The horizontal small-format character set is a block of 768 (decimal) bytes. It is broken down into 96 characters, each 8 bytes in length. The characters are written from these bit maps from the top down, again with the most significant bit representing the far left pixel in each scan line.

A lower case "b" would look like this:

PIXEL PATTERN	HEX CODE
	00H 40H 40H 58H 64H 44H 64H 58H

NOTE---The upper row of pixels and the right-most two columns of pixels must ALWAYS remain blank, or the characters will run into each other. This is actually a 5 x 7 matrix.

The assembly code line for this character would be:

```
DB      00H,40H,40H,58H,64H,44H,64H,58H
```

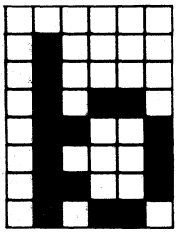
A character set load file may be created in the same way as a large-format file as shown before except that the length count will be different.

```
DB      1BH,63H          ; ESC c - LOAD CHAR. SET
DW      768              ; CHAR. SET LENGTH.
```

VERTICAL-FORMAT CHARACTERS

The vertical-format character set is a block of 576 (decimal) bytes. It is broken down into 96 characters, each 6 bytes in length. The characters are written from these bit maps from LEFT TO RIGHT, with the most significant bit in each byte representing the top pixel in each scan line. Thus, a lower case "b" would look like this:

PIXEL PATTERN



```
0 7 0 1 1 0
0 F A 1 1 E     HEX CODES
H H H H H H
```

The code line for this is:

```
DB      00H,7FH,0AH,11H,11H,0EH
```

Note the difference in coding scheme from the horizontal characters. Since the screen is turned 90 degrees, the scan lines are now running up and down, rather than left to right. The coding for the characters must be compatible with this.

The header for this file would be:

```
DB      1BH,63H         ; ESC c - LOAD CHAR. SET
DW      576             ; CHAR. SET LENGTH.
```

See Appendix A for examples of these structures.

2.3.9 CHARACTER POSITIONING ON THE SCREEN

HORIZONTAL MODE

The first character displayed on the screen after a clear screen command is set flush with the left- hand side of the screen, 6 pixels down from the top. This places the UPPER RIGHT hand corner of the first character at coordinate 0,553 (x,y). This positioning is consistent with either small or large character sets. To find the pixel position of any alpha mode character cell, use the following conversions.

FOR LARGE CHARACTERS:

All values start at 0.
Cell is 10 pixels wide by 21 pixels high.

x = col x 10 **y = 553 - (row x 21)**

FOR SMALL CHARACTERS:

Cell is 6 pixels wide by 11 pixels high.

x = col x 6 **y = 553 - (row x 11)**

In these equations (x,y) are the screen coordinates of the upper-left hand pixel of the character cell, and (row,col) are the alpha mode character positions.

NOTE---These coordinate values represent the upper left-hand pixel of the cell. They do NOT directly correspond to the origin of a character positioned using graphics mode in the Tektronics 4010 emulation. When you position a character at a non-standard coordinate by entering graphics mode, issuing a coordinate, and reentering character mode, the coordinate issued points to the pixel 12 points down from the top of the character cell, on the left hand side. This point approximates the LOWER LEFT-HAND SIDE of the character cell (a little higher for the large characters, a little lower for the small characters), the same way a 4010 does. Thus, if your goal is to pixel-position characters relative to alpha-position character cells, the above formulas apply, except that the y origin is equal to 541. (553-12)

Diagram 2-1. Shows characters and sample positioning coordinates in horizontal and vertical display mode.

VERTICAL ORIENTATION

In the vertical orientation, the first character cell (the character in the upper left hand corner of the screen) is set flush with the top left of the screen. The same rules that govern character positioning in horizontal mode apply in vertical mode.

This formula calculates the pixel position for the alpha character cell in vertical mode:

x = col x 7 y = 799 - (row x 12)

All values start at zero.
Cell is 7 pixels wide by 12 pixels high.

2.3.10 DESCENDER TABLES

If a new character set is loaded into the CRT controller a **descender table** should accompany it. A descender table is a string of 96 bytes with each byte representing a character of the character set. If a byte is set to 0, the character will be displayed on the base line. If the byte is set to anything but 0, such as 1 or ff, the character will be dropped down two or four pixels so that part of it will be below the base line. An "e" would have a 0 descender byte and a "p" or "q" would have a 1 descender byte.

The code for loading a descender table is **ESC** ' (1BH,27H), followed by the 96-byte sequence. If a new character set has descending characters in the same position as the default ASCII set, then it is not necessary to create and load a new table.

2.3.11 TRANSPOSITION TABLES

In some applications using custom character sets, it is desirable to change the codes coming from the keyboard to accommodate different keyboard layouts. For this purpose, a loadable

transposition feature has been implemented. A table, stored in RAM, is used to translate each code coming from the keyboard into an alternate code. For a normal ASCII keyboard, a default table would be the binary numbers 0 through 127, in effect transposing each code for itself. This default table is pre-initialized on startup and when the 5000 is reset or sent an **ESC x.**

An **ESC w** will prompt the host CPU to load a 128-byte transposition table into RAM. If, for example, the upper-case characters "A" and "Q" were to be transposed, the 66th byte would become 81 (decimal), and the 82nd byte would become 65. Since the transposition table is only 128 bytes long, the numeric keypad cannot be changed with this feature.

ESC w (1BH,77H)	**[128 byte sequence]** Load keyboard transposition table.
ESC x (1BH,78H)	Reset transposition table to default values.

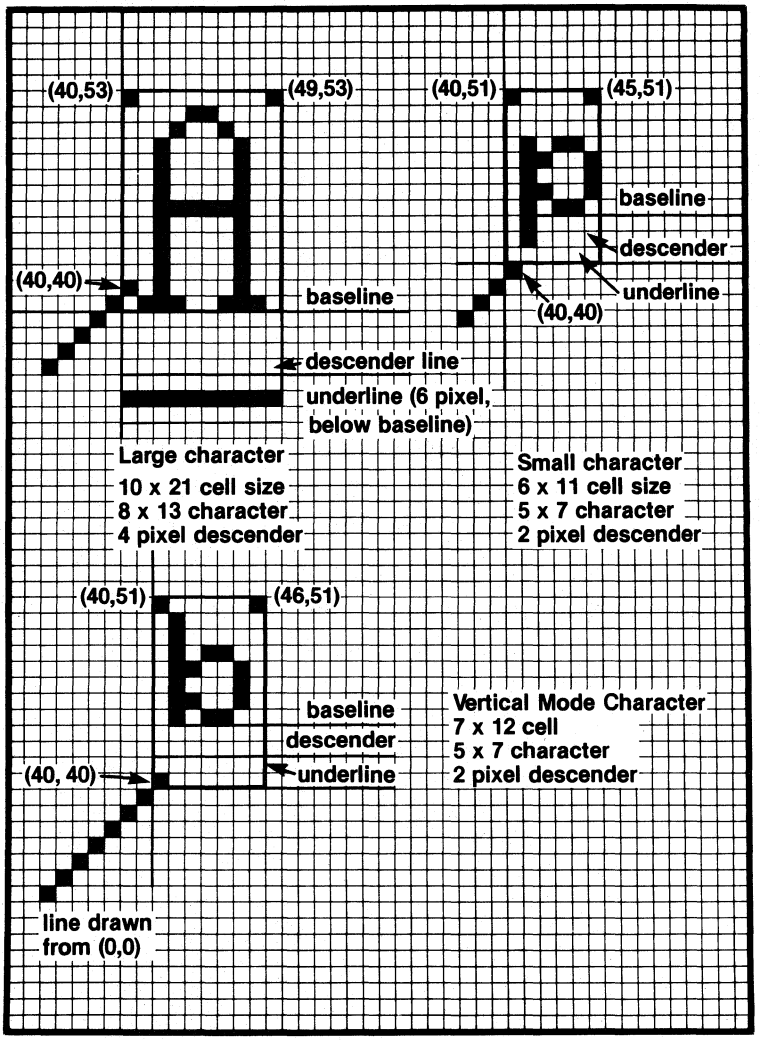


Diagram 2-1. Character positioning and sample coordinates. Note use of graphic line drawing mode to position characters. (See Section 2.4)

2.4 GRAPHIC LINE-DRAWING MODE

Upon receipt of the code **CTRL]** (1DH) the 5000 will enter graphic line-drawing mode. This mode is a direct emulation of the line-drawing mode in the Tektronix 4010 graphics terminal.

The screen coordinates are transmitted as a sequence of four bytes. The first pair of bytes defines the Y position and the second pair the X position. Within each byte, the lower-order 5 bits determine the coordinate value and the upper 2 bits are tag bits that mark which part of the coordinate--high or low--the byte is. A maximum of ten bits are available to be used as coordinate values.

NOTE---The absolute positions of X and Y screen coordinates are dependent upon the CRT's orientation. A pixel coordinate given for a horizontal screen will point to a different location on the screen in vertical mode. For example, the approximate center position on a horizontal screen is pixel 390,280. This would be a point that is 110 pixels down and 110 pixels to the right of the center (280,390) of the screen in vertical orientation. (Refer to Diagram 2-3.)

The bytes are transmitted in the sequence: High Y, Low Y, High X, Low X. (See Diagram 2-2.) After the **CTRL]** and the initial four bytes have been sent to the 5000 CRT controller, additional bytes that do not change--except for the Low X byte--need not be sent. If the High X byte is changed then you MUST send the Low Y byte. The

Low X byte must be sent each time to draw the point or vector.

Once the line-drawing mode is entered, the first coordinate pair received is treated as a "dark vector," that is, a starting point for subsequent plotting. For each pair sent after that, a line will be drawn from the last point received to the current coordinate. A non-plotting vector may be made by giving the plot-mode code again (**CTRL]** (1DH)), and sending a new starting coordinate. Line-drawing mode is exited and character mode entered by sending a **CTRL ^** (1EH).

By using the line-drawing mode, you can place a character anywhere on the screen, not only at row and column positions. Once in line-drawing mode, the four-byte coordinate pair for the desired position of the character on the screen is sent. The lower left-hand corner of the character, descender excluded, is placed at the pixel indicated by the coordinates. The character's position can also be defined by using the graphics mode report format. (Refer to section 2.3.9 for specific information on positioning characters on the screen.)

Once the coordinates have been sent, exiting back to character mode will leave the alpha cursor at the position of the last coordinates given. Characters can then be entered at that point. The next time a carriage return is entered, the alpha cursor will home to the top left of the screen. Therefore, after leaving graphic mode, a carriage return should always be issued before row and column addressing.

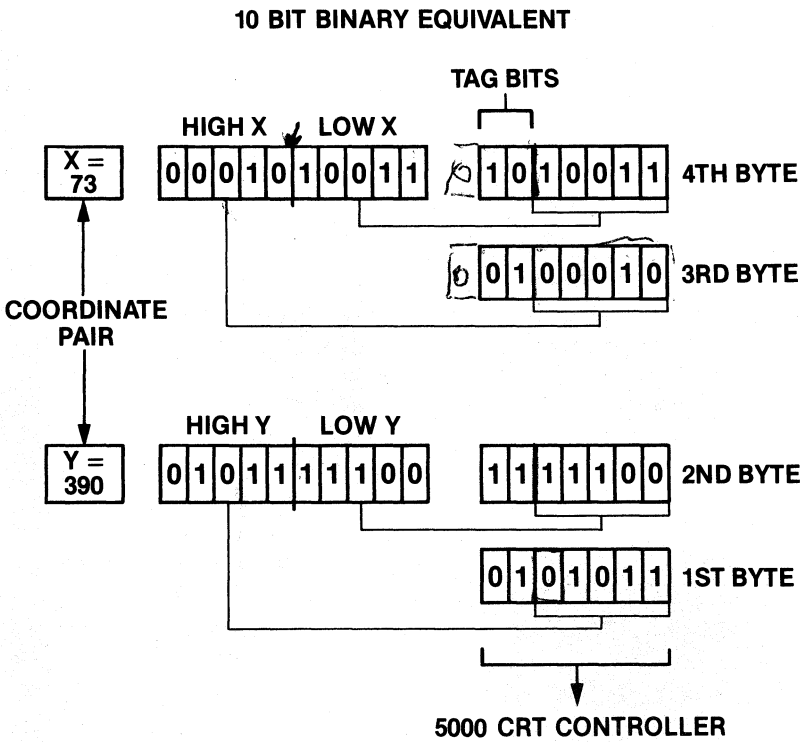


Diagram 2-2. Report format for the cursor position coordinates in graphic line-drawing mode.

The pixel limits of the DMS-5000 screen are 0 to 799 (decimal) on the X-axis, and 0 to 559 on the Y-axis. The lower left-hand corner of the screen (in both orientations) is position (0,0). In the interest of compatibility with the 4010, all incoming coordinates are scaled down to fit on the 5000 screen. For example, a line drawn from the lower-left corner to the upper-right corner would be transmitted as (0,0) to (1023,780) since the limits of the 4010 screen are 0 to 1023 (X) and 0 to 780 (Y).

If a graphic application is being specifically written for the 5000, it is advisable to do away with scaling altogether. Scaling takes extra time and, because of round off error, it is difficult to tell exactly what coordinates will appear on the screen. If there is no reason to use the 4010 screen coordinate limits, coordinate scaling may be disabled or enabled with the following codes:

ESC # (1BH,23H) Disable coordinate scaling.

ESC % (1BH,25H) Enable coordinate scaling.

Do not exceed the coordinate limits. These are:

0 - 1023 for X and
0 - 780 for Y with scaling enabled.

0 - 799 for X and
0 - 559 for Y with scaling disabled.

Bit-mapped shapes that are too large for the screen limits will be displayed wrapped around the screen.

NOTE---ESCAPE and CONTROL codes may be used in graphic modes as well as character modes.

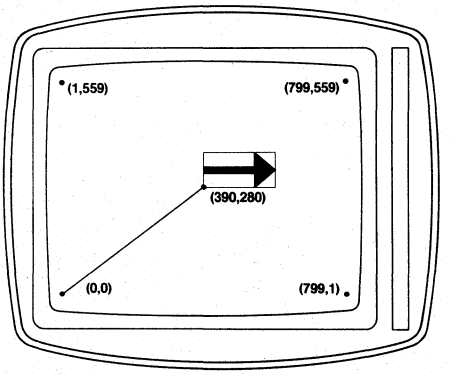
2.4.1 SHAPE DISPLAY AND SCREEN ORIENTATION

When the 5000 is mounted in the vertical orientation, graphics coordinates that were originally prepared for a horizontal screen will still have their origin at the lower left hand corner of the screen. The CRT controller will rotate the coordinates to fit in the vertical screen. The lower left-hand corner will be (0,0) and the upper-right corner will be (559,799). However, the shape's orientation will still be the same as it was in horizontal mode. Diagram 2-3 illustrates this display feature. Any lines with either or both end points off the screen will be rejected and will not be displayed.

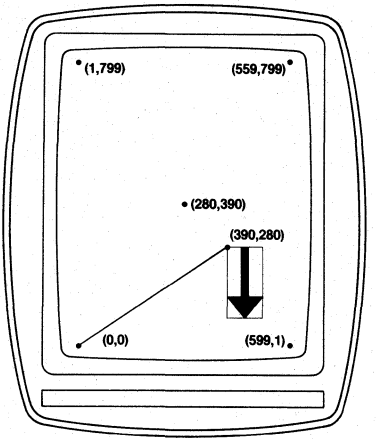
2.4.2 STATUS ENQUIRY

If a graphic application is being written that is to be used in either orientation, a STATUS ENQUIRY may be made of the 5000 that will tell, among other things, the orientation in which that the display is currently mounted.

ESC ENQ (1BH,05H) Transmit status to host.



Horizontal



Vertical

Diagram 2-3. Shape fill is independent of screen orientation. Note how coordinates are rotated to fit screen. (All coordinates are approximate.)

Upon receipt of the above escape code, the 5000 will transmit 6 bytes to the host controller. These are to be interpreted as follows:

1. Status byte
(x,y) Cursor Position
2. High X (5 bits)
3. Low X (5 bits)
4. High Y (5 bits)
5. Low Y (5 bits)
6. Carriage Return

The status byte is read:

Bit 0- Orientation
Vertical = 1
Horizontal = 0

Bit 1- Video Display
Inverse Video = 1
Normal Video = 0

Bit 2- Mode
Graphic = 1
Character = 0

Bit 3- Unused

Bit 4- Unused

Bit 5- Always 1

Bit 6- Always 0

Bit 7- Always 0

2.4.3 SCREEN WRITE MODES

When using any of the graphic functions such as line drawing, point plotting, or block filling, the logic with which the pixels are plotted onto the screen may be set to one of three modes. These modes may be selected with the following escape codes:

ESC | (1BH,7CH) "OR" Plot mode- each point written to the screen will be written as a light point. (This mode is the power-up default.)

ESC & (1BH,26H) "AND" Plot mode- each point written to the screen will be written as a dark point.

ESC ~ (1BH,7EH) "XOR" Plot mode- each point written to the screen will be the complement, at that point, of whatever was previously on the screen. (E.g., a light point will become a dark point.)

2.5 POINT PLOT MODE

Point-plotting mode is entered using the code **CTRL ** (1CH). Coordinates for point-plotting are formatted in the same way as in the line-drawing mode and adhere to the same rules concerning scaling and screen limits. One point is plotted onto the screen for each four byte coordinate pair received. The code **CTRL ^** (1EH) returns the CRT to character mode.

2.6 GRAPHIC-INPUT MODE

Graphic-input mode is implemented as an emulation of the Tektronix 4010 GIN mode. To enter this mode, transmit the codes **ESC SUB** (1BH,1AH). When this command is received by the CRT controller, the alpha cursor is replaced by a full-size crosshair cursor. This cursor may be

moved using the cursor control keypad on the right hand side of the keyboard. Striking any of the arrow keys will move the crosshairs one pixel in the given direction. The cursor can be moved diagonally by using the corner keys (7,9,1,3). Holding down the shift key while striking a cursor key will move the cursor 10 pixels at a time.

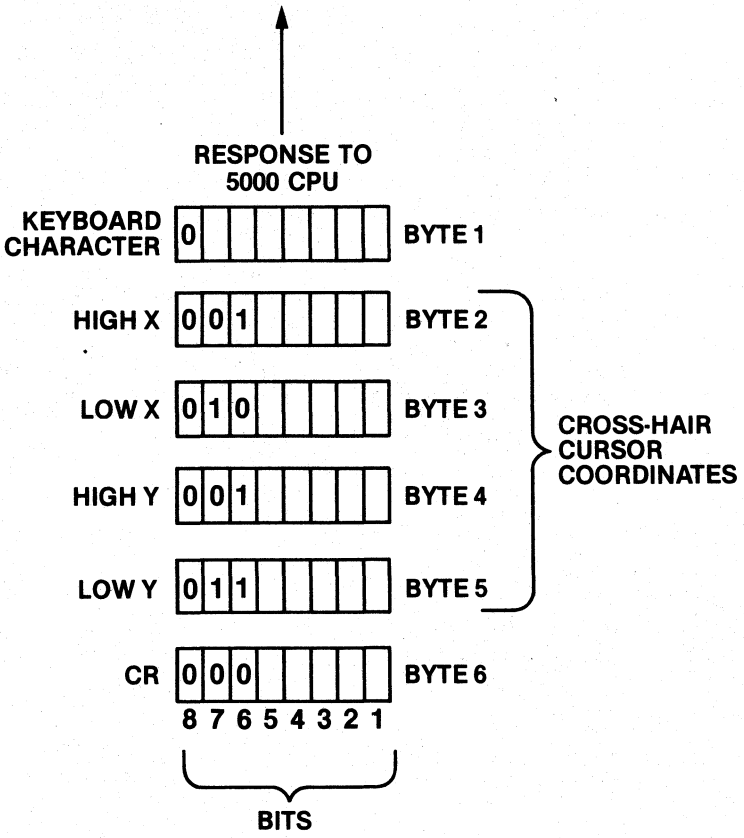
All of this cursor movement goes on transparent to the host computer until any ASCII key is struck. The CRT controller will then send a six byte sequence to the host computer reporting the key struck and the position of the crosshair cursor. The report format is as follows:

Byte 1 - Keyboard character
Byte 2 - High X
Byte 3 - Low X
Byte 4 - High Y
Byte 5 - Low Y
Byte 6 - Carriage Return

Diagram 2-4 illustrates the report format for the DMS-5000 in graphic input mode.

These coordinates are reported in the 4010 format and scaled to 4010 screen proportions. The scaling may be disabled using the commands described in Section 2.4, Line-Drawing Mode.

After the key that reports the cursor position is struck, the 5000 controller will exit graphic input mode and return to alpha



REPORT FORMAT

CPU: ESC SUB
OPERATOR: 1) SETS CROSS-HAIR CURSOR
2) STRIKES KEY TO REPORT CURSOR POSITION

Diagram 2-4. The six-byte report format for the crosshair cursor position in Graphic Input Mode.

mode. If the controller requires more data, according to the applications program, it will return to graphic input mode. This sequence is the same as for the Tektronics 4010.

By using an ESC code you can control the full screen crosshair cursor with an applications program. This will allow the control of cursor movement by means other than the cursor arrow keys on the 5000 numeric keypad (e.g., with a graphics pad or a trackball interfaced to the host computer).

To use this feature send:

ESC G (1BH,47H) **4 byte coordinate** (4010 format)

This will result in the graphic cursor moving to the four-byte coordinate location.

2.6.1 BIT MAP BLOCK SCREEN FILL

The CRT screen can be loaded with a bit-mapped picture transmitted as a stream of bytes from the host processor. (See example in section 2.6.3.)

The first (x,y) coordinate transmitted is the lower left-hand corner of the block on the screen to be filled. The following bytes are loaded onto the screen from left to right starting at that initial point. When the width limit is reached, the fill commences on the next line up. The CRT controller keeps count of the incoming bytes and returns normal control to the

host processor when the block is filled. The transmission sequence is as follows:

1	**ESC X** (1BH,58H) Block fill code.
2	**Y coordinate**--most significant byte.
3	**Y coordinate**--least significant byte.
4	**X coordinate**--most significant byte.
5	**X coordinate**--least significant byte.
6	**Vertical height in pixels**--most significant byte.
7	**Vertical height in pixels**--least significar byte.
8	**Horizontal width in bytes (x 8 pixels)**--one byte.
9-N	**Bytes of bit mapped block.**

Each byte transmitted equals 8 pixels--1 bit equals 1 pixel. In the default write mode, a 0 bit would be written as dark point and a 1 bit would be a light point. (See section 2.4.3) The first byte--a string of 8 pixels--is sent to the CRT at the lower left-hand corner of the pre-defined block. Each succeeding byte fills in the block from the left to the right until the first row is filled. The next byte begins one row up from the first, starting again at the left corner.

2.6.2 SHAPE DISPLAY AND SCREEN ORIENTATION

The direction and sequence in which a shape will fill on the screen is independent of the screen orientation. A shape will always fill as if it was being displayed in horizontal mode. Therefore if a shape is to be displayed in vertical mode, the block fill will begin in the

upper left corner of the shape and fill downward until it reaches the bottom of the shape. The fill will then return to the top of the shape, one pixel to the right of the first column, and begin filling downward again.

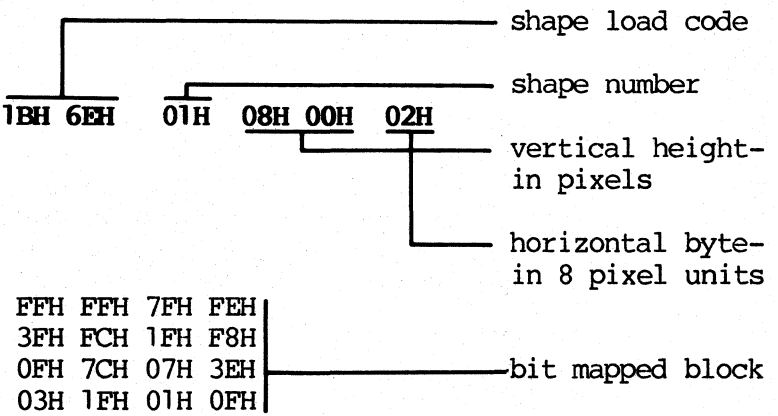
Remember that even though the coordinates are rotated when you change from horizontal to vertical, the scan lines still run in the same direction as in horizontal mode. Therefore the shape will fill as if the screen was still in a horizontal position. (Refer to Diagram 2-3.)

2.6.3 BIT MAPPED SHAPE MEMORIZATION

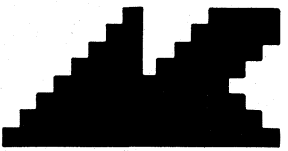
This feature enables the user to store a given shape or symbol in the CRT controller for repetitive use. The shape is loaded into the controller in much the same fashion as a function key string. The sequence to load the shape is as follows:

1-2 **ESC n** (1BH,6EH) Shape load code.
3 **Shape number (1 to 128)**--one byte.
4 **Vertical height in pixels**--least significant byte.
5 **Vertical height in pixels**--most significant byte.
6 **Horizontal width in bytes (x 8 pixels)**--one byte.
7-N **Bytes of bit-mapped block.**

For example, sending the sequence:



will cause this 8 pixel x 16 pixel shape to be memorized:



Once the shape is loaded into memory, the screen coordinates where the user wishes the shape to appear are given by entering the line-drawing mode (**CTRL]**) and sending the four-byte coordinate sequence. To write the shape onto the screen the user will send the following sequence:

ESC @ (1BH,40H) **Shape number**

The shape will fill up and to the right until its size limits are reached. If the shape

is too large for the screen (e.g., a full screen horizontal shape displayed on a vertically-orientated screen), then the shape will wrap around to the other edge of the screen. (Remember, this is not the case in Line Drawing Mode. Lines that have origins or endpoints beyond the screen limits will not be displayed at all.)

Up to 128 shapes may be memorized at a time. Shapes may be UP TO 253 BYTES LONG and no longer. The shape buffer has a storage limit of 3K. If too many large shapes are loaded and the buffer is over-filled, the 5000 cannot be guaranteed to function properly.

2.6.4 PRINTING GRAPHICS

Graphics can only be printed using the screen-dump commands. Only parallel printers with graphics formats similar to the ones specified here can be used for a graphics screen dump.

A screen dump can be initiated from either the keyboard or from an applications program.

From the keyboard use CTRL/SHIFT F8 and F9.

CTRL/SHIFT F8 - Graphics screen dump to Epson MX-100, or equivalent, on parallel port.

CTRL/SHIFT F9 - Graphics screen dump to Okidata Microline 83A, or equivalent, on parallel port.

ESC ETB (1BH,17H) - Graphics screen dump to Epson MX-100 on parallel port.

ESC CAN (1BH,18H) - Graphics screen dump to Okidata Microline 83A on parallel port.

If a printer is not attached to the parallel print port or if the printer is not turned on when the screen dump keys are used, the 5000 will wait for 12 seconds, emit a short beep and then exit screen dump mode.

See Appendix C for information on the parallel port pin assignments.

PRINTING TEXT WITH SERIAL PRINTERS

For information on the 5000's serial and parallel printer ports and HiNet's I/O byte structure, see Appendix C.

3.0 PROGRAMMABLE FUNCTION KEYS

Across the top of the keyboard are sixteen function keys. Each are programmable with up to three separate strings of variable length. In addition, the ten numeric keys, the decimal point key on the numeric/cursor-control keypad and the three blank keys in the main key group, are all programmable. Each key may hold three separate values, one for the key alone, one for the key with the SHIFT key held down, and one for the key with the CTRL key held down. This gives you 90 programmable keys in all. Diagram 3-1 shows you which keys are programmable function keys.

Upon startup, the top row of function keys are initialized with the following HiNet-CP/M commands:

Key	Command	Key	Command
F1	Assign	F9	Load
F2	Dir	F10	Save
F3	Pip	F11	Setbaud
F4	Stat	F12	Settime
F5	Type	F13	Time
F6	Submit	F14	Customiz
F7	Ren	F15	Dirnet
F8	Era	F16	Who

The CTRL and SHIFT values of the sixteen programmable keys are left null.

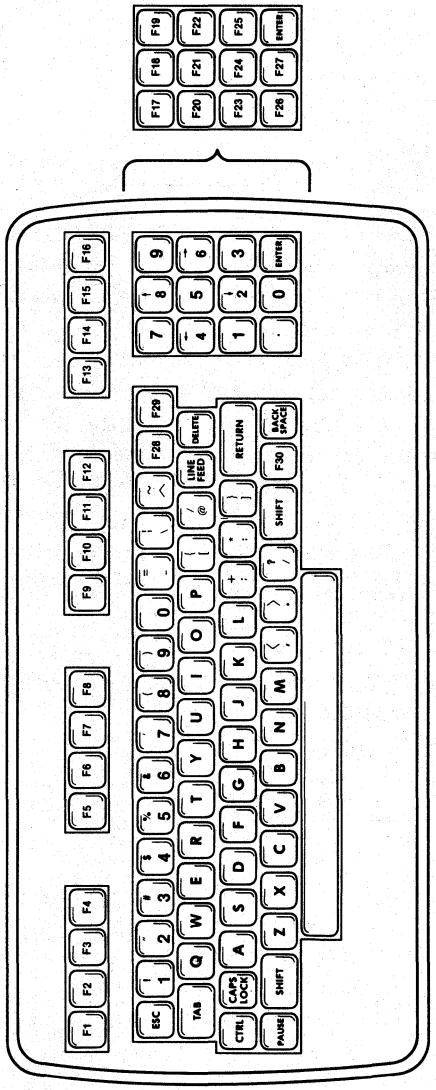


Diagram 3-1. An illustration of the 5000 keyboard with all programmable function keys labele

Also upon startup, the eleven keys of the right-hand keypad are initialized with numeric values according to their legends. (The decimal point is programmed with a period. "ENTER" is a permanent, non-programmable carriage return.)

The shifted values of the arrow keys are programmed with cursor control codes:

UP - CTRL Z (1AH)
DOWN - LINE FEED (0AH)
LEFT - BACKSPACE (08H)
RIGHT- CTRL F (06H)

The control values for the numeric keys are left null.

To reprogram the function key output strings the following code is used:

ESC 1
function key number
length of string
string

3.1 FUNCTION KEY NUMBER

"Function key number" is the one-byte binary identifier of the function key you want to program. To calculate the function key number of a function key, add the hex value of the key (e.g., F1=1H, F16=10H) to 80H. For example, the function key number for F17 is:

17=11H 11H + 80H = 91H F17=91H.

To program the SHIFTED value of a function key, set bit 5 in its function-key number byte to one. To program the CONTROL value of a key, set bit 6 in the byte to one. See Diagram 3-2 for the byte structure of the function key numbers.

To find the function key number for the shifted value of a function key, add A0H to the key number. For example:

SHIFT/F6 = 6 + A0H = A6H

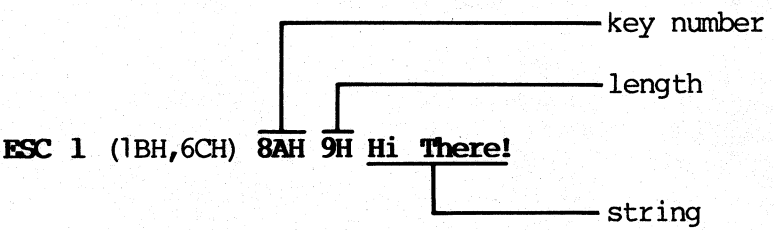
To find the function key number for a CTRL/Function key, add C0H to the key number. For example:

CTRL/F6 = 6 + C0H = C6H

As another example, if you wanted to program key F10 to output the string:

Hi There!

the programming command would look like this:



In BASIC the program code would be:

```
10 PRINT CHR$(1BH);CHR$(6CH);CHR$(8AH);CHR$(9H);
20 PRINT Hi There!;
```

The maximum buffer space allotted to function keys is 1k Hex bytes. Each function key can be programmed with a maximum of 125 characters.

3.1.1 FUNCTION KEY NUMBER REPORT

Normally, when a function key is pressed, the string programmed into it is sent to the host CPU. A mode can be entered in which only the function key number (in hex) will be reported to the host CPU when a function key is pressed. For example, F1 will send 81H and F6 will send 86H.

ESC p (1BH,70H) - Report function key number only.

ESC P (1BH,50H) - Report function key's programmed string.

BYTE STRUCTURE FOR FUNCTION KEYS

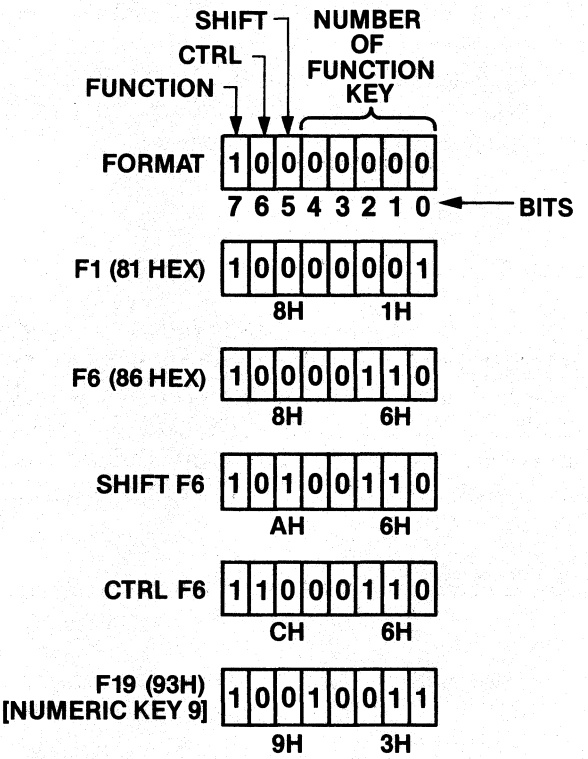


Diagram 3-2. Byte structure of the Function Keys and examples for calculating the Function Key Number.

- For all function keys, bit 7 is set to 1.
- Bit 6 is set to 1 only for the CTRL value of a key.
- Bit 5 is set to 1 only for the SHIFT value of a key.
- Bits 4 through 0 are for the hex value of the function key label.

4.0 DMS-5000 CRT CONTROL CODES SUMMARY

Release Version 1.2

4.1 SCREEN BRIGHTNESS CONTROL

The screen brightness is controlled locally by holding down the control and shift keys while striking a function key as follows:

CTRL/SHIFT F1 Turns screen brightness down one increment.

CTRL/SHIFT F2 Turns screen brightness up one increment.

Alternatively, the intensity may be stepped up or down by the host computer with the following codes:

ESC b (1BH,62H) Turns screen brightness up one increment.

ESC d (1BH,64H) Turns screen brightness down one increment.

4.2 CHARACTER ENHANCEMENT CONTROL

ESC R (1BH,52H) Turns character inverse video on for all following characters.

ESC U (1BH,55H) Turns underline on for all following characters.

ESC O (1BH,4FH) Turns on character overstrike mode.

ESC N (1BH,4EH) Returns all following characters to normal.

ESC T (1BH,54H) Full screen inverse video.

4.3 HORIZ. MODE CHARACTER SIZE SELECTION

ESC S (1BH,53H) Select small characters. (132 x 50)

ESC H (1BH,48H) Select large characters. (80 x 24)

ESC v (1BH,76H) Sets a horizontally orientated screen to display in vertical mode.

4.4 CURSOR DISPLAY CONTROLS

ESC D (1BH,44H) Makes the cursor invisible.

ESC E (1BH,45H) Makes the cursor visible again.

ESC f (1BH,66H) Non-flashing cursor.

ESC F (1BH,46H) Flashing cursor.

4.5 SCROLLING CONTROL

ESC ((1BH,28H) Turn scrolling off. (Home cursor to end of screen.)

ESC) (1BH,29H) Turn scrolling on.

ESC ! (1BH,21H) Scroll bottom three lines only.

4.6 GRAPHIC SCREEN WRITE MODES

These codes determine the screen plotting logic. These modes are used for point-plotting, line-drawing, block-filling and shape memorization functions. The default mode is 'OR'.

ESC |(7CH) "OR" Plot mode--each point written to the screen will be written as a light point.

ESC &(26H) "AND" Plot mode--each point written to the screen will be written as a dark point.

ESC ~(7EH) "XOR" Plot mode--each point written to the screen will be the complement of whatever was previously on the screen.

4.7 BIT MAP BLOCK SCREEN FILL

The transmission sequence is as follows:

1	**ESC X** (1BH,58H) Block fill code.
2	**Y coordinate**--most significant byte.
3	**Y coordinate**--least significant byte.
4	**X coordinate**--most significant byte.
5	**X coordinate**--least significant byte.
6	**Vertical height in pixels**--most significant byte.
7	**Vertical height in pixels**--least significant byte.
8	**Horizontal width in bytes (x 8 pixels)** --one byte.
9-N	**Bytes of bit mapped block.**

4.8 BIT MAPPED SHAPE MEMORIZATION

1-2	**ESC n** (1BH,6EH) Shape load code.
3	**Shape number** (1 to 128)--one byte
4	**Vertical height in pixels**--least significant byte.
5	**Vertical height in pixels**--most significant byte.
6	**Horizontal width in bytes (x 8 pixels)**--one byte.
7-N	**Bytes of bit-mapped block.**

To write the shape onto the screen:

ESC @ (1BH,40H) [**Shape number**]

4.9 CURSOR POSITIONING CONTROL

CTRL A (01H) — Homes cursor to upper left position of screen.

CTRL L (0CH) — Clears screen and homes cursor.

backspace (**CTRL H**) (08H) — Moves cursor one column left.

CTRL F (06H) — Moves cursor one column right.

linefeed **CTRL J** (0AH) — Moves cursor one line down.

CTRL Z (1AH) — Moves cursor one line up.

<CR> (**CTRL M**) (0DH) — Moves cursor to beginning of current line.

ESC Y (1BH,59H) **row+20h,col+20h**
Set absolute cursor position. Row and column number must be in hex.

4.10 ALTERNATE CHARACTER SETS

ESC L (1BH,4CH) **Character set code**

Character set codes:

A = ASCII	**H** = HOLLAND (DUTCH)
B = BRITISH	**I** = ITALIAN
D = DANISH	**N** = NORWEGIAN
F = FRENCH	**S** = SWEDISH
G = GERMAN	

ESC c (1BH,63H) Download character set.

ESC w (1BH,77H) Load keyboard transposition table.

ESC x (1BH,78H) Reset transposition table to default.

ESC ' (1BH,27H) Load character descender table.

4.11 GRAPHIC LINE DRAWING MODE

CTRL [(1DH) Enter line-drawing mode.

CTRL ^ (1EH) Exit Line drawing mode to character mode.

ESC # (1BH,23H) Disable coordinate scaling

ESC % (1BH,25H) Enable coordinate scaling.

(Coordinate scaling is enabled on power up condition

4.12 GRAPHIC-INPUT MODE

In graphic-input mode, the crosshair graphic cursor is displayed. Upon striking a key, the key struck and the current coordinate are reported to the host computer. (See section 2.6 for report format.)

The graphic cursor is controlled from the right-hand keypad. The arrows on the keys represent the direction of cursor movement. Depressing an arrow key alone moves the cursor a distance of one pixel. Holding down the shift key along with the cursor key moves the cursor a distance of ten pixels at one time. The corner keys will move the cursor in a diagonal direction.

ESC CTRL Z (1BH,1AH) Enter graphic input mode

CTRL ^ (1EH) Exit to character mode.

4.13 PARALLEL PRINTER PORT ACCESS

ESC [(1BH,5BH) . Sends only the next character to the printer. ESC [must be sent before every character that is to be printed through the parallel port. Characters are not displayed on the screen.

4.14 DMS-5000 TEST FUNCTIONS

The 5000 has two test functions built in for service purposes. The 5000 must be reset to get out of either of these functions.

ESC ? (1BH,3FH) Extended RAM test.

ESC DEL (1BH,7FH) Display alignment pattern.

4.15 LOCAL FUNCTIONS

Local functions can be accessed from the keyboard without interrupting the application currently running. To use these functions, hold down the CONTROL and SHIFT keys simultaneously and strike a function key as follows:

CTRL/SHIFT F1	Turns screen brightness down one increment.
CTRL/SHIFT F2	Turns screen brightness up one increment.
CTRL/SHIFT F3	Swaps foreground and background intensities. Full screen reverse video.
CTRL/SHIFT F4	Enter Monitor mode.
CTRL/SHIFT F5	Enter Local mode. Cursor stops flashing.

CTRL/SHIFT F6	Exit Local mode. Cursor resumes flashing.
CTRL/SHIFT F7	Exit Monitor mode.
CTRL/SHIFT F8	Graphics screen dump to printer for Epson MX-100 Printer on parallel port.
CTRL/SHIFT F9	Graphics screen dump to printer for Okidata Microline 83A Printer on parallel port.
CTRL/SHIFT F11	Turns on Half-Duplex Mode.
CTRL/SHIFT F12	Turns on Full-Duplex Mode. (Default operation mode.)
CTRL/SHIFT F14	ASCII character set test. (Printable characters 20H-7FH.)

5.0 ESCAPE AND CONTROL CODES INDEX

CONTROL CODES INDEX

CONTROL/SHIFT CODES INDEX

```
;       These listings, when assembled, will create object files that
;       when transmitted to the 5000 console will install a character
;       set equivalent to the default character set on startup.
;
;       This character set is for the vertical orientation.
;
;       These codes initiate the download.
;
        DB      1BH,43H                         ; ESC,c - LOAD CHAR. SET CODE
        DW      VCSLEN                          ; LENGTH OF CHARACTER SET
;
;       Here are the character bit maps, in ASCII order starting with blank.
;
VCSET   DB      00H,00H,00H,00H,00H,00H         ;blank
        DB      00H,00H,00H,00H,7BH,00H         ; !
        DB      00H,00H,70H,00H,70H,00H         ; "
        DB      00H,14H,7FH,14H,7FH,14H         ; #
        DB      00H,12H,2AH,7FH,2AH,24H         ; $
        DB      00H,62H,64H,8H,13H,23H          ; %
        DB      00H,37H,49H,35H,2H,5H           ; &
        DB      00H,00H,00H,70H,00H,00H         ; single '
        DB      00H,1CH,22H,41H,00H,00H         ; (
        DB      00H,00H,00H,41H,22H,1CH         ; )
        DB      00H,22H,14H,7FH,14H,22H         ; *
        DB      00H,8H,8H,3EH,8H,8H             ; +
        DB      00H,00H,02H,0CH,00H,00H         ; comma
        DB      00H,8H,8H,8H,8H,8H              ; minus
        DB      00H,00H,00H,3H,00H,00H          ; .
        DB      00H,2H,4H,8H,10H,20H            ; /
        DB      00H,3EH,45H,49H,51H,3EH         ; 0
        DB      00H,00H,21H,7FH,1H,00H          ; 1
        DB      00H,23H,45H,49H,49H,31H         ; 2
        DB      00H,42H,41H,49H,59H,66H         ; 3
        DB      00H,0CH,14H,24H,7FH,4H          ; 4
        DB      00H,79H,49H,49H,49H,46H         ; 5
        DB      00H,1EH,29H,49H,49H,46H         ; 6
        DB      00H,40H,47H,48H,50H,60H         ; 7
        DB      00H,36H,49H,49H,49H,36H         ; 8
        DB      00H,31H,49H,49H,49H,3EH         ; 9
        DB      00H,00H,00H,6CH,00H,00H         ; :
        DB      00H,00H,1H,66H,00H,00H          ; ;
        DB      00H,8H,14H,22H,41H,00H          ; <
        DB      00H,14H,14H,14H,14H,14H         ; =
        DB      00H,41H,22H,14H,8H,00H          ; >
        DB      00H,20H,40H,4DH,50H,20H         ; ?
        DB      00H,3EH,41H,5DH,4DH,39H         ; @
        DB      00H,1FH,24H,44H,24H,1FH         ; A
        DB      00H,7FH,49H,49H,49H,36H         ; B
        DB      00H,3EH,41H,41H,41H,22H         ; C
        DB      00H,7FH,41H,41H,41H,3EH         ; D
        DB      00H,7FH,49H,49H,49H,41H         ; E
        DB      00H,7FH,48H,48H,48H,40H         ; F
        DB      00H,3EH,41H,41H,45H,47H         ; G
        DB      00H,7FH,8H,8H,8H,7FH            ; H
        DB      00H,00H,41H,7FH,41H,00H         ; I
```

```
        DB      00H,2H,1H,1H,1H,7EH             ; J
        DB      00H,7FH,8H,14H,22H,41H          ; K
        DB      00H,7FH,1H,1H,1H,1H             ; L
        DB      00H,7FH,20H,10H,20H,7FH         ; M
        DB      00H,7FH,10H,8H,4H,7FH           ; N
        DB      00H,3EH,41H,41H,41H,3EH         ; O
        DB      00H,7FH,48H,48H,48H,30H         ; P
        DB      00H,3EH,41H,45H,42H,3DH         ; Q
        DB      00H,7FH,48H,4CH,4AH,31H         ; R
        DB      00H,32H,49H,49H,49H,26H         ; S
        DB      00H,40H,40H,7FH,40H,40H         ; T
        DB      00H,7EH,1H,1H,1H,7EH            ; U
        DB      00H,7CH,2H,1H,2H,7CH            ; V
        DB      00H,7FH,2H,4H,2H,7FH            ; W
        DB      00H,63H,14H,8H,14H,63H          ; X
        DB      00H,60H,10H,0FH,10H,60H         ; Y
        DB      00H,43H,45H,49H,51H,61H         ; Z
        DB      00H,7FH,41H,41H,41H,00H         ; [
        DB      00H,20H,10H,8H,4H,2H            ; /
        DB      00H,41H,41H,41H,7FH,00H         ; ]
        DB      00H,8H,10H,3FH,10H,8H           ; ^
        DB      00H,1,1,1,1,1           ; underline
        DB      00H,40H,20H,10H,00H,00H ; apostrophe
        DB      00H,0EH,11H,11H,0AH,1FH         ; a
        DB      00H,7FH,0AH,11H,11H,0EH         ; b
        DB      00H,0EH,11H,11H,11H,11H         ; c
        DB      00H,0EH,11H,11H,0AH,7FH         ; d
        DB      00H,0EH,15H,15H,15H,0CH         ; e
        DB      00H,00H,8H,3FH,48H,48H          ; f
        DB      00H,38H,45H,45H,45H,7EH         ; g
        DB      00H,7FH,8H,10H,10H,0FH          ; h
        DB      00H,00H,11H,5FH,1H,00H          ; i
        DB      00H,2H,1H,1H,5EH,00H            ; j
        DB      00H,7FH,4H,0CH,12H,21H          ; k
        DB      00H,00H,41H,7FH,1H,00H          ; l
        DB      00H,1FH,10H,0FH,10H,0FH         ; m
        DB      00H,1FH,8H,10H,10H,0FH          ; n
        DB      00H,0EH,11H,11H,11H,0EH         ; o
        DB      00H,7FH,44H,44H,44H,38H         ; p
        DB      00H,38H,44H,44H,7EH,1H          ; q
        DB      00H,1FH,8H,10H,10H,10H          ; r
        DB      00H,9H,15H,15H,15H,12H          ; s
        DB      00H,10H,7EH,11H,11H,2H          ; t
        DB      00H,1EH,1H,1H,1H,1FH            ; u
        DB      00H,18H,6H,1H,6H,18H            ; v
        DB      00H,1EH,1H,2H,1H,1EH            ; w
        DB      00H,11H,0AH,4H,0AH,11H          ; x
        DB      00H,78H,5H,5H,5H,7EH            ; y
        DB      00H,11H,13H,15H,19H,11H         ; z
        DB      00H,00H,8H,37H,41H,00H          ; {
        DB      00H,00H,00H,7FH,00H,00H         ; |
        DB      00H,00H,41H,37H,8H,00H          ; }
        DB      00H,4H,8H,8H,8H,10H             ; ~
        DB      00H,24H,49H,12H,24H,49H         ; DEL
VCSLEN  EQU     OFFSET $-OFFSET VCSET           ; LENGTH OF VERTICAL CSET
```

```
;       This character set is for the horizontal large character format.
;
;       Here is the initialization:
;
        DB      1BH,63H                         ; ESC c, START DOWNLOAD
        DW      HCSLEN                          ; LENTH OF CHAR. SET
;
;       And here is the image:
;
HCSET   DB      0,0,0,0,0,0,0,0,0,0,0,0,0,0                                     ;BLANK
        DB      0,18H,18H,18H,18H,18H,18H,18H,18H,18H,0,0,18H,18H               ;!
        DB      0,14H,14H,14H,14H,0,0,0,0,0,0,0,0,0                             ;"
        DB      0,24H,24H,24H,24H,0FFH,24H,24H,24H,0FFH,24H,24H,24H,24H         ;#
        DB      0,8,8,3EH,49H,48H,48H,3EH,9,9,49H,3EH,8,8                       ;$
        DB      0,0,0E2H,0A2H,0E4H,4,8,8,10H,10H,27H,25H,47H,40H                ;%
        DB      0,30H,48H,84H,84H,48H,30H,30H,49H,85H,82H,82H,45H,39H           ;&
        DB      0,8,8,8,8,0,0,0,0,0,0,0,0,0                                     ;SINGQ
        DB      0,6,8,10H,10H,20H,20H,20H,20H,20H,10H,10H,8,6                   ;(
        DB      0,60H,10H,8,8,4,4,4,4,4,8,8,10H,60H                             ;)
        DB      0,0,0,8,49H,2AH,1CH,1CH,2AH,49H,8,0,0,0                         ;*
        DB      0,0,0,0,8,8,8,7FH,8,8,8,0,0,0                                   ;+
        DB      0,0,0,0,0,0,0,0,18H,18H,8,8,10H,0                               ;COMMA
        DB      0,0,0,0,0,0,0,7FH,0,0,0,0,0,0                                   ;-
        DB      0,0,0,0,0,0,0,0,0,0,0,0,18H,18H                                 ;.
        DB      0,1,2,2,4,4,8,8,10H,10H,20H,20H,40H,40H                         ;/
;NUMERICAL CHARACTERS
;
        DB      0,3CH,42H,42H,81H,83H,85H,89H,91H,0A1H,0C1H,42H,42H,3CH ;0
        DB      0,8,18H,28H,8,8,8,8,8,8,8,8,8,3EH                               ;1
        DB      0,3CH,42H,81H,81H,1,2,4,18H,60H,80H,80H,0F1H,8EH                ;2
        DB      0,0FFH,81H,2,4,8,1CH,2,1,1,1,81H,42H,3CH                        ;3
        DB      0,2,6,0AH,12H,22H,42H,82H,0FFH,2,2,2,2,7                        ;4
        DB      0,0FFH,81H,80H,80H,80H,0BCH,0C2H,1,1,1,81H,42H,3CH              ;5
        DB      0,3CH,42H,81H,80H,80H,0BCH,0C2H,81H,81H,81H,81H,42H,3CH ;6
        DB      0,0FFH,81H,1,2,2,4,8,10H,10H,20H,20H,20H,18H                    ;7
        DB      0,3CH,42H,81H,81H,42H,3CH,42H,81H,81H,81H,81H,42H,3CH   ;8
        DB      0,3CH,42H,81H,81H,81H,81H,43H,3DH,1,1,81H,42H,3CH               ;9
        DB      0,0,0,18H,18H,0,0,0,0,0,18H,18H,0,0                             ;:
        DB      0,0,0,18H,18H,0,0,0,0,0,18H,18H,8,10H                           ;;
        DB      0,2,4,8,10H,20H,40H,80H,40H,20H,10H,8,4,2                       ;<
        DB      0,0,0,0,0,0,7EH,0,0,7EH,0,0,0,0                                 ;=
        DB      0,40H,20H,10H,8,4,2,1,2,4,8,10H,20H,40H                         ;>
        DB      0,3CH,42H,81H,1,1,2,4,8,8,8,0,8,8                               ;?
        DB      0,3CH,42H,81H,81H,9DH,0A5H,0A5H,0A5H,9EH,80H,81H,42H,3CH;@
        DB      0,18H,24H,42H,42H,42H,42H,7EH,42H,42H,42H,42H,42H,0E7H   ;A
        DB      0,0FCH,42H,41H,41H,41H,42H,7CH,42H,41H,41H,41H,42H,0FCH ;B
        DB      0,3CH,42H,81H,81H,80H,80H,80H,80H,80H,81H,81H,42H,3CH    ;C
        DB      0,0FCH,42H,41H,41H,41H,41H,41H,41H,41H,41H,41H,42H,0FCH ;D
        DB      0,0FFH,41H,40H,40H,40H,40H,7CH,40H,40H,40H,40H,41H,0FFH ;E
        DB      0,0FFH,41H,40H,40H,40H,40H,7CH,40H,40H,40H,40H,40H,0E0H ;F[
        DB      0,3CH,42H,81H,81H,80H,80H,80H,80H,8FH,81H,81H,43H,3DH    ;G
        DB      0,0E7H,42H,42H,42H,42H,42H,7EH,42H,42H,42H,42H,42H,0E7H ;H
        DB      0,1CH,8,8,8,8,8,8,8,8,8,8,8,1CH                          ;I
        DB      0,7,2,2,2,2,2,2,2,2,82H,82H,44H,38H                      ;J
        DB      0,0C3H,44H,44H,48H,48H,50H,50H,68H,68H,44H,44H,42H,0E3H ;K
```

```
        DB      0,70H,20H,20H,20H,20H,20H,20H,20H,20H,20H,21H,7FH    ;L
        DB      0,0C1H,63H,63H,55H,55H,49H,49H,41H,41H,41H,41H,0E3H ;M
        DB      00H,0C3H,41H,61H,61H,51H,51H,49H,49H,45H,45H,43H,43H,0E1H;N
        DB      0,3CH,42H,81H,81H,81H,81H,81H,81H,81H,81H,81H,42H,3CH   ;O
        DB      0,0FCH,42H,41H,41H,41H,42H,7CH,40H,40H,40H,40H,40H,0E0H ;P
        DB      0,3CH,42H,81H,81H,81H,81H,81H,81H,85H,85H,82H,45H,39H   ;Q
        DB      0,0FCH,42H,41H,41H,41H,42H,7CH,44H,44H,42H,42H,41H,0E1H ;R
        DB      0,3CH,42H,81H,81H,80H,40H,3CH,2,1,81H,81H,42H,3CH       ;S
        DB      0,7FH,49H,8,8,8,8,8,8,8,8,8,1CH                         ;T
        DB      0,0E3H,41H,41H,41H,41H,41H,41H,41H,41H,41H,41H,22H,1CH  ;U
        DB      0,0E3H,41H,41H,41H,22H,22H,22H,14H,14H,14H,8,8,8        ;V
        DB      0,0DBH,49H,49H,49H,49H,49H,49H,49H,49H,55H,55H,22H,22H  ;W
        DB      0,0E3H,41H,22H,22H,14H,14H,8,14H,14H,22H,22H,41H,0E3H   ;X
        DB      0,0E3H,41H,22H,22H,14H,14H,8,8,10H,10H,20H,0A0H,40H     ;Y
        DB      0,0FFH,82H,2,4,4,8,3CH,10H,10H,20H,20H,41H,0FFH         ;Z
        DB      0,3CH,20H,20H,20H,20H,20H,20H,20H,20H,20H,20H,20H,3CH   ;[
        DB      0,40H,40H,20H,20H,10H,10H,8,8,4,4,2,2,1                 ;\
        DB      0,3CH,4,4,4,4,4,4,4,4,4,4,4,3CH                         ;]
        DB      0,8,1CH,1CH,2AH,2AH,49H,49H,8,8,8,8,8,8                 ;UPAR
        DB      0,0,0,0,0,0,0,0,0,0,0,0FFH,0,0                          ;ULIN
        DB      0,18H,18H,10H,10H,8,0,0,0,0,0,0,0,0                     ;APOS
        DB      0,0,0,0,0,0,7CH,2,2,7EH,82H,82H,82H,7DH                 ;a
        DB      0,0C0H,40H,40H,40H,40H,5CH,62H,41H,41H,41H,41H,62H,0DCH ;b
        DB      0,0,0,0,0,0,3CH,42H,81H,80H,80H,81H,42H,3CH             ;c
        DB      0,6,2,2,2,2,3AH,46H,82H,82H,82H,82H,46H,3BH             ;d
        DB      0,0,0,0,0,0,3CH,42H,81H,0FFH,80H,80H,41H,3EH            ;e
        DB      0,0EH,11H,11H,10H,10H,7EH,10H,10H,10H,10H,10H,10H,38H   ;f
        DB      0,0,3BH,44H,82H,82H,44H,0B8H,80H,7CH,82H,81H,81H,7EH    ;g
        DB      0,0C0H,40H,40H,40H,40H,5CH,62H,41H,41H,41H,41H,41H,0C3H ;h
        DB      0,0,0,8,8,0,18H,8,8,8,8,8,8,1CH                         ;i
        DB      0,4,0,0CH,4,4,4,4,4,4,84H,84H,44H,38H                   ;j
        DB      0,0C0H,40H,40H,40H,40H,43H,44H,48H,50H,68H,44H,42H,0C1H ;k
        DB      0,18H,8,8,8,8,8,8,8,8,8,8,8,1CH                         ;l
        DB      0,0,0,0,0,0,0D6H,69H,49H,49H,49H,49H,49H,0DBH           ;m
        DB      0,0,0,0,0,0,0DCH,62H,41H,41H,41H,41H,41H,0C3H           ;n
        DB      0,0,0,0,0,0,3CH,42H,81H,81H,81H,81H,42H,3CH             ;o
        DB      0,0,0DCH,62H,41H,41H,41H,41H,62H,5CH,40H,40H,40H,0C0H   ;p
        DB      0,0,3AH,46H,82H,82H,82H,82H,46H,3AH,2,2,2,7             ;q
        DB      0,0,0,0,0,0,0CEH,51H,61H,40H,40H,40H,40H,0E0H           ;r
        DB      0,0,0,0,0,0,7EH,81H,80H,7EH,1,1,81H,7EH                 ;s
        DB      0,0,0,10H,10H,10H,7EH,10H,10H,10H,10H,11H,11H,0EH       ;t
        DB      0,0,0,0,0,0,0C6H,42H,42H,42H,42H,42H,26H,1BH            ;u
        DB      0,0,0,0,0,0,0C3H,41H,41H,22H,22H,14H,14H,8              ;v
        DB      0,0,0,0,0,0,0DBH,49H,49H,49H,55H,55H,22H,22H            ;w
        DB      0,0,0,0,0,0,0C3H,42H,24H,18H,18H,24H,42H,0C3H           ;x
        DB      0,0,0C3H,41H,41H,41H,22H,22H,14H,14H,8,8,90H,60H        ;y
        DB      0,0,0,0,0,0,0FFH,82H,4,8H,10H,20H,41H,0FFH              ;z
        DB      0,0EH,10H,20H,20H,20H,20H,40H,20H,20H,20H,20H,10H,0EH   ;{
        DB      0,8,8,8,8,8,8,8,8,8,8,8,8,8                             ;|
        DB      0,70H,8,4,4,4,4,4,2,4,4,4,4,8,70H                       ;}
        DB      0,0,0,0,0,0,6,89H,91H,60H,0,0,0,0                       ;~
        DB      0,55H,0AAH,55H,0AAH,55H,0AAH,55H,0AAH,55H,0AAH,55H,0AAH ;HLF
        DB      55H,0
HCSLEN  EQU     (OFFSET $)-HCSADD       ; LENGTH OF HORIZONTAL CSET
```

APPENDIX B MONITOR MODE CODE ABBREVIATIONS

Displayed Character	ASCII Character	Hex Code	Description
N_U	NULL	00	Null character
S_H	SOH	01	start of heading
S_X	STX	02	start of text
E_X	ETX	03	end of text
E_T	EOT	04	end of transmission
E_Q	ENQ	05	enquiry
A_K	ACK	06	acknowledge
B_L	BEL	07	bell
B_S	BS	08	backspace
H_T	HT	09	horizontal tab
L_F	LF	0A	linefeed
V_T	VT	0B	vertical tab
F_F	FF	0C	formfeed
C_R	CR	0D	carriage return
S_O	SO	0E	shift out
S_I	SI	0F	shift in
D_L	DLE	10	data link escape
D_1	D1	11	device control 1
D_2	D2	12	device control 2
D_3	D3	13	device control 3
D_4	D4	14	device control 4
N_K	NAK	15	negative acknowledgment
S_Y	SYN	16	synchronous idle
E_B	ETB	17	end transmitted block
C_N	CAN	18	cancel
E_M	EM	19	end of medium

S_B	SUB	1A	substitute
E_C	ESC	1B	escape
F_S	FS	1C	file separator
G_S	GS	1D	group separator
R_S	RS	1E	record separator
U_S	US	1F	unit separator
	SP	20	space
	DEL	7F	delete

APPENDIX C

PRINTER CONNECTIONS

RS-232 CRT CONTROLLER BOARD JUMPER CONNECTIONS

The chart on pages C-2 and C-3 shows the correspondence between pins on the CRT board connectors, the CRT board jumper blocks, and the ZSBC board for each of the three serial I/O ports. This information can be used to construct a cable and/or jumper block to connect a printer to the DMS-5000.

Zilog's names for pin abbreviations

RxD	Receive Data
TxD	Transmit Data
CTS	Clear to send
RTS	Request to send
DTR	Data terminal ready
DCD	Data carrier detected

See the Zilog SIO Technical Manual or ZILOG 82/83 Data Book for pin descriptions.

Serial port 1 (CRT board connector G2) is the HiNet port.

Serial port 0 (CRT board connector G5)

CRT board DB25S (G5)	CRT Board jumper block	G9 on CRT brd J1 on ZSBC3 J6 on HNS86	ZSBC 3 Pinouts	HNS86 Pinouts
pin 1	chassis gnd			
pin 2		pin 3	U2 pin 4 RxD I	12F pin 1 RXD I
pin 3	no jumper blk	pin 5	U1 pin 6 TxD O	12G pin 3 TXD O
pin 4	for port 0	pin 1	no connect CTS I	no connect CTS I
pin 5		pin 10	pullup RTS O	pullup RTS O
pin 6		pin 6	pullup DTR O	pullup DTR O
pin 7	signal ground	pin 13	logic GND	logic GND
pin 8		no connect		
pin 20		pin 2	no connect DCD	no connect DCD

Serial port 2 (CRT board connector G3)

CRT board DB25S (G3)	CRT Board jumper block at U3	G9 on CRT brd J1 on ZSBC3 J6 on HNS86	ZSBC3 board pinouts	HNS86 serial ports	5000 CRT controller
pin 1	chassis gnd				
pin 2	pin 1-pin 16	pin 17	U2 pin 13 RxD I	F12 pin 13 RXD I	U10 pin 10 RXD I
pin 3	pin 2-pin 15	pin 24	U1 pin 8 TxD O	G12 pin 11 TXD O	U26 pin 3 TXD O
pin 4	pin 3-pin 14	pin 12	U4 pin 4 CTS I	F12 pin 10 CTS I	U11 pin 10 CTS I
pin 5	pin 4-pin 13	pin 18	U3 pin 3 RTS O	G12 pin 8 RTS O	U26 pin 8 RTS O
pin 6	pin 5-pin 12	pin 15	U1 pin 11 DTR O	G12 pin 6 DTR O	U26 pin 6 DTR O
pin 7	signal ground	pin 13	logic ground	logic ground	logic ground
pin 8	pin 7-pin 10	pullup on CRTc			
pin 11	pin 8				
pin 14	pin 9				
pin 20	pin 6-pin 11	pin 11	U4 pin 13 DCD I	F12 pin 4 DCD I	U10 pin 13 DCD I

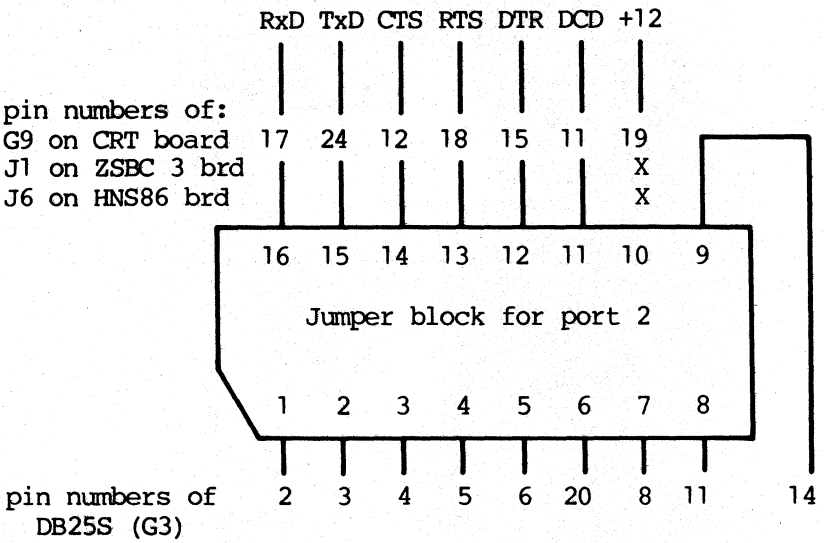
Serial port 3 (CRT board connector G4)

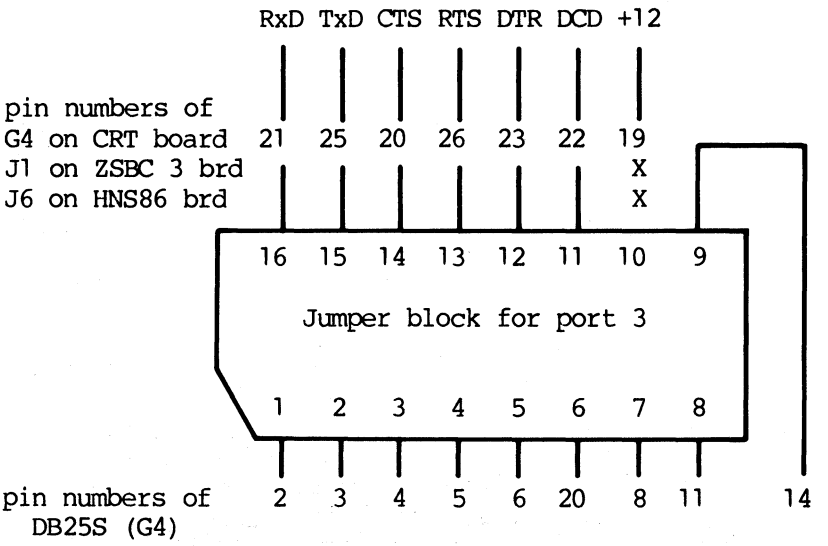
CRT board DB25S (G4)	CRT Board jumper block	G9 on CRT brd J1 on ZSBC 3 J6 on HNS86	ZSBC 3 brd serial ports	HNS86 board serial ports	5000 CRT controller
pin 1					
pin 2	pin 1-pin 16	pin 21	U2 pin 10 RxD	E12 pin 10 RXD I	U11 pin 1 RXD I
pin 3	pin 2-pin 15	pin 25	U3 pin 8 TxD	C12 pin 8 TXD O	U27 pin 8 TXD O
pin 4	pin 3-pin 14	pin 20	U4 pin 1 CTS	E12 pin 1 CTS I	U10 pin 4 CTS I
pin 5	pin 4-pin 13	pin 26	U3 pin 11 RTS	C12 pin 3 RTS O	U27 pin 3 RTS O
pin 6	pin 5-pin 12	pin 23	U3 pin 6 DTR	C12 pin 6 DTR O	U27 pin 6 DTR O
pin 7	signal ground	pin 13	logic ground	logic ground	logic ground
pin 8	pin 7-pin 10	pullup on CRTc			
pin 11	pin 8				
pin 14	pin 9				
pin 20	pin 6-pin 11	pin 22	U4 pin 10 DCD	E12 pin 4 DCD I	U10 pin 1 DCD I

HARDWIRED CONNECTIONS TO JUMPER BLOCK PINS

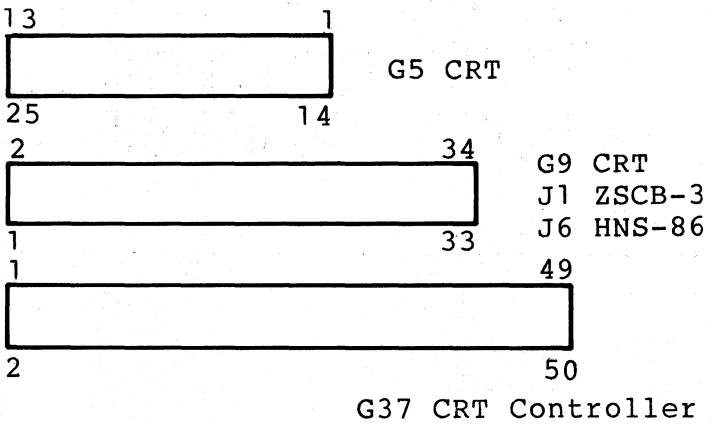
The following diagrams represent the two jumper blocks on the CRT controller board, U3 and U14. For each jumper block, refer to the previous tables for the chip connections

Note--- X represents a 4.7K ohm resistor.
X

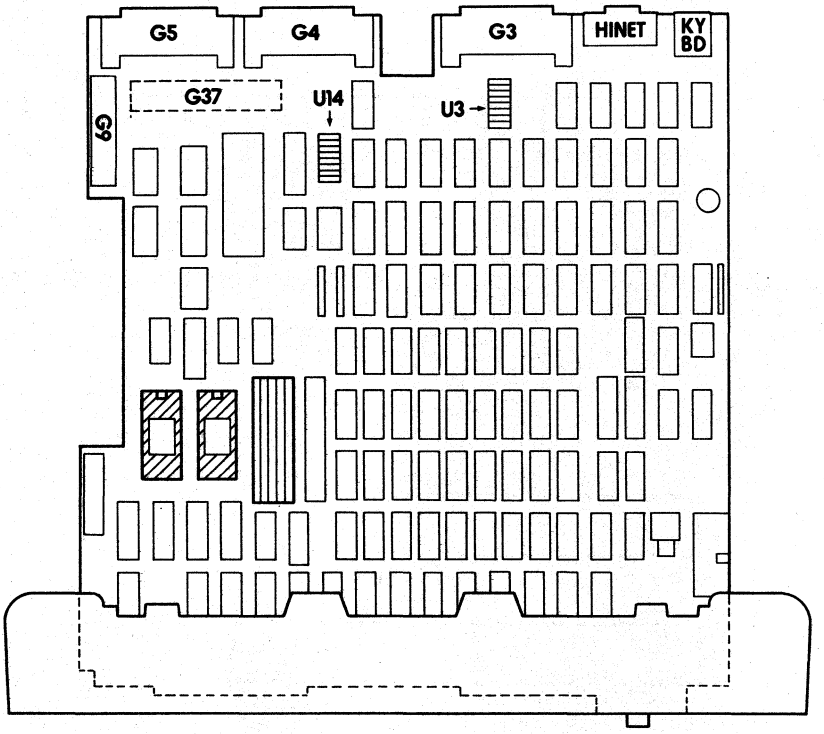




Pin layout and pin numbers for the G5 (CRT board), G9 (CRT board), J1 (ZSCB 3 board), J6 (HNS86 board) and G37 Parallel port.



Inside view of CRT controller board for DMS-5000 Series showing jumper blocks and serial ports.



CHANGING JUMPER BLOCKS

1- Turn off the DMS-5000 and disconnect the power cord, keyboard, HiNet cable, and anything else plugged into the unit.

2- Loosen the thumb-wheel on the base (clockwise turn) and lift the DMS-5000 off of the base. Because it fits tightly you may need to have someone hold down the base while you lift off the unit. Place the DMS-5000 face down (CRT down) on a clean table top. MAKE SURE THERE IS NOTHING ON THE TABLE SURFACE THAT CAN DAMAGE THE SCREEN.

3- The back of the DMS-5000 should now be facing up. Loosen the four screws that are in the middle of the back of the cabinet. Lift off the cover straight up. If the cover does not come up easily the screws need to be loosened some more.

4- If the unit was in use immediately before beginning opening the case, allow 5 minutes to cool with the cover off before proceeding.

5- With the case removed you can see the two green circuit boards on the main chassis. One of them is on the side next to the fan. The second circuit board is away from the fan, and if the DMS-5000 were upright in the vertical operating mode this circuit board would be at the bottom of the unit. This bottom board is where the printer port jumper blocks are located.

6- Your body may be carrying a slight charge of static electricity which might harm the circuit board and chips when you handle the PROMs. Briefly

touch the metal box that the circuit boards are attached to, this will harmlessly drain off any static electricity.

7- As you look at the board, the printer port jumper blocks are located near the ports on the edge of the board away from the casing around the screen. See Diagram C-1. Both jumper blocks have 8 gold coloured metal strips across them. They are labeled on the board **U3** and **U14.** U3 is for Port 2 and U14 is for Port 3.

8- To alter or replace a jumper block you must first remove it from the board. Using a chip puller, hook the feet of the puller under the jumper block's rim and pull straight up. It will come straight out with no twisting or turning. Take care not to damage any of the adjacent circuits. AVOID TOUCHING OTHER PARTS OF THE UNIT, PARTICULARLY THOSE AREAS MARKED **'CAUTION'.**

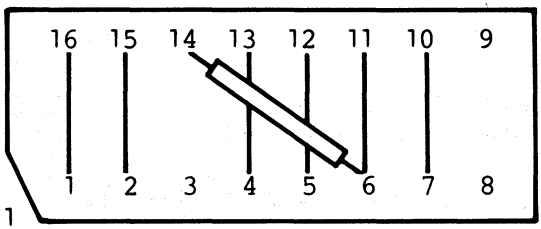
9- Make the appropriate changes to the jumper block or replace it with another one. One corner of a jumper block is removed to indicate the orientation of the block. The beveled corner must be positioned in the upper left corner of the jumper block when the block is inserted in the board and the board is orientated as in Diagram C-1. The jumper block must be inserted correctly or it will not function properly. Push the jumper block gently back into the socket. Make sure none of the pins are bent or twisted.

10- When the new jumper block is inserted, replace the case and tighten the screws. Plug in the power cord, HiNet cable, and keyboard.

EXAMPLES OF JUMPER BLOCKS

Jumper Block for a Texas Instruments 810 printer

The TI 810 printer has an onboard buffer. When the buffer is full, the printer sends a busy signal to the DMS-5000 by lowering DTR on RS-232 pin 20. The serial I/O port on the ZSBC3 or HNS86 will stop transmission automatically if this signal is connected to its CTS input. This is accomplished by connecting pin 14 to pin 6 instead of pin 3 on the corresponding jumper block.



To order, use DMS part number: 90-7309

The DNB option on the TI 810 must be enabled by setting the panel switches to:

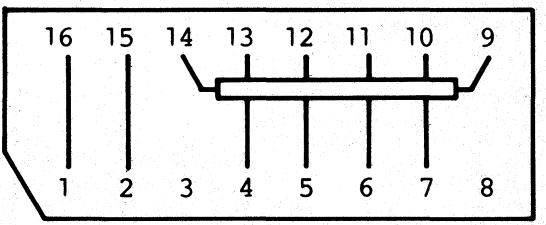
1	2	3	4	5	6	7
low	high	high	low	low	high	high

Remove the processor board of the printer (green card ejectors) and jumper E5 to E6 instead of E4. Verify that the resistor network near the printer's RS-232 connector has not been pulled loose before replacing the card cage enclosure.

NOTE---QUME and TI printers send the same signals; the same jumper blocks can be used.

Jumper block for a Teletype model 40 printer

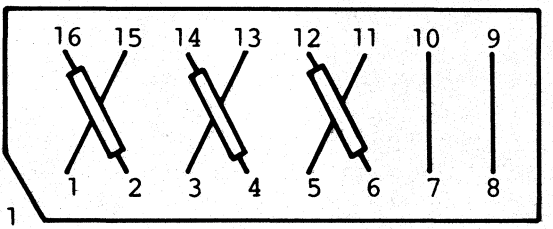
The teletype model 40 uses RS-232 pin 14 to tell the computer that it is busy and no more characters can be sent to it. This signal may be connected to CTS on the proper serial I/O port of the DMS-5000 by connecting pin 14 to pin 9 instead of pin 3 on the corresponding jumper block on the CRT board.



To order, use DMS part number: 90-7310

Jumper block for a standard modem

Most modems require the following pairs of signals to be interchanged (compared to their arrangement for a CRT): RxD and TxD, CTS and RTS, and DCD and DTR.



To order, use DMS part number: 90-7311

PARALLEL PRINTER CABLE

The following table lists the pin connections required to connect a DMS-5000 with a printer using a parallel interface.

In the DMS-5000 Series, the parallel printer interface is driven directly by the CRT controller. It is accessed by preceeding each character sent to the controller by ESC _ (1BH,5FH). The CRT controller will then pass the single character following the ESC sequence to a printer, if there is one connected. If no printer is connected, an internal buffer will fill and no more characters will be passed to the printer or to the CRT controller.

5000 CRT controller	G37	Printer Connector	Centronics Signal Name	In or Out of Printer
PDAT1	Pin 3	Pin 2	DATA 1	Input
PDAT2	Pin 5	Pin 3	DATA 2	Input
PDAT3	Pin 7	Pin 4	DATA 3	Input
PDAT4	Pin 9	Pin 5	DATA 4	Input
PDAT5	Pin 11	Pin 6	DATA 5	Input
PDAT6	Pin 13	Pin 7	DATA 6	Input
PDAT7	Pin 15	Pin 8	DATA 7	Input
PDAT8	Pin 17	Pin 9	DATA 8	Input
ACK/	Pin 19	Pin 10	ACK	Output
BUSY	Pin 21	Pin 11	BUSY	Output
IN 40/	Pin 1	Pin 1	DATA STROBE/	Input
GROUND	Pin 2	Pin 19	DATA STB RET	Ground
GROUND	Pin 4	Pin 20	DATA 1 RET	Ground
GROUND	Pin 6	Pin 21	DATA 2 RET	Ground
GROUND	Pin 8	Pin 22	DATA 3 RET	Ground
GROUND	Pin 10	Pin 23	DATA 4 RET	Ground
GROUND	Pin 12	Pin 24	DATA 5 RET	Ground
GROUND	Pin 14	Pin 25	DATA 6 RET	Ground
GROUND	Pin 16	Pin 26	DATA 7 RET	Ground
GROUND	Pin 22	Pin 29	BUSY	Ground

I/O BYTE FOR HINET

The INPUT/OUTPUT BYTE, at location 03h in memory, is implemented for the mapping of logical to physical devices. The mapping is performed by splitting the I/O BYTE into four distinct fields called the CONSOLE, READER, PUNCH, and LIST fields, as shown below.

Bit number:	7 6 5	4	3	2	1 0
Device name:	LST:	PUN:	--	RDR:	CON:

Whenever a reference to a logical device is made for I/O, the BIOS first looks at the appropriate bit field in the I/O BYTE, and then uses one of the physical device service routines. The following table shows the association of I/O BYTE values with physical devices.

Device Name	bit Value	Physical Device	Device Name Used by STAT
CON:	00 01 10 11	serial port 0 serial port 2 parallel port2 serial port 3	TTY: CRT: BAT: UC1:
RDR:	0 1	serial port 0 serial port 3	TTY: PTR:
PUN:	0 1	serial port 0 serial port 3	TTY: PTP:

HiNet versions 2.245 and previous will not allow a printer to be assigned to the parallel port on the DMS-5000 Series.

The default for the LST: device can be changed with ASSIGN. The other defaults can be changed by using the STAT command. The default can be permanently changed by following the customization instructions in the HiNet manual.

The I/O byte values in the following table assume that the Console is assigned to parallel port 2 (10), the Reader is assigned to serial port 3 (1) and the Punch is assigned to serial port 3 (1). The hex values will be different if the Console, Reader and Punch are assigned to other ports.

List devices for DMS-3/F, DMS-15, DMS-5080, DMS-5086

Device Name	LST: bits	Physical Device	IOBYTE value in: binary	hex
LST:	000	serial port 0	0001 0110	16h
	001	serial port 3	0011 0110	36h
	010	serial port 2	0101 0110	56h
	011	serial port 1	0111 0110	76h
	100	reserved		---
	101	parallel port1	1011 0110	B6h *
	110	custom driver	1101 0110	D6h
	111	HiNet spooler	1111 0110	F6h

* Only for the Fox, no parallel printer port on others

INDEX

DMS-5000 PROGRAMMERS MANUAL ADDENDUM

Changes to ESC codes.

FUNCTION	OLD ESC CODE	NEW ESC CODE	PAGE NUMBERS
To Disable Scaling	ESC #	ESC " (1BH,22H)	2-24,4-6,5-2
Large Horizontal Screen (80x24)	ESC H	ESC M (1BH,4DH)	2-3,4-2,5-1
Reset Keyboard Transposition Table	ESC x	ESC z (1BH,7AH)	2-19,4-6,5-1
Extended RAM Test	ESC ?	ESC ^ (1BH,5EH)	4-8

Display Alignment Pattern--ESC DEL, must be accessed in Local Mode

ESC < and ESC > are DMS in-house test functions and will 'crash' the 5000 if used for any other reasons.

2.7 SECOND PLANE GRAPHICS

With the addition of a new 'daughter board' to the CRT controller board, the DMS 5000 series is capable of displaying characters and graphics in different intensities.

2.7.1 WRITING TO A PLANE

Either plane can be written to by sending the proper ESC code to the CRT controller. When a plane is selected, the blink mode is automatically turned off. Whatever is displayed on the other plane will still be displayed. Therefore, to avoid any undesired overlap, you should clear the screen (ESC C or CTRL L) before changing planes.

ESC } (1BH,7DH) Write to plane 1 only

ESC { (1BH,7BH) Write to plane 2 only.

ESC C (1BH,43H) Clear currently selected plane.

CTRL L (0CH) Clear both planes.

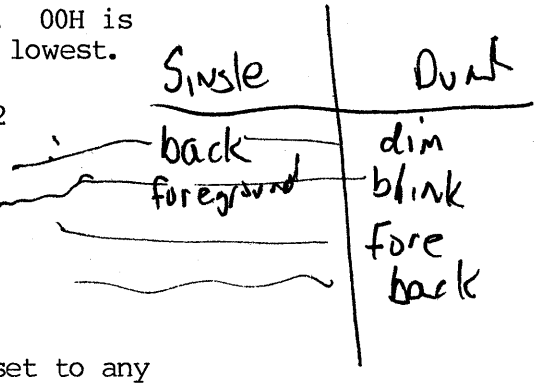
2.7.2 SETTING PLANE INTENSITY

The intensity of the CRT display can be set independently for each plane. There are 15 intensity levels. A register holds the intensity values for four combinations of planes. The levels are set by sending the string:

ESC i num1 num2.

Num1 selects which combination of planes will be effected. Num2 sets the intensity level. The fifteen intensity levels are entered as Hex codes. 00H is the highest intensity setting and 0FH the lowest.

Num1 =		plane 1	plane2
0	0	off	on
1	1	on	on
2	2	on	off
3	3	off	off



The four intensity registers can be set to any of the 15 levels. By default the levels are:

register	intensity
0	06
1	03
2	03
3	0F

NOTE---If there is no daughter board, the 0 register has an intensity value of 0F and the 1 register a value of 03. Plane 2 always appears to be on.

Given these default intensity values, a pixel that was turned:

on in plane 1 and off in plane 2 would be displayed in full-intensity (06).

off in plane 1 and on in plane 2 would be displayed in half-intensity (03).

on in plane 1 and on in plane 2 would be displayed in full-intensity (06).

off in plane 1 and off in plane 2 would be displayed in zero-intensity (0F).

Once you change the intensity values in the register, they will stay that way until you change them again or the 5000 is RESET.

For example, to display a bright graphic image along with dim characters, set the intensities with the sequence:

ESC i 2 0 (set plane one to bright intensity)

ESC i 0 8 (set plane two to dim intensity)

After the intensities are set, select plane one (ESC }) and write the graphic image. Then select plane two (ESC {) and write the characters that you want to be displayed in half-intensity.

2.7.3 HALF-INTENSITY MODE

Sending an ESC H (1BH,48H) causes characters to be written in half-intensity. This is done by lowering the intensity of the back plane relative to the front plane. All characters are then written only to the back plane.

Both ESC H and ESC B will automatically set the screen to character mode.

ESC N (1BH, 4EH) will return the display to normal display mode.

NOTE---You cannot use both blinking and half-intensity displays at the same time. Either mode can be used with character inverse, underline and overstrike modes.

ADJUSTING HALF-INTENSITY

The half-intensity setting can be adjusted either with a CTRL/SHIFT function key or by sending an ESC code. Each time CTRL/SHIFT F6 or F5 is struck or ESC t or ESC u is sent, the screen intensity is changed by one level.

Half-Intensity Up	Half-Intensity Down
CTRL/SHIFT F6	**CTRL/SHIFT F5**
ESC t (1BH,74H)	**ESC u** (1BH,75H)

NOTE---Once the half-intensity setting is changed, either through the keyboard or from a program with an ESC sequence, the new setting will remain in effect until the system is reset.

2.7.4 BLINKING DISPLAY

Sending an ESC B (1BH,42H) will set the display to blinking mode. The intensity value for register 1 is cycled between bright (whatever is set in register 2) and 0F (zero intensity) on a timed basis.

NOTE---Using either Blinking mode or Half-intensity mode will slow down the process of writing to the screen since both planes must be written to for each command that is given. The writing speed will not be increased until you return to normal mode and then clear the screen.

2.3.12 ALTERNATE CHARACTER SETS

Up to three character sets can be downloaded and stored in memory. Number 01H is always considered the main character set. Number 02H and 03H can be any alternate character set. You must download a character set that is appropriate to the screen size that you are displaying (large, small or vertical).

A large horizontal character set is 1344 bytes (540H) in size. The small horizontal character set is 768 bytes (300H) and the vertical is 576 (240H)bytes.

See section 2.3.8 and Appendix A for the format of a down-loadable character set.

DOWNLOADING A CHARCTER SET

To download alternate character sets 02H or 03H, use the sequence:

ESC j setnum numlow numhigh charset

where setnum defines character set 02H or 03H and numlow and numhigh are the number of bytes to be loaded. (Large = 540H, Small = 300H, Vertical = 240H.)

The primary character set (01H) must be downloaded with the sequence:

ESC c (1BH,63H) **[charset]**

as detailed in section 2.3.8.

SELECT CHARACTER SET

Once the alternate character sets are downloaded, you can select which character set is to be displayed. Send the sequence:

ESC J (1BH,4AH) **setnum**

where setnum is either CTRL A for the primary character set, CTRL B for the secondary character set or CTRL C for the third character set.

NOTE---If you select a character set that has not been downloaded, the result will be undesirable (probably a blank screen). You can recover the main character set by entering ESC J CTRL A.

2.4.2 STATUS BYTE

The Status Byte has been redefined for version 1.4. The third bit is set to 1 if the second plane is present and to 0 if the first plane is present. The status byte is now read:

Bit 0 - Orientation
Vertical = 1
Horizontal = 0

Bit 1 - Video Display
Inverse video = 1
Normal video = 0

Bit 2 - Mode
Graphic Mode = 1
Character = 0

Bit 3 - Planes Present
Second Plane Present = 1
One Plane Only = 0

Bit 4 - Unused

Bit 5 - Always 1

Bit 6 - Always 0

Bit 7 - Always 0

NOTE---The low bit is Bit 0 and the high bit is Bit 7.

3.1.2 FUNCTION KEY REPORT

When the host CPU sends the sequence:

ESC g (1BH,67H) **keynum**

to the CRT controller, the controller sends back the length of the string that is programmed into the function key and then the string.

3.2 LOAD FUNCTION KEYS FROM KEYBOARD

Version 1.4 allows you to load function keys directly from the keyboard. CTRL/SHIFT F13 enters the function key load mode. The top line of the screen will clear and the prompt: **Which Function Key?** will be displayed. Press the key that you wish to program. If you enter an invalid key the bell will sound.

When you have selected a key to program, the screen will display:

F(keynumber) Enter String. ^$F14 to delete >.

(The program uses the symbol **$** to denote SHIFT and **^** for CTRL.) Characters are entered after the > symbol. Use the ^$F14 key to delete mistakes. To exit the program, press CTRL/SHIFT F13 again.

Whenever you press ^$F13 and a function key that you wish to program, the contents of the function key are erased. Therefore, you cannot edit a function key's buffer using this method.

Strings can be between 1 and 125 bytes. The maximum amount of RAM storage available for programming the function keys is 1K bytes. Any entries over the 1K limit will not be accepted.

3.3 MEMORY DUMP AND MEMORY LOAD

To Dump the contents currently loaded into the CRT controller's memory, use the sequence:

ESC s (1BH,73H) offset segment length

where offset, segment and length are two byte numbers, with the low byte sent first.

NOTE---The bytes are sent through the host output queue so the queue will change as the process proceeds.

To Download code into memory send the sequence:

ESC > (1BH,3EH) offset segment length

where the offset, segment and length are defined in the same way as the memory dump. The program starts by disabling interrupts. The process reads and loads code of the length stated to the segment and offset named. With the interrupts still disabled, it jumps to the beginning of the loaded code.

NOTE---Do not load to address 0:0 through 0:4 as this contains the jump vector.

5.0 VERSION 1.4 ESCAPE AND CONTROL CODES INDEX

CONTROL CODES INDEX

CONTROL/SHIFT CODES INDEX